BACKWARD-FORWARD ALGORITHMS FOR STRUCTURED MONOTONE INCLUSIONS IN HILBERT SPACES

HÉDY ATTOUCH, JUAN PEYPOUQUET, AND PATRICK REDONT

ABSTRACT. In this paper, we study the *backward-forward* algorithm as a splitting method to solve structured monotone inclusions, and convex minimization problems in Hilbert spaces. It has a natural link with the *forward-backward* algorithm and has the same computational complexity, since it involves the same basic blocks, but organized differently. Surprisingly enough, this kind of iteration arises when studying the time discretization of the regularized Newton method for maximal monotone operators. First, we show that these two methods enjoy remarkable involutive relations, which go far beyond the evident inversion of the order in which the forward and backward steps are applied. Next, we establish several convergence properties for both methods, some of which were unknown even for the forward-backward algorithm. This brings further insight into this well-known scheme. Finally, we specialize our results to structured convex minimization problems, the gradient-projection algorithms, and give a numerical illustration of theoretical interest.

KEYWORDS. Monotone inclusion, forward-backward algorithm, proximal-gradient method

MSC: 47J25, 47J30, 49M15, 49M37, 65K15, 90C25, 90C53

1. INTRODUCTION

The forward-backward algorithm was introduced by Lions and Mercier [25] and Passty [34] in order to find a zero of the sum of two maximal monotone operators. It can be naturally traced back to the projected-gradient method considered by Goldstein [22] and Levitin and Polyak [24] for constrained optimization problems. Each iteration of the algorithm consists of a forward (explicit) step with respect to a cocoercive (thus Lipschitz-continuous) operator B, and a backward (implicit) step with respect to a general maximal monotone operator A. A variant includes an additional relaxation step, which may improve its numerical performance.

Forward-backward algorithms have proved to be efficient tools for solving structured monotone inclusions, and convex minimization problems. They provide parallel splitting methods which can be easily implemented, and which are particularly interesting for large-scale systems. They play an important role in signal and image processing, especially when dealing with sparse optimization. They are also adapted to domain decomposition techniques for PDE's (see [7]).

An important number of contributions, dealing with various topics, have been devoted to the development of this flexible method. The forward-backward-forward algorithm deals with maximal monotone Lipschitz operators B that are not necessarily cocoercive (like linear skew-symmetric operators) with application to Lagrangian methods [17, 13, 41].

Effort sponsored by the Air Force Office of Scientific Research, Air Force Material Command, USAF, under grant number FA9550-14-1-0056, Fondecyt Grant 1140829, Conicyt Anillo ACT-1106, ECOS-Conicyt Project C13E03, Millenium Nucleus ICM/FIC RC130003, Conicyt Project MATHAMSUD 15MATH-02, Conicyt Redes 140183, and Basal Project CMM Universidad de Chile.

FISTA is an acceleration of the FB method based on Nesterov's approach [15, 31, 32] (see also [20] for applications to signal recovery). Approximate data and computational errors are considered in [39] among many others. An inertial forward-backward algorithm is studied in [10]. In [8, 33], the method is coupled with approximation or penalization techniques. Based on Kurdyka-Lojasiewicz property, convergence of forward-backward algorithms has been recently obtained in a nonconvex, nonsmooth setting, for tame optimization and semi-algebraic problems [5, 4, 21, 16]. For a recent account on these methods one can consult [6, 13, 15, 18, 39] and the bibliography therein.

To our knowledge, the existing methods of forward-backward type all consider the explicit step first and the implicit step next. The alternative, a backward-forward algorithm, has not been explored. Surprisingly enough, this kind of iteration arises when studying the time discretization of the regularized Newton method for maximal monotone operators proposed in [11], and thereafter extended to the case of structured monotone operators in [2]. A semi-implicit discretization of the dynamical system studied in [9] (different from the one considered in [10]) produces this type of method as well.

Having in mind this connection with Newton-like systems, our original aim was to study backward-forward algorithms both theoretically and numerically, and assess their performance, especially in connection with traditional forward-backward methods. As research progressed, we found out some remarkable involutive relationships between the forward and backward steps. These properties allow us to understand forward-backward algorithms more deeply, obtain convergence results beyond the classical monotone setting, account for an over-relaxed combination step, and deduce further properties of the limits. When coupled with a relaxation step, which may accelerate convergence, the forward-backward and the backward-forward are different. Yet they share the same computational complexity (a gradient and a proximal operation) and the same convergence properties. They account for the numerical observation that reversing the order of the gradient and the proximal step is not important.

The paper is organized as follows: In Section 2, we describe the forward-backward and backward-forward algorithms, point out some relevant facts concerning set-valued operators, and present some *involutive* relations that allow to consider both algorithms in a somewhat unified manner. Convergence results − old and new − for both algorithms are presented in Section 3 in the operator setting. The case where the operators A and B derive from convex potentials is investigated in Section 4. A numerical illustration is given in Section 5, while further remarks and perspectives are commented in Section 6.

2. FORWARD-BACKWARD AND BACKWARD-FORWARD ALGORITHMS

Throughout this paper, $\mathscr{H}$ is a real Hilbert space with scalar product $\langle .,. \rangle$ and norm $\|\cdot\|$. We mostly adopt the definitions and notations of [13].

Let $F : \mathscr{H} \rightrightarrows \mathscr{H}$ be a set-valued operator. The inverse operator $F^{-1} : \mathscr{H} \rightrightarrows \mathscr{H}$ is defined by the relation $y \in F^{-1}x \Leftrightarrow x \in Fy$.

For $\gamma \in \mathbf{R} \setminus \{0\}$, the *resolvent of F of index γ* is the operator $J_{\gamma F} = (I + \gamma F)^{-1}$. For $\gamma \in \mathbf{R}$, the *Yosida approximation of F of index γ* is $F_\gamma = (F^{-1} + \gamma I)^{-1}$. In other words, $y \in F_\gamma x \Leftrightarrow y \in F(x - \gamma y)$. When $\gamma \neq 0$, F_γ can be seen as the *parallel sum* of F and $\frac{1}{\gamma}I$. Usually, the Yosida approximation F_γ and the resolvent $J_{\gamma F}$ are only defined for $\gamma > 0$. We will need those operator extensions to $\gamma < 0$ and we retain the same vocabulary.

An operator $F : \mathscr{H} \rightrightarrows \mathscr{H}$ that is *everywhere defined* ($F(x) \neq \emptyset$ for each $x \in \mathscr{H}$) and *single-valued* ($F(x)$ is a singleton for each x) will be systematically identified with a *function* $F : \mathscr{H} \to \mathscr{H}$, and viceversa. Even if F is everywhere defined and single-valued, this

need not be the case for $J_{\gamma F}$ and F_γ. However, if F is maximally monotone and $\gamma > 0$, then $J_{\gamma F}$ and F_γ are everywhere defined and single-valued. Moreover, they are Lipschitz-continuous functions with constants 1 and $\frac{1}{\gamma}$, respectively.

The following properties will be useful in the sequel:

Lemma 2.1. *Given $F : \mathscr{H} \rightrightarrows \mathscr{H}$, and $\gamma, \delta \in \mathbf{R}$, we have*

(i) *The resolvent identity: $(F_\gamma)_\delta = F_{\gamma+\delta}$. In particular, $(F_{-\gamma})_\gamma = F$.*
(ii) *For $\gamma \neq 0$, $J_{\gamma F} = I - \gamma F_\gamma$. Hence, F_γ is everywhere defined and single-valued whenever $J_{\gamma F}$ is.*
(iii) *For $\gamma \neq 0$, $J_{\gamma(F_{-\gamma})} = I - \gamma F$. Hence, F is everywhere defined and single-valued whenever $J_{\gamma(F_{-\gamma})}$ is.*

Proof. Part (i) is immediate from the definition:

$$y \in (F_\gamma)_\delta x \Leftrightarrow y \in F_\gamma(x - \delta y) \Leftrightarrow y \in F(x - \gamma y - \delta y) \Leftrightarrow y \in F_{\gamma+\delta} x.$$

For part (ii), since $\gamma \neq 0$, we have

$$y \in J_{\gamma F} x \Leftrightarrow x \in y + \gamma F y \Leftrightarrow \frac{x-y}{\gamma} \in F\left(x - \gamma\frac{x-y}{\gamma}\right) \Leftrightarrow \frac{x-y}{\gamma} \in F_\gamma x \Leftrightarrow y \in x - \gamma F_\gamma x.$$

For (iii), replace F with $F_{-\gamma}$ in (ii) to obtain $J_{\gamma(F_{-\gamma})} = I - \gamma(F_{-\gamma})_\gamma = I - \gamma F$, in view of (i). □

2.1. **The algorithms and their relationship.** We fix $\gamma \in \mathbf{R}$, and take a real sequence $(\lambda_n)_{n\in\mathbf{N}}$. Let $A : \mathscr{H} \rightrightarrows \mathscr{H}$ be a set-valued operator such that $J_{\gamma A}$ is everywhere defined and single-valued, and let $B : \mathscr{H} \to \mathscr{H}$. The *Forward-Backward Algorithm* applied to the pair (A, B) is defined by the iterations:

$$\text{(FB)} \qquad \left\{ \begin{array}{rcl} y_n &=& x_n - \gamma B x_n \\ z_n &=& J_{\gamma A} y_n \\ x_{n+1} &=& x_n + \lambda_n(z_n - x_n), \quad n \geq 0. \end{array} \right.$$

In turn, the *Backward-Forward Algorithm* for (A, B) is given by:

$$\text{(BF)} \qquad \left\{ \begin{array}{rcl} v_n &=& J_{\gamma A} u_n \\ w_n &=& v_n - \gamma B v_n \\ u_{n+1} &=& u_n + \lambda_n(w_n - u_n), \quad n \geq 0. \end{array} \right.$$

Notice that (FB) and (BF) only differ by the order in which operators $I - \gamma B$ and $J_{\gamma A}$ are applied. The reason for introducing (FB) and (BF) is to generate sequences, namely x_n and v_n, converging to a zero of the sum $A + B$.

Loosely speaking, (FB) applied to (A, B) is (BF) applied to $(B_{-\gamma}, A_\gamma)$, and, conversely, (BF) applied to (A, B) is (FB) applied to $(B_{-\gamma}, A_\gamma)$. More precisely, we have the following:

Theorem 2.2. *Take $\gamma \neq 0$ and a real sequence $(\lambda_n)_{n\in\mathbf{N}}$. Let $A : \mathscr{H} \rightrightarrows \mathscr{H}$ be such that $J_{\gamma A}$ is everywhere defined and single-valued, and let $B : \mathscr{H} \to \mathscr{H}$. Then*

(i) *For each $x_0 \in \mathscr{H}$, algorithm (FB) applied to (A, B) uniquely defines a sequence (x_n, y_n, z_n). For each $u_0 \in \mathscr{H}$, algorithm (BF) applied to $(B_{-\gamma}, A_\gamma)$ uniquely defines a sequence (u_n, v_n, w_n). Moreover, if $u_0 = x_0$, then $(x_n, y_n, z_n) = (u_n, v_n, w_n)$ for all $n \geq 0$.*
(ii) *For each $u_0 \in \mathscr{H}$, algorithm (BF) applied to (A, B) uniquely defines a sequence (u_n, v_n, w_n). For each $x_0 \in \mathscr{H}$, algorithm (FB) applied to $(B_{-\gamma}, A_\gamma)$ uniquely defines a sequence (x_n, y_n, z_n). Moreover, if $u_0 = x_0$, then $(x_n, y_n, z_n) = (u_n, v_n, w_n)$ for all $n \geq 0$.*

Proof. Let us prove (i). The uniqueness of the sequence (x_n, y_n, z_n) follows obviously from the hypotheses. In view of Lemma 2.1 (ii), with $F = A$, and of Lemma 2.1 (iii), with $F = B$, the sequence (u_n, v_n, w_n) is uniquely defined (A_γ and $J_{\gamma B_{-\gamma}}$ are single-valued and everywhere defined) and satisfies the same defining relations as (x_n, y_n, z_n) does, namely:

$$v_n = J_{\gamma B_{-\gamma}} u_n = u_n - \gamma B u_n,$$

$$w_n = v_n - \gamma A_\gamma v_n = J_{\gamma A} v_n,$$

and

$$u_{n+1} = u_n + \lambda_n (w_n - u_n).$$

For (ii), it suffices to apply (i), replacing (A, B) with $(B_{-\gamma}, A_\gamma)$.

□

2.2. **The stationary points of (FB) and (BF).** Given a set-valued operator $F : \mathscr{H} \rightrightarrows \mathscr{H}$, the *set of zeros* of F is $\operatorname{zer}(F) := F^{-1}(0)$, and its *set of fixed points* is $\operatorname{fix}(F) := \{x \in \mathscr{H},\ x \in F(x)\}$.

As before, take $\gamma \in \mathbf{R}$, a real sequence $(\lambda_n)_{n\in\mathbf{N}}$, a set-valued operator $A : \mathscr{H} \rightrightarrows \mathscr{H}$ such that $J_{\gamma A}$ is everywhere defined and single-valued, and $B : \mathscr{H} \to \mathscr{H}$. If we set

$$T = J_{\gamma A} \circ (I - \gamma B) \quad \text{and} \quad S = (I - \gamma B) \circ J_{\gamma A},$$

then (FB) and (BF) reduce to

$$x_{n+1} = x_n + \lambda_n (T x_n - x_n), \quad n \geq 0, \tag{FB}$$

and

$$u_{n+1} = u_n + \lambda_n (S u_n - u_n), \quad n \geq 0, \tag{BF}$$

respectively.

The following result characterizes the stationary points of (FB) and (BF), *i. e.* the fixed points of T and S:

Proposition 2.3. *Take $\gamma \neq 0$. Let $A : \mathscr{H} \rightrightarrows \mathscr{H}$ be such that $J_{\gamma A}$ is everywhere defined and single-valued, and let $B : \mathscr{H} \to \mathscr{H}$. Then*

(i) $x \in \operatorname{fix}(T) \Leftrightarrow x \in \operatorname{zer}(A + B) \Leftrightarrow A_\gamma \circ (I - \gamma B)x + Bx = 0$;
(ii) $y \in \operatorname{fix}(S) \Leftrightarrow y \in \operatorname{zer}(B_{-\gamma} + A_\gamma) \Leftrightarrow B \circ J_{\gamma A} y + A_\gamma y = 0$;
(iii) $I - \gamma B : \operatorname{zer}(A + B) \to \operatorname{zer}(B_{-\gamma} + A_\gamma)$ *is a bijection with inverse* $J_{\gamma A}$.

Proof. (i) On the one hand, $x \in \operatorname{fix}(T) \Leftrightarrow x = J_{\gamma A} \circ (I - \gamma B)x \Leftrightarrow x - \gamma B x \in x + \gamma A x \Leftrightarrow x \in \operatorname{zer}(A + B)$.
On the other hand, $x \in \operatorname{fix}(T) \Leftrightarrow x \in \operatorname{fix}[(I - \gamma A_\gamma) \circ (I - \gamma B)] \Leftrightarrow x = x - \gamma B x - \gamma A_\gamma \circ (I - \gamma B)x \Leftrightarrow A_\gamma \circ (I - \gamma B)x + Bx = 0$.
(ii) Operator $B_{-\gamma}$ is such that $J_{\gamma B_{-\gamma}}$ is single-valued by Lemma 2.1 (iii). Further, by Lemma 2.1 (ii), A_γ is also single-valued since $J_{\gamma A}$ is. Applying (i) with (A, B) replaced with $(B_{-\gamma}, A_\gamma)$, and using Lemma 2.1 (ii) and (iii), yields the desired result.
(iii) Take $x \in \operatorname{zer}(A + B)$. From (i), we deduce $A_\gamma \circ (I - \gamma B)x + B \circ J_{\gamma A} \circ (I - \gamma B)x = 0$. Now, using (ii), we obtain $(I - \gamma B)x \in \operatorname{zer}(A_\gamma + B_{-\gamma})$. Similarly, for $y \in \operatorname{zer}(A_\gamma + B_{-\gamma})$, by (ii), we have $B \circ J_{\gamma A} y + A_\gamma \circ (I - \gamma B) \circ J_{\gamma A} y = 0$, and so $J_{\gamma A} y \in \operatorname{zer}(A + B)$, by (i). Finally, $J_{\gamma A} \circ (I - \gamma B)x = x$, and $(I - \gamma B) \circ J_{\gamma A} y = y$ for $x \in \operatorname{zer}(A + B)$ and $y \in \operatorname{zer}(A_\gamma + B_{-\gamma})$ are mere rewritings of the fixed point equalities $Tx = x$ and $Sy = y$. □

3. CONVERGENCE: OPERATOR SETTING

Theorem 2.2 displays a complete, if formal, symmetry between (FB) and (BF). The temptation is to deduce convergence properties of algorithm (BF), which is new, from those of (FB), which has already been studied in the literature. But when it comes to convergence, the usual context is: A maximally monotone and B cocoercive. Then the symmetry is broken since $B_{-\gamma}$ need not be maximally monotone, which prevents from deriving convergence properties of (BF) applied to the pair (A,B) from those, hypothetical, of (FB) applied to ($B_{-\gamma}$,A_γ). So we are led to enlarge the class of maximally monotone operators A so that, first, $B_{-\gamma}$ will fall into this new class, and, second, that (FB) will have good convergence properties. So doing, we show new convergence results for this algorithm. See subsection 3.4 below for further discussion.

For $\rho \in \mathbf{R}$, a set-valued operator $F : \mathscr{H} \rightrightarrows \mathscr{H}$ is *maximally ρ-cohypomonotone* if $F_\rho = (F^{-1} + \rho I)^{-1}$ is maximally monotone. (On cohypomonotonicity, its usefulness and related notions see [13, Prop. 23.12], [19, 23, 40], [38, 12.28 Ex.]). Notice that, if F is maximally ρ-cohypomonotone, with $\rho \leq 0$, then F is maximally monotone; indeed: $F = (F_\rho)_{-\rho}$.

Let $\beta > 0$. A function $F : \mathscr{H} \to \mathscr{H}$ is *β-cocoercive* if $\langle Fx_2 - Fx_1, x_2 - x_1\rangle \geq \beta \|Fx_2 - Fx_1\|^2$ for every x_1, x_2 in $\mathscr{H}$.

We have the following:

Lemma 3.1. *Let $\gamma \in \mathbf{R}$ and $\alpha > 0$. Let $F : \mathscr{H} \rightrightarrows \mathscr{H}$. The following are equivalent:*

(i) *F is maximally $(\gamma - \alpha)$-cohypomonotone ;*
(ii) *F_γ is everywhere defined, single-valued and α-cocoercive.*

Proof. (i)⇒(ii) Let $u_1 \in F_\gamma x_1$. Then, with Lemma 2.1 (i), $u_1 \in (F_{\gamma-\alpha})_\alpha x_1$; hence $u_1 \in F_{\gamma-\alpha}(x_1 - \alpha u_1)$. So, for $u_2 \in F_\gamma x_2$, we have $\langle u_2 - u_1, x_2 - \alpha u_2 - (x_1 - \alpha u_1)\rangle \geq 0$, hence $\langle u_2 - u_1, x_2 - x_1\rangle \geq \alpha \|u_2 - u_1\|^2$. As a consequence F_γ is single-valued. It is also everywhere defined as the Yosida approximate of index $\alpha > 0$ of the maximally monotone operator $F_{\gamma-\alpha}$.
(ii)⇒(i) Let $u_1 \in F_{\gamma-\alpha} x_1 = (F_\gamma)_{-\alpha} x_1$; then $u_1 \in F_\gamma(x_1 + \alpha u_1)$. So, for $u_2 \in F_{\gamma-\alpha} x_2$ we have $\langle u_2 - u_1, x_2 + \alpha u_2 - (x_1 + \alpha u_1)\rangle \geq \alpha \|u_2 - u_1\|^2$, hence $\langle u_2 - u_1, x_2 - x_1\rangle \geq 0$ and $F_{\gamma-\alpha}$ is monotone . Further $I - \alpha F_\gamma = I - \alpha(F_{\gamma-\alpha})_\alpha = J_{\alpha F_{\gamma-\alpha}} = (I + \alpha F_{\gamma-\alpha})^{-1}$. Consequently $\operatorname{ran}(I + \alpha F_{\gamma-\alpha}) = \operatorname{dom}(I - \alpha F_\gamma) = \mathscr{H}$ and $F_{\gamma-\alpha}$ is maximally monotone by Minty's Theorem ([27]).

□

In order to ensure the convergence of the (FB) and (BF) algorithms, we make the following standing assumptions:

Assumption 1.
The parameters α, β and γ satisfy $0 < \gamma < 2\min\{\alpha, \beta\}$.
The operator $A : \mathscr{H} \rightrightarrows \mathscr{H}$ is maximally $(\gamma - \alpha)$-cohypomonotone.
The function $B : \mathscr{H} \to \mathscr{H}$ is β-cocoercive.
$\operatorname{zer}(A + B) \neq \emptyset$.
The sequence $(\lambda_n)_{n\in\mathbf{N}}$ of relaxation parameters satisfies $0 \leq \lambda_n \leq \delta$ and $\sum_{n\in\mathbf{N}} \lambda_n(\delta - \lambda_n) = +\infty$, where

$$\delta = \frac{1}{2} + \frac{1}{\gamma}\min\{\alpha, \beta\}.$$

Some comments are in order:

(1) Since $\delta > 1$, the setting accounts for over-relaxation.

(2) The operator $B_{-\gamma}: \mathscr{H} \rightrightarrows \mathscr{H}$ is such that $(B_{-\gamma})_\gamma = B$ is everywhere defined single-valued and β-cocoercive, while A_γ is α-cocoercive. Further, with Proposition 2.3 (iii): $\operatorname{zer}(A+B) \neq \emptyset \Leftrightarrow \operatorname{zer}(B_{-\gamma}+A_\gamma) \neq \emptyset$. Therefore, the pair $(B_{-\gamma}, A_\gamma)$ satisfies the same assumptions as (A,B), up to a permutation of α and β.

(3) The operator B is constant on $\operatorname{zer}(A+B)$. Indeed, if x_1 and x_2 are two zeros of $A+B$, by Proposition 2.3 (i), we have $-Bx_1 = A_\gamma(x_1 - \gamma Bx_1)$ and $-Bx_2 = A_\gamma(x_2 - \gamma Bx_2)$. In view of the α-cocoerciveness of A_γ, this entails

$$\langle -Bx_2 + Bx_1, x_2 - \gamma Bx_2 - (x_1 - \gamma Bx_1)\rangle \geq \alpha \| - Bx_2 + Bx_1\|^2.$$

Further, the β-cocoerciveness of B gives

$$\gamma \| - Bx_2 + Bx_1\|^2 - \beta \| - Bx_2 + Bx_1\|^2 \geq \alpha \| - Bx_2 + Bx_1\|^2.$$

We deduce that $\|Bx_2 - Bx_1\| = 0$ because $\gamma < \alpha + \beta$. Likewise, A_γ is constant on $\operatorname{zer}(B_{-\gamma} + A_\gamma)$.

3.1. **Convergence of the forward-backward algorithm.**

Theorem 3.2. *Let Assumption 1 hold and let $(x_n, y_n, z_n)_{n\in\mathbf{N}}$ be generated by the (FB) Algorithm applied to (A,B). We have the following:*

(i) $\|z_n - x_n\| \to 0$.
(ii) $x_n \rightharpoonup x \in \operatorname{zer}(A+B)$, *with* $A_\gamma \circ (I - \gamma B)x + Bx = 0$.
(iii) $Bx_n \to Bx$.
(iv) $z_n \rightharpoonup x$ *and* $Bz_n \to Bx$.
(v) $y_n \rightharpoonup y \in \operatorname{zer}(B_{-\gamma} + A_\gamma)$, *with* $B \circ J_{\gamma A} y + A_\gamma y = 0$.
(vi) $x = J_{\gamma A} y$, $y = x - \gamma Bx$.
(vii) $A_\gamma y_n \to A_\gamma y = -Bx$.

Proof. (i) - (ii): Set $T = J_{\gamma A} \circ (I - \gamma B)$. Algorithm (FB) reads

$$x_{n+1} = x_n + \lambda_n (Tx_n - x_n) \text{ with } Tx_n = z_n.$$

Since A_γ and B are respectively α-cocoercive and β-cocoercive, $J_{\gamma A} = I - \gamma A_\gamma$ and $I - \gamma B$ are respectively $\gamma/2\alpha$-averaged and $\gamma/2\beta$-averaged ([13, Proposition 4.33]). Hence T is $1/\delta$-averaged ([13, Proposition 4.32]), and is relevant to the extension of the Krasnoselskii-Mann Theorem given in [13, Proposition 5.15]. So, $\|Tx_n - x_n\| = \|z_n - x_n\| \to 0$ and x_n converges weakly to a fixed point x of T. We conclude using Proposition 2.3 (i).
(iii): From $z_n = (I - \gamma A_\gamma)(x_n - \gamma Bx_n)$ we deduce that $\frac{1}{\gamma}(x_n - z_n) - Bx_n = A_\gamma(x_n - \gamma Bx_n)$; moreover $-Bx = A_\gamma(x - \gamma Bx)$ by Proposition 2.3 (i). In view of the α-cocoerciveness of A_γ we have

$$\langle \frac{1}{\gamma}(x_n - z_n) - (Bx_n - Bx), x_n - x - \gamma(Bx_n - Bx)\rangle \geq \alpha \| \frac{1}{\gamma}(x_n - z_n) - (Bx_n - Bx)\|^2.$$

An elementary computation gives

$$\langle \frac{1}{\gamma}(x_n - z_n), \zeta_n\rangle - \langle Bx_n - Bx, x_n - x\rangle \geq (\alpha - \gamma)\|Bx_n - Bx\|^2,$$

with $\zeta_n = (x_n - x) + (2\alpha - \gamma)(Bx_n - Bx)$. In view of the β-cocoerciveness of B we have further

$$\langle \frac{1}{\gamma}(x_n - z_n), \zeta_n\rangle \geq (\alpha + \beta - \gamma)\|Bx_n - Bx\|^2.$$

Now, the sequence $(\zeta_n)_{n\in\mathbf{N}}$ is bounded: indeed the sequences $(x_n)_{n\in\mathbf{N}}$, $(Bx_n)_{n\in\mathbf{N}}$ are bounded, the latter because B is $1/\beta$-Lipschitz continuous. Hence $\|Bx_n - Bx\| \to 0$ in view of (i) and

since $\gamma < \beta + \alpha$.
(iv): The first assertion follows from (i) and (ii). The second assertion follows from the β-cocoerciveness of B: $\langle Bz_n - Bx_n, z_n - x_n\rangle \geq \beta\|Bz_n - Bx_n\|^2$, along with (i) and (iii).
(v) - (vi): From points (ii) and (iii), it is clear that $y_n = x_n - \gamma Bx_n$ converges weakly to $y = x - \gamma Bx$. The fixed point equality for x reads $x = J_{\gamma A} \circ (I - \gamma B)x = J_{\gamma A}y$. Applying the operator $I - \gamma B$ to the equality $x = J_{\gamma A}y$ yields $y = (I - \gamma B) \circ J_{\gamma A}y$, which shows that y is a fixed point of $(I - \gamma B) \circ J_{\gamma A}$, hence a zero of $B_{-\gamma} + A_\gamma$, which satisfies $B \circ J_{\gamma A}y + A_\gamma y = 0$ by Proposition 2.3 (ii).
(vii): Write $A_\gamma y_n = \frac{1}{\gamma}(y_n - J_{\gamma A}y_n) = \frac{1}{\gamma}((y_n - x_n) + (x_n - J_{\gamma A}y_n)) = -Bx_n + \frac{1}{\gamma}(x_n - J_{\gamma A}y_n)$ to conclude that $A_\gamma y_n \to -Bx$ with (i) and (iii). Finally, from (v), we deduce that $-Bx = (y - J_{\gamma A}y)/\gamma = A_\gamma y$.

□

When A is maximally monotone, choosing $\alpha = \gamma$, we have the following:

Corollary 3.3. *Let $A : \mathscr{H} \rightrightarrows \mathscr{H}$ be maximally monotone, let $B : \mathscr{H} \to \mathscr{H}$ be β-cocoercive, and assume $\operatorname{zer}(A+B) \neq \emptyset$. Take $0 < \gamma < 2\beta$, and let the sequence of relaxation parameters satisfy $0 \leq \lambda_n \leq \delta$ and $\sum_{n\in\mathbf{N}} \lambda_n(\delta - \lambda_n) = +\infty$, with $\delta = \min\{1, \beta/\gamma\} + 1/2$. Then, all the conclusions of Theorem 3.2 remain true for every sequence $(x_n, y_n, z_n)_{n\in\mathbf{N}}$ generated by the (FB) Algorithm applied to (A, B).*

3.2. **Convergence of the backward-forward algorithm.**

Theorem 3.4. *Let Assumption 1 hold and let $(u_n, v_n, w_n)_{n\in\mathbf{N}}$ be generated by the (BF) Algorithm applied to (A, B). We have the following:*

(i) $\|w_n - u_n\| \to 0$.
(ii) $u_n \rightharpoonup u \in \operatorname{zer}(B_{-\gamma} + A_\gamma)$, *with* $B \circ J_{\gamma A}u + A_\gamma u = 0$.
(iii) $A_\gamma u_n \to A_\gamma u$.
(iv) $w_n \rightharpoonup u$ *and* $A_\gamma w_n \to A_\gamma u$.
(v) $v_n \rightharpoonup v \in \operatorname{zer}(A + B)$, *with* $A_\gamma \circ (I - \gamma B)v + Bv = 0$.
(vi) $u = v - \gamma Bv$, $v = J_{\gamma A}u$.
(vii) $Bv_n \to Bv = -A_\gamma u$.

Proof. By Theorem 2.2 (ii), Algorithm (BF) applied to (A, B) is Algorithm (FB) applied to $(B_{-\gamma}, A_\gamma)$. Since the pair $(B_{-\gamma}, A_\gamma)$ satisfies the same assumptions as (A, B), up to a permutation of α and β, the theorem is proved by applying Theorem 3.2 to $(B_{-\gamma}, A_\gamma)$ in place of (A, B).

□

For a maximally monotone A we have:

Corollary 3.5. *Let $A : \mathscr{H} \rightrightarrows \mathscr{H}$ be maximally monotone, let $B : \mathscr{H} \to \mathscr{H}$ be β-cocoercive, and assume $\operatorname{zer}(A+B) \neq \emptyset$. Take $0 < \gamma < 2\beta$, and let the sequence of relaxation parameters satisfy $0 \leq \lambda_n \leq \delta$ and $\sum_{n\in\mathbf{N}} \lambda_n(\delta - \lambda_n) = +\infty$, with $\delta = \min\{1, \beta/\gamma\} + 1/2$. Then, all the conclusions of Theorem 3.4 remain true for every sequence $(u_n, v_n, w_n)_{n\in\mathbf{N}}$ generated by the (FB) Algorithm applied to (A, B).*

3.3. **Inexact computation of the iterates.** The (FB) algorithm can be written as

$$x_{n+1} = P_n^{FB}(x_n), \qquad \text{where} \qquad P_n^{FB}(x) = (1 - \lambda_n)x + \lambda_n J_{\gamma A}(x - \gamma B(x)).$$

In a similar fashion, the (BF) algorithm can be expressed as

$$u_{n+1} = P_n^{BF}(u_n), \qquad \text{with} \qquad P_n^{BF}(u) = (1 - \lambda_n)u + \lambda_n(J_{\gamma A}(u) - \gamma B(J_{\gamma A}(u))).$$

A simple computation shows that for each n, the operators P_n^{FB} and P_n^{BF} are nonexpansive, whenever $\lambda_n \in [0,1]$. As a consequence of [3, Proposition 6.1] (see also [35, Lemma 5.3]), we have the following:

Corollary 3.6. *Let $\lambda_n \leq 1$ for all sufficiently large n, and let $(\varepsilon_n) \in \ell^1$. Then*

(1) *If $(\hat{x}_n)$ satisfies $\|\hat{x}_{n+1} - P_n^{FB}(\hat{x}_n)\| \leq \varepsilon_n$ for all n, then the conclusions of Theorem 3.2 hold for $(\hat{x}_n)$ and the corresponding $(\hat{y}_n, \hat{z}_n)$.*
(2) *If $(\hat{u}_n)$ satisfies $\|\hat{u}_{n+1} - P_n^{BF}(\hat{u}_n)\| \leq \varepsilon_n$ for all n, then the conclusions ot Theorem 3.4 hold for $(\hat{u}_n)$ and the corresponding $(\hat{v}_n, \hat{w}_n)$.*

Additive errors may appear at any stage of the iteration due to computational or measurement imprecisions, namely:

$$(FB_\varepsilon) \qquad x_{n+1} = (1-\lambda_n)x_n + \lambda_n J_{\gamma A}(x_n - \gamma B(x_n + \xi^1) + \xi_n^2) + \xi_n^3.$$

$$(BF_\varepsilon) \qquad u_{n+1} = (1-\lambda_n)u_n + \lambda_n (J_{\gamma A}(u_n + \zeta_n^1) - \gamma B(J_{\gamma A}(u_n + \zeta_n^1) + \zeta_n^2) + \zeta_n^3.$$

If $\sum_{i=1}^3 \sum_{n=1}^\infty \|\xi_n^i\| < +\infty$, or, correspondingly, $\sum_{i=1}^3 \sum_{n=1}^\infty \|\zeta_n^i\| < +\infty$, the convergence remains unaltered.

3.4. **Discussion.**

3.4.1. *About the assumptions.* Central to the preceding study are the class $\mathscr{C}(\alpha,\beta)$ of pairs of operators (A,B) satisfying Assumption 1, and the transform $\mathscr{D}_\gamma : (A,B) \to (B_{-\gamma}, A_\gamma)$ acting from $\mathscr{C}(\alpha,\beta)$ to $\mathscr{C}(\beta,\alpha)$. Adopting this general framework enables a complete parallelism between algorithms (FB) and (BF), and justifies the assumptions imposed on operator A.

A maximal monotone A satisfies its share of Assumption 1 with $\alpha = \gamma$. But supposing A maximally monotone from the onset, although more consistent with the literature, would not guarantee the symmetry between (FB) and (BF), since then $B_{-\gamma}$ need not be monotone. A direct proof of Theorem 3.4 would have to be given, that would be completely parallel to, but different from, the proof of Theorem 3.2.

It is worth mentioning that Corollary 3.5 is a consequence of Corollary 3.3 when $0 < \gamma \leq \beta$, because, in this (more constraining) case, $B_{-\gamma}$ is maximally monotone. Indeed, since $B : \mathscr{H} \to \mathscr{H}$ is β-cocoercive, $B_{-\gamma}$ is maximally $(\gamma-\beta)$-cohypomonotone by Lemma 3.1. Then, by definition, $B_{-\beta} = (B_{-\gamma})_{\gamma-\beta}$ is maximally monotone. Hence $B_{-\gamma} = (B_{-\beta})_{\beta-\gamma}$ is also maximally monotone.

It is likely that we have made the most from that frame since transform $\mathscr{D}_\gamma$ is involutive by Lemma 2.1 (i), namely: $((A_\gamma)_{-\gamma}, (B_{-\gamma})_\gamma) = (A,B)$.

3.4.2. *Novelty.* The symmetry between (FB) and (BF) is not only mathematically pleasant. Beyond the study of the (BF) Algorithm, it allows to discover the convergence of the auxiliary sequences $(y_n)_{n\in\mathbf{N}}$ and $(u_n)_{n\in\mathbf{N}}$, and to identify their limits as zeros of $B_{-\gamma} + A_\gamma$. To our knowledge, this fact is new. Only conclusions (i), (ii) and (iii) of Theorem 3.2 concerning (FB) seem to be reported in the literature (e. g. [13, Theorem 25.8]). Theorem 3.4 is new.

3.4.3. *On the relaxation parameters.* If $\lambda_n \equiv 1$, and with consistent initial values, Algorithms (FB) and (BF) essentially generate the same sequences. Of course, when $\lambda_n \not\equiv 1$ there is no reason why sequences $(x_n, y_n)_{n\in\mathbf{N}}$ and $(v_n, u_n)_{n\in\mathbf{N}}$ should coincide. Theorems 3.2 and 3.4 show that their asymptotic behavior is similar and that their limits enjoy the same properties.

3.4.4. *Summing up.* To emphasize the symmetry between (FB) and (BF) we summarize below the algorithms with the main convergence results.

$\mathbf{FB}(A,B)$	$\mathbf{BF}(A,B)$
$y_n = x_n - \gamma B x_n$	$v_n = J_{\gamma A}\, u_n$
$z_n = J_{\gamma A}\, y_n$	$w_n = v_n - \gamma B x_n$
$x_{n+1} = x_n + \lambda (z_n - x_n)$	$u_{n+1} = u_n + \lambda (w_n - u_n)$
Convergence results under Assumption 1	
$\|z_n - x_n\| \to 0$	$\|w_n - u_n\| \to 0$
$x_n \rightharpoonup x \in \operatorname{zer}(A+B)$	$u_n \rightharpoonup u \in \operatorname{zer}(B_{-\gamma} + A_\gamma)$
$Bx_n \to Bx$	$A_\gamma u_n \to A_\gamma u$
$y_n \rightharpoonup y \in \operatorname{zer}(B_{-\gamma} + A_\gamma)$	$v_n \rightharpoonup v \in \operatorname{zer}(A+B)$
$A_\gamma y_n \to A_\gamma y$	$Bv_n \to Bv$

4. Convergence: Functional framework

In this section, we study the case where the operators A and B derive from convex potentials. The following assumptions will be in force:

Assumption 2.
The parameters β and γ satisfy $0 < \gamma < 2\beta$.
The function $f : \mathscr{H} \to \mathbf{R} \cup \{+\infty\}$ is proper, convex, and lower-semicontinuous.
The function $g : \mathscr{H} \to \mathbf{R}$ is differentiable and its gradient ∇g is β-cocoercive.
$\operatorname{Argmin}(f+g) \neq \emptyset$.
The sequence $(\lambda_n)_{n\in\mathbf{N}}$ of relaxation parameters satisfies $0 \leq \lambda_n \leq \delta$ and $\sum_{n\in\mathbf{N}} \lambda_n(\delta - \lambda_n) = +\infty$, where

$$\delta = \frac{1}{2} + \min\{1, \beta/\gamma\}.$$

Remark 4.1.

(1) The operator $A = \partial f$ is maximally monotone by [37, Theorem A] (see also [36, Example 1.3]). In turn, by the Baillon-Haddad Theorem [12, Corollaire 10], ∇g is β-cocoercive if, and only if, g is convex and ∇g is Lipschitz-continuous with constant $1/\beta$ (see also [35, Theorem 3.13]).
(2) Assumption 1 is satisfied with $\alpha = \gamma$, $A = \partial f$, $B = \nabla g$ (and $\delta = 1/2 + \min\{1, \beta/\gamma\}$). Indeed, in view of the maximal monotonicity of A, the operator A_γ is everywhere defined, single-valued and γ-cocoercive, ([13, Corollary 23.10 (iii)]). Obviously, $0 < \gamma < 2\gamma$. Finally, the Moreau-Rockafellar Theorem (see, for instance, [35, Theorem 3.30]) gives $\partial(f+g) = \partial f + \nabla g$, hence $\operatorname{zer}(A+B) = \operatorname{zer}(\partial f + \nabla g) = \operatorname{zer}(\partial(f+g)) = \operatorname{Argmin}(f+g) \neq \emptyset$.
(3) If f_γ denotes the Moreau envelope of f, then $\nabla f_\gamma = (I - \operatorname{Prox}_{\gamma f})/\gamma$ (see [13, Chapter 12]). Hence $\nabla f_\gamma = A_\gamma$.

Since $J_{\gamma A} = \operatorname{Prox}_{\gamma f}$ (see, for instance, [13, Proposition 16.34]), the forward-backward algorithm applied to $(\partial f, \nabla g)$ gives

$$\text{(FB)} \qquad \left\{ \begin{array}{rcl} y_n &=& x_n - \gamma \nabla g(x_n) \\ z_n &=& \operatorname{Prox}_{\gamma f} y_n \\ x_{n+1} &=& x_n + \lambda_n (z_n - x_n). \end{array} \right.$$

Since no confusion should arise, we shall say (FB) applied to (f,g) instead of $(\partial f, \nabla g)$, just for simplicity. Similarly, the backward-forward algorithm applied to $(\partial f, \nabla g)$ (or (f,g)) is

$$\text{(BF)} \qquad \left\{ \begin{array}{rcl} v_n & = & \mathrm{Prox}_{\gamma f} u_n \\ w_n & = & v_n - \gamma \nabla g(v_n) \\ u_{n+1} & = & u_n + \lambda_n (w_n - u_n). \end{array} \right.$$

Beyond the results of Section 3, which may be invoked here with $A = \partial f$ and $B = \nabla g$, new properties are brought out in the current functional context. In particular, the values of f and g are made precise as algorithms (FB) and (BF) proceed.

4.1. **Convergence of the forward-backward algorithm.**

Theorem 4.2. *Let Assumption 2 hold and let $(x_n, y_n, z_n)_{n\in\mathbf{N}}$ be generated by the (FB) Algorithm applied to (f,g). We have the following:*

(i) $\|z_n - x_n\| \to 0$.
(ii) $x_n \rightharpoonup x \in \mathit{Argmin}(f+g)$, *with* $\nabla f_\gamma(x - \gamma\nabla g(x)) + \nabla g(x) = 0$.
(iii) $\nabla g(x_n) \to \nabla g(x)$.
(iv) $z_n \rightharpoonup x$ *and* $\nabla g(z_n) \to \nabla g(x)$.
(v) $y_n \rightharpoonup y$ *with* $\nabla f_\gamma(y) + \nabla g(\mathit{Prox}_{\gamma f} y) = 0$.
(vi) $x = \mathit{Prox}_{\gamma f} y$, $y = x - \gamma\nabla g(x)$.
(vii) $\nabla f_\gamma(y_n) \to \nabla f_\gamma(y) = -\nabla g(x)$.
(viii) $g(x_n) \to g(x)$.
(ix) $g(z_n) \to g(x)$.
(x) $f(z_n) \to f(x)$.
(xi) *If, moreover,* $x_0 \in \mathit{dom} f$ *and* $0 \le \lambda_n \le 1$, *then* $(f+g)(x_n)$ *is nonincreasing.*
(xii) *If, furthermore,* $\limsup \lambda_n > 0$, *then* $(f+g)(x_n) \to (f+g)(x) = \inf(f+g)$ *and* $f(x_n) \to f(x)$.

Proof. Parts (i) through (vii) follow from Theorem 3.2 and Remark 4.1.

In view of the weak convergences $x_n \rightharpoonup x$, $z_n \rightharpoonup x$, and the lower-semicontinuity of f and g for the weak topology, we have

$$g(x) \le \liminf g(x_n), \quad g(x) \le \liminf g(z_n), \quad f(x) \le \liminf f(z_n). \tag{1}$$

(viii): We use the subgradient inequality for g at point x_n and the strong convergence in (iii), to obtain

$$\begin{array}{rcl} g(x) & \ge & g(x_n) + \langle x - x_n, \nabla g(x_n)\rangle \\ & \ge & g(x_n) + \langle x - x_n, \nabla g(x)\rangle + \langle x - x_n, \nabla g(x_n) - \nabla g(x)\rangle \\ & \ge & \limsup g(x_n). \end{array}$$

This yields the desired result, in view of the first inequality in (1).
(ix): The subgradient inequality for g at point z_n yields

$$g(x_n) \ge g(z_n) + \langle x_n - z_n, \nabla g(z_n)\rangle.$$

Now $(\nabla g(z_n))_{n\in\mathbf{N}}$ is a bounded sequence because ∇g is Lipschitz continuous. Hence, with (i) and (viii), $g(x) = \lim g(x_n) \ge \limsup g(z_n)$. Together with the second inequality in (1), this yields the desired result.
(x): From $z_n = \mathrm{Prox}_{\gamma f}(x_n - \gamma\nabla g(x_n))$ we deduce that $\frac{1}{\gamma}(x_n - z_n) - \nabla g(x_n) \in \partial f(z_n)$. The

subgradient inequality for f at point z_n yields

$$\begin{aligned} f(x) &\geq f(z_n) + \langle x - z_n, \frac{1}{\gamma}(x_n - z_n) - \nabla g(x_n)\rangle \\ &\geq f(z_n) + \langle x - z_n, \frac{1}{\gamma}(x_n - z_n) - (\nabla g(x_n) - \nabla g(x)\rangle + \langle x - z_n, -\nabla g(x)\rangle. \end{aligned}$$

Hence, with (i), (ii) and (iii), $f(x) \geq \limsup f(z_n)$. Together with the last inequality in (1), this yields the desired result.
(xi): Point x_{n+1} is then a convex combination of x_n and $z_n = \mathrm{Prox}_{\gamma f} y_n \in \mathrm{dom} f$. If $x_0 \in \mathrm{dom} f$, a straightforward induction shows that $x_n \in \mathrm{dom} f$, $\forall n \in \mathbf{N}$. The subgradient inequality for f at point z_n yields

$$\begin{aligned} f(z_n) &\leq f(x_n) + \langle z_n - x_n, \frac{1}{\gamma}(x_n - z_n) - \nabla g(x_n)\rangle \\ &= f(x_n) - \frac{1}{\gamma}\|z_n - x_n\|^2 - \langle z_n - x_n, \nabla g(x_n)\rangle. \end{aligned}$$

Further, the Descent Lemma (see, for instance, [35, Lemma 1.30]) yields

$$g(z_n) \leq g(x_n) + \langle z_n - x_n, \nabla g(x_n)\rangle + \frac{1}{2\beta}\|z_n - x_n\|^2.$$

Adding the previous two inequalities we obtain

$$(f+g)(z_n) + \left(\frac{1}{\gamma} - \frac{1}{2\beta}\right)\|z_n - x_n\|^2 \leq (f+g)(x_n). \tag{2}$$

Since x_{n+1} is a convex combination of x_n and z_n, we obtain by convexity of $f+g$

$$(f+g)(x_{n+1}) \leq (1-\lambda_n)(f+g)(x_n) + \lambda_n (f+g)(z_n). \tag{3}$$

With (2) we finally get

$$(f+g)(x_{n+1}) \leq (f+g)(x_n) - \lambda_n \left(\frac{1}{\gamma} - \frac{1}{2\beta}\right)\|z_n - x_n\|^2.$$

which shows that $(f+g)(x_n)$ is nonincreasing.
(xii): Rewrite inequality (3) as

$$(f+g)(x_{n+1}) + \lambda_n (f+g)(x_n) \leq (f+g)(x_n) + \lambda_n (f+g)(z_n).$$

Set $l = \inf(f+g)(x_n) = \lim(f+g)(x_n)$ and let $n \to +\infty$; we obtain

$$l + (\limsup \lambda_n)\, l \leq l + (\limsup \lambda_n)(f+g)(x).$$

Hence $l \leq (f+g)(x)$. But obviously $l \geq (f+g)(x)$.

□

4.2. **Convergence of the backward-forward algorithm.**

Theorem 4.3. *Let Assumption 2 hold and let $(u_n, v_n, w_n)_{n\in\mathbf{N}}$ be generated by the (BF) Algorithm applied to (f,g). We have the following:*

(i) $\|w_n - u_n\| \to 0$.
(ii) $u_n \rightharpoonup u$, *with* $\nabla f_\gamma(u) + \nabla g(Prox_{\gamma f}(u)) = 0$.
(iii) $\nabla f_\gamma(u_n) \to \nabla f_\gamma(u)$.
(iv) $w_n \rightharpoonup u$ *and* $\nabla f_\gamma(w_n) \to \nabla f_\gamma(u)$.
(v) $v_n \rightharpoonup v \in Argmin(f+g)$; *further* $\nabla f_\gamma(v - \gamma\nabla g(v)) + \nabla g(v) = 0$.
(vi) $u = v - \gamma\nabla g(v)$, $v = Prox_{\gamma f} u$.
(vii) $\nabla g(v_n) \to \nabla g(v) = -\nabla f_\gamma(u)$.

(viii) $(f+g)(v_n) \to (f+g)(v) = \inf(f+g)$, $f(v_n) \to f(v)$ *and* $g(v_n) \to g(v)$.
(ix) *If* $0 \leq \lambda_n \leq 1$, *then* $(f+g)(v_n)$ *is nonincreasing.*

Proof. As before, (i)-(vii) are straightforword consequences of Theorem 3.4 and Remark 4.1.
(viii): Firstly, we claim that $(f+g)(v_n) \to (f+g)(v)$.
On the one hand, from the weak convergence in (v) and from the lower semicontinuity of $(f+g)$, we deduce $(f+g)(v) \leq \liminf(f+g)(v_n)$.
On the other hand, from $v_n = \mathrm{Prox}_{\gamma f} u_n$ and $w_n = v_n - \gamma \nabla g(v_n)$ we deduce

$$u_n \in v_n + \gamma \partial f(v_n) = w_n + \gamma \nabla g(v_n) + \gamma \partial f(v_n) = w_n + \gamma \partial (f+g)(v_n). \tag{4}$$

Hence, the subgradient inequality for $f+g$ at point v_n reads

$$(f+g)(v) \geq (f+g)(v_n) + \langle v - v_n, \frac{1}{\gamma}(u_n - w_n) \rangle.$$

Whence we derive, in view of (i), $(f+g)(v) \geq \limsup(f+g)(v_n)$, which proves our first claim.

Secondly we claim that $f(v_n) \to f(v)$. On the one hand, from the weak convergence in (v) and from the lower semicontinuity of f, we deduce $f(v) \leq \liminf f(v_n)$.
On the other hand, with (4), the subgradient inequality for f at point v_n yields

$$\begin{aligned} f(v) &\geq f(v_n) + \langle v - v_n, \frac{1}{\gamma}(u_n - w_n) - \nabla g(v_n) \rangle \\ &\geq f(v_n) + \langle v - v_n, \frac{1}{\gamma}(u_n - w_n) - (\nabla g(v_n) - \nabla g(v)) \rangle + \langle v - v_n, -\nabla g(v) \rangle. \end{aligned}$$

Whence we derive, in view of (i), (v), (vii), $f(v) \geq \limsup f(v_n)$, which proves our second claim and the desired result.
(ix): With (4), we have $(u_{n+1} - v_{n+1})/\gamma \in \partial f(v_{n+1})$, and the subgradient inequality for f at point v_{n+1} gives

$$f(v_{n+1}) \leq f(v_n) + \langle \frac{1}{\gamma}(u_{n+1} - v_{n+1}), v_{n+1} - v_n \rangle.$$

The descent lemma for g at point v_n yields

$$g(v_{n+1}) \leq g(v_n) + \langle \nabla g(v_n), v_{n+1} - v_n \rangle + \frac{1}{2\beta} \|v_{n+1} - v_n\|^2.$$

Adding the two inequalities above we obtain

$$\begin{aligned} &(f+g)(v_{n+1}) \leq \\ &(f+g)(v_n) + \langle \frac{1}{\gamma}(u_{n+1} - v_{n+1}) + \nabla g(v_n), v_{n+1} - v_n \rangle + \frac{1}{2\beta} \|v_{n+1} - v_n\|^2. \end{aligned} \tag{5}$$

From the definition of u_{n+1} in (BF) we derive

$$\nabla g(v_n) = \frac{1}{\gamma}(v_n - u_n) - \frac{1}{\lambda_n \gamma}(u_{n+1} - u_n).$$

Hence

$$\frac{1}{\gamma}(u_{n+1} - v_{n+1}) + \nabla g(v_n) = -\frac{1 - \lambda_n}{\lambda_n \gamma}(u_{n+1} - u_n) - \frac{1}{\gamma}(v_{n+1} - v_n).$$

Inequality (5) now reads

$$(f+g)(v_{n+1}) \leq \\ (f+g)(v_n) - \frac{1-\lambda_n}{\lambda_n \gamma}\langle u_{n+1}-u_n, v_{n+1}-v_n\rangle - \left(\frac{1}{\gamma}-\frac{1}{2\beta}\right)\|v_{n+1}-v_n\|^2.$$

Since $\mathrm{Prox}_{\gamma f}$ is 1-cocoercive ([13, Prop. 4.2 (i)(iv)])), we have $\langle u_{n+1}-u_n, v_{n+1}-v_n\rangle \geq 0$, hence $((f+g)(v_n))$ is a nonincreasing sequence.

□

Remark 4.4. Partial conclusions of the preceding theorem can be found in [1].

4.3. **Case $0<\gamma\leq\beta$.** Given $\lambda \in \mathbf{R}$ and $h : \mathscr{H} \to \mathbf{R}\cup\{+\infty\}$, define $h_\lambda = (h^* + \frac{\lambda}{2}\|.\|^2)^*$ where ψ^* denotes the Fenchel conjugate of a function ψ (see [13, Definition 13.1]). The notation is consistent with the notation for the Moreau envelope recalled in Remark 4.1 since, for $\lambda > 0$ and convex h, the function $(h^* + \frac{\lambda}{2}\|.\|^2)^*$ coincides with the Moreau envelope of h ([13, Proposition 14.1]).

The definition of h_λ is parallel to the expression for A_γ given in Section 2. Yet, while the transform $A \to A_\gamma$ has good properties for all $\gamma \in \mathbf{R}$, the transform $h \to h_\lambda$ requires regularity properties for h and a restricted domain for λ, to be interesting. The following lemma provides details and a more tractable expression for h_λ in some cases.

Lemma 4.5. *Let $0<\gamma\leq\beta$ and let $g : \mathscr{H} \to \mathbf{R}$ be convex and differentiable, with ∇g being β-cocoercive.*

(i) *For $\lambda \geq -\beta$, $\mu \in \mathbf{R}$, we have $(g_\lambda)_\mu = g_{\lambda+\mu}$. In particular, $(g_{-\gamma})_\gamma = g$.*
(ii) *$g_{-\gamma}$ is convex, lower semicontinuous, proper and $Prox_{\gamma g_{-\gamma}} = I - \gamma\nabla g$.*
(iii) *For all $x \in \mathscr{H}$, $g_{-\gamma}(x) = \sup_{\xi\in\mathscr{H}}\{g(\xi) - \frac{1}{2\gamma}\|x-\xi\|^2\}$.*

Proof. Notice that ∇g is γ-cocoercive as well.
(i): With [13, Corollary 13.33], g^* is convex, lower semicontinuous and proper; similarly for $g^* + \frac{\lambda}{2}\|.\|^2$: indeed, if $-\beta \leq \lambda < 0$ this is [13, Theorem 18.15 (v)(vii)], otherwise it is obvious. Hence $(g_\lambda)^* = (g^* + \frac{\lambda}{2}\|.\|^2)^{**} = g^* + \frac{\lambda}{2}\|.\|^2$ and $(g_\lambda)_\mu = ((g_\lambda)^* + \frac{\mu}{2}\|.\|^2)^* = (g^* + \frac{\lambda+\mu}{2}\|.\|^2)^* = g_{\lambda+\mu}$.
Using $\lambda = -\gamma$ and $\mu = \gamma$, we have $(g_{-\gamma})_\gamma = g_0 = g^{**} = g$.
(ii): These are straightforward consequences of [13, Theorem 18.15 (viii) - (ix)].
(iii): Since $g = (g_{-\gamma})_\gamma$, for all $\zeta \in \mathscr{H}$ we have

$$g(\zeta) = g_{-\gamma}(y) + \frac{1}{2\gamma}\|\zeta - y\|^2, \text{ with } y = \mathrm{Prox}_{\gamma g_{-\gamma}}\zeta. \tag{6}$$

Besides, for $(x,\xi) \in \mathscr{H}\times\mathscr{H}$ and $0 < \gamma \leq \beta$ define the functions $\Gamma_\gamma(x,\xi) : \mathscr{H}\times\mathscr{H} \to \mathbf{R}$ and $G_\gamma : \mathscr{H} \to \mathbf{R}$ by

$$\Gamma_\gamma(x,\xi) = \frac{1}{2\gamma}\|x-\xi\|^2 - g(\xi)$$

and

$$G_\gamma(x) = -\inf_{\xi\in\mathscr{H}}\Gamma_\gamma(x,\xi) = \sup_{\xi\in\mathscr{H}}\{g(\xi) - \frac{1}{2\gamma}\|x-\xi\|^2\},$$

respectively.

First, suppose $0<\gamma<\beta$. For fixed (γ,x), the function $\Gamma_\gamma(x,.)$ is differentiable and also strictly convex and coercive since $\Gamma_\gamma(x,.) = \Gamma_\beta(x,.) + \frac{\beta-\gamma}{2\beta\gamma}\|x-.\|^2$, with $\Gamma_\beta(x,.)$ convex ([13, Theorem 18.15 (vi)]). Hence $\Gamma_\gamma(x,.)$ attains its minimum at a unique point ξ which

satisfies $\frac{1}{\gamma}(x-\xi)-\nabla g(\xi)=0$; equivalently $x=\xi-\gamma\nabla g(\xi)=\mathrm{Prox}_{\gamma g_{-\gamma}}\xi$, in view of (ii). Consequently, for any $x\in\mathscr{H}$, we have

$$G_\gamma(x)=-\frac{1}{2\gamma}\|x-\xi\|^2+g(\xi),\ \text{with } x=\mathrm{Prox}_{\gamma g_{-\gamma}}\xi. \tag{7}$$

Now, setting $\zeta=\xi$ in (6) we get: $g(\xi)=g_{-\gamma}(x)+\frac{1}{2\gamma}\|\xi-x\|^2$. Comparing this equality with (7) we obtain $g_{-\gamma}(x)=G_\gamma(x)$, $\forall x\in\mathscr{H}$.

Finally the case $\gamma=\beta$ is dealt with by a limit process, possibly in $\mathbf{R}\cup\{+\infty\}$: On the one hand, $g_{-\beta}=\sup_{\gamma<\beta}(g_{-\beta})_{\beta-\gamma}$ ([13, Proposition 12.32 (i)]); hence $g_{-\beta}=\sup_{\gamma<\beta}g_{-\gamma}$. On the other hand, $\Gamma_\beta(x,\xi)=\inf_{\gamma<\beta}\Gamma_\gamma(x,\xi)$, since the function $\gamma\mapsto\Gamma_\gamma(x,\xi)$ is decreasing and continuous. Hence

$$\begin{aligned}G_\beta(x) &= -\inf_{\xi\in\mathscr{H}}\Gamma_\beta(x,\xi)=-\inf_{\xi\in\mathscr{H}}\inf_{\gamma<\beta}\Gamma_\gamma(x,\xi)=-\inf_{\gamma<\beta}\inf_{\xi\in\mathscr{H}}\Gamma_\gamma(x,\xi)\\ &= -\inf_{\gamma<\beta}(-G_\gamma(x))=\sup_{\gamma<\beta}G_\gamma(x).\end{aligned}$$

We conclude $g_{-\beta}=G_\beta$, since $g_{-\gamma}=G_\gamma$ for $0<\gamma<\beta$. □

Proposition 4.6. *Let $0<\gamma\le\beta$. Let $f:\mathscr{H}\to\mathbf{R}\cup\{+\infty\}$ be convex, lower semicontinuous and proper. Let $g:\mathscr{H}\to\mathbf{R}$ be convex, differentiable with ∇g β-cocoercive. Then*

(i) *$I-\gamma\nabla g:Argmin(f+g)\to Argmin(f_\gamma+g_{-\gamma})$ is a bijection with inverse $Prox_{\gamma f}$.*
(ii) $\inf(f+g)=\inf(g_{-\gamma}+f_\gamma)$.

Proof. (i): Set $A=\partial f$ and $B=\nabla g$. Then $\mathrm{Argmin}(f+g)=\mathrm{zer}(A+B)$. We claim that $\mathrm{Argmin}(f_\gamma+g_{-\gamma})=\mathrm{zer}(A_\gamma+B_{-\gamma})$. Since $A_\gamma=\nabla f_\gamma$ it is sufficient to show that $B_{-\gamma}=\partial g_{-\gamma}$. Lemma 4.5 (ii) tells us that $g_{-\gamma}$ is convex, lower semicontinuous, proper and justifies the second step in the following computation

$$\begin{aligned}y\in\partial g_{-\gamma}(x) &\Leftrightarrow x+\gamma y\in x+\gamma\partial g_{-\gamma}(x)\\ &\Leftrightarrow x=\mathrm{Prox}_{\gamma g_{-\gamma}}(x+\gamma y)=x+\gamma y-\gamma\nabla g(x+\gamma y)\\ &\Leftrightarrow y=\nabla g(x+\gamma y)=B(x+\gamma y)\\ &\Leftrightarrow y\in B_{-\gamma}(x).\end{aligned}$$

The conclusion should now be clear from Remark 4.1 and Proposition 2.3 (iii).

(ii): A simple calculation gives

$$\begin{aligned}\inf_y\{g_{-\gamma}(y)+f_\gamma(y)\} &= \inf_y\{g_{-\gamma}(y)+\inf_x\{f(x)+\frac{1}{2\gamma}\|x-y\|^2\}\}\\ &= \inf_{(y,x)}\{g_{-\gamma}(y)+f(x)+\frac{1}{2\gamma}\|x-y\|^2\}\\ &= \inf_x\{f(x)+\inf_y\{g_{-\gamma}(y)+\frac{1}{2\gamma}\|x-y\|^2\}\}\\ &= \inf_x\{f(x)+(g_{-\gamma})_\gamma(x)\}.\end{aligned}$$

The conclusion follows from Lemma 4.5 (i).

□

Let Assumption 2 be in force in this section now, except that the inequality $0<\gamma<2\beta$ is strenghtened to $0<\gamma\le\beta$ (notice that then $\delta=3/2$).

Remark 4.7. We already know that $\mathrm{Prox}_{\gamma f} = I - \gamma\nabla f_\gamma$, but, with Lemma 4.5 (ii), we have further $I - \gamma\nabla g = \mathrm{Prox}_{\gamma g_{-\gamma}}$. Taking this fact into account, let us rewrite algorithms (FB) and (BF) applied to (f,g):

$$\text{(FB)} \qquad \left\{ \begin{array}{rcccl} y_n & = & x_n - \gamma\nabla g(x_n) & = & \mathrm{Prox}_{\gamma g_{-\gamma}} x_n \\ z_n & = & \mathrm{Prox}_{\gamma f} y_n & = & y_n - \gamma\nabla f_\gamma(y_n) \\ x_{n+1} & = & x_n + \lambda_n(z_n - x_n) & = & x_n + \lambda_n(z_n - x_n). \end{array} \right.$$

$$\text{(BF)} \qquad \left\{ \begin{array}{rcccl} v_n & = & \mathrm{Prox}_{\gamma f} u_n & = & u_n - \gamma\nabla f_\gamma(u_n) \\ w_n & = & v_n - \gamma\nabla g(v_n) & = & \mathrm{Prox}_{\gamma g_{-\gamma}}(v_n) \\ u_{n+1} & = & u_n + \lambda_n(w_n - u_n) & = & u_n + \lambda_n(w_n) - u_n). \end{array} \right.$$

Consequently, for suitable initial values, namely $x_0 = u_0$:

(1) The sequence (x_n, y_n) generated by algorithm (FB) applied to (f,g) is exactly the same as the sequence (u_n, v_n) generated by algorithm (BF) applied to $(g_{-\gamma}, f_\gamma)$.
(2) The sequence (u_n, v_n) generated by algorithm (BF) applied to (f,g) is exactly the same as the sequence (x_n, y_n) generated by algorithm (FB) applied to $(g_{-\gamma}, f_\gamma)$.

This remark allows to complete the properties of algorithm (FB) in Theorem 4.2 in the light of the properties of algorithm (BF) in Theorem 4.3, and conversely, provided that $g_{-\gamma}$ and f_γ meet the required assumptions, whenever f and g do.

Lemma 4.8. *Let (f,g) satisfy Assumption 2 with $0 < \gamma \leq \beta$. Then $(g_{-\gamma}, f_\gamma)$ satisfies Assumption 2 with $\gamma = \beta$.*

Proof. With Lemma 4.5 (ii), $g_{-\gamma}$ is convex, lower semicontinuous and proper, while f_γ is convex, differentiable with ∇f_γ γ-cocoercive. Further, $\delta = \min(1,1) + 1/2 = 3/2$. By Proposition 4.6 (i), $\mathrm{Argmin}(f+g) \neq \emptyset \Rightarrow \mathrm{Argmin}(f_\gamma + g_{-\gamma}) \neq \emptyset$.

□

Theorem 4.9 (Theorem 4.2 completed)**.** *Under the hypotheses of Theorem 4.2 plus $0 < \gamma \leq \beta$, the following additional conclusions hold:*

(i) $y_n \rightharpoonup y \in \mathit{Argmin}(g_{-\gamma} + f_\gamma)$.
(ii) $(g_{-\gamma} + f_\gamma)(y_n) \to (g_{-\gamma} + f_\gamma)(y) = \inf(g_{-\gamma} + f_\gamma)$, $g_{-\gamma}(y_n) \to g_{-\gamma}(y)$, $f_\gamma(y_n) \to f_\gamma(y)$.
(iii) *If* $0 \leq \lambda_n \leq 1$, *then* $(g_{-\gamma} + f_\gamma)(y_n)$ *is nonincreasing.*

Proof. Apply Lemma 4.8 and Therorem 4.3 to $(g_{-\gamma}, f_\gamma)$.

□

Theorem 4.10 (Theorem 4.3 completed)**.** *Under the hypotheses of Theorem 4.3 plus $0 < \gamma \leq \beta$, the following additional conclusions hold:*

(i) $u_n \rightharpoonup u \in \mathit{Argmin}(g_{-\gamma} + f_\gamma)$.
(ii) $f_\gamma(u_n) \to f_\gamma(u)$.
(iii) $f_\gamma(v_n - \gamma\nabla g(v_n)) \to f_\gamma(u)$.
(iv) $g_{-\gamma}(v_n - \gamma\nabla g(v_n)) \to g_{-\gamma}(u)$.
(v) *If, moreover,* $u_0 \in \mathit{dom}(g_{-\gamma})$ *and* $0 \leq \lambda_n \leq 1$ *then* $(g_{-\gamma} + f_\gamma)(u_n)$ *is nonincreasing.*
(vi) *If, furthermore,* $\limsup \lambda_n > 0$ *then* $(g_{-\gamma} + f_\gamma)(u_n) \to (g_{-\gamma} + f_\gamma)(u) = \inf(g_{-\gamma} + f_\gamma)$ *and* $g_{-\gamma}(u_n) \to g_{-\gamma}(u)$.

Proof. Apply Lemma 4.8 and Theorem 4.2 to $(g_{-\gamma}, f_\gamma)$.

□

4.4. **Gradient-projection algorithms.** When $f = \delta_C$ is the indicator function of a nonempty closed convex set C, the proximity operator of δ_C is the projection operator onto C, Proj_C. The (FB) and (BF) algorithms read

$$(BF) \qquad v_n = \mathrm{Proj}_C(u_n), \quad w_n = v_n - \gamma\nabla g(v_n), \quad u_{n+1} = u_n + \lambda_n(w_n - u_n),$$

and

$$(FB) \qquad y_n = x_n - \gamma\nabla g(x_n), \quad z_n = \mathrm{Proj}_C(y_n), \quad x_{n+1} = x_n + \lambda_n(z_n - x_n),$$

respectively. The latter corresponds to a relaxed form of the projected gradient method from [22, 24].

Figures 1 and 2 below illustrate how the two subiterations are performed. The thick line represents part of the boundary of the set C, while the ellipses depict the level sets of the function g. The point x^* minimizes g over C. The triangle in long dashes depicts the (BF) iteration, while the one in short dashes corresponds to the (FB) iteration. We use $\lambda < 1$ in Figure 1 and $\lambda > 1$ in Figure 2.

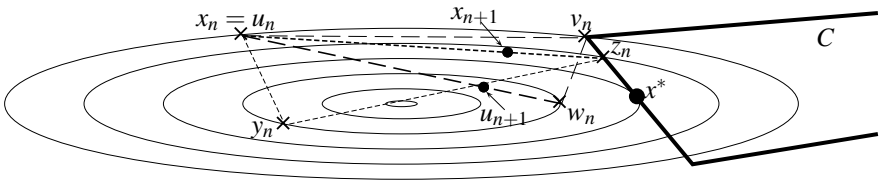

FIGURE 1

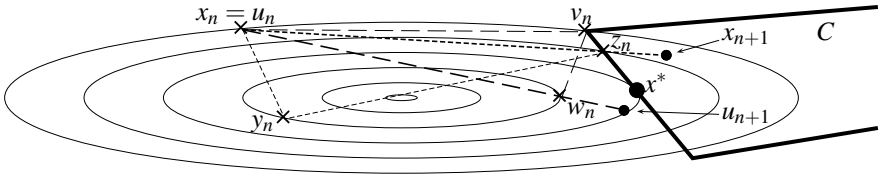

FIGURE 2

4.5. **Discussion.**

4.5.1. *Symmetry lost.* The pleasant symmetry of algorithms (FB) and (BF) and of Theorems 3.2 and 3.4 in the operator context cannot be transferred to the functional context. The reason is that, while the proximal step in (BF) is also a gradient step $v_n = \mathrm{Prox}_{\gamma f} u_n = u_n - \gamma\nabla f_\gamma(u_n)$, the gradient step in (FB) is a proximal step $y_n = x_n - \gamma\nabla g(x_n) = \mathrm{Prox}_{\gamma g_{-\gamma}} x_n$ only if $\gamma \leq \beta$. Hence there is no functional analogue of the operator transform $B \to B_{-\gamma}$ for $\beta < \gamma < 2\beta$. An argument such as: *algorithm (FB) applied to (f,g) is algorithm (BF)*

applied to $(g_{-\gamma}, f_\gamma)$, which would allow to deduce Theorem 4.3 from Theorem 4.2, does not hold because we do not know how to define $g_{-\gamma}$ as a function for $\beta < \gamma < 2\beta$. So when it comes to the behavior of the values of f, g as the algorithms proceed, we have to give independent proofs.

4.5.2. *Properties of the limits.* Consider algorithm (FB) in the context of Theorem 4.2. While $x = \lim x_n = \lim z_n$ is a solution to a minimization problem, namely $\inf(f+g)$, the limit $y = \lim y_n$ does not seem to enjoy minimization properties when $\beta < \gamma < 2\beta$. We could have added to the assertions of Theorem 4.2 the following one, which is a straight consequence of Theorem 3.2 (v): $y \in \operatorname{zer}((\nabla g)_{-\gamma} + \nabla f_\gamma)$. This would have brought little information, since we are not able to express $(\nabla g)_{-\gamma}$, for $\beta < \gamma < 2\beta$, as the (possibly generalized) subdifferential of some real function defined on H. Yet, as pointed in Theorem 4.2 (v), y is a solution to $\nabla f_\gamma y + \nabla g(\operatorname{Prox}_{\gamma f} y) = 0$, a problem with Lipschitz regularity.

Of course, the above remark holds, *mutatis mutandis*, for algorithm (BF) in the context of Theorem 4.3.

4.5.3. *Symmetry regained.* For $0 < \gamma \leq \beta$ a complete symmetry is recovered between algorithms (FB) and (BF). Theorem 4.2 completed with Theorem 4.9 is the exact counterpart of Theorem 4.3 completed with Theorem 4.10. Parameter λ_n appears as a genuine overrelaxation parameter, since it may be chosen nearly as great as $\delta = 3/2$.

Each of algorithms (FB) and (BF) solves two minimization problems, namely: $\inf(f+g)$ and $\inf(g_{-\gamma} + f_\gamma)$. They have the same minimum value and are dual to each other *via* the involutive transformation $(f,g) \to (g_{-\gamma}, f_\gamma)$. But if the main concern is to solve $\inf(f+g)$, it is likely that problem $\inf(g_{-\gamma} + f_\gamma)$ cannot help much, if only because implementing f_γ and $g_{-\gamma}$ is not so easy (even with Lemma 4.5 (iii)).

4.5.4. *(FB) versus (BF).* Under Assumption 2, Theorems 4.2 and 4.3 determine the limits of the sequences $(x_n, y_n, z_n, \nabla g(x_n), \nabla f_\gamma(y_n), \nabla g(z_n))_{n\in\mathbf{N}}$ and $(u_n, v_n, w_n, \nabla f_\gamma(u_n), \nabla g(v_n), \nabla f_\gamma(w_n))_{n\in\mathbf{N}}$. Most of these results seem to be new.

Algorithm (BF) generates the sequence $(v_n)_{n\in\mathbf{N}}$ which minimizes $f+g$. If moreover, $0 \leq \lambda_n \leq 1$, the sequence $((f+g)(v_n))_{n\in\mathbf{N}}$ is nonincreasing.

Algorithm (FB) generates the sequence $(z_n)_{n\in\mathbf{N}}$ which minimizes $f+g$. But the sequence $((f+g)(z_n))_{n\in\mathbf{N}}$ is not proved to be nonincreasing under simple extra hypotheses. It is the sequence $(x_n)_{n\in\mathbf{N}}$ which is minimizing for $f+g$ and such that $((f+g)(x_n))_{n\in\mathbf{N}}$ is nonincreasing under the additional conditions $0 \leq \lambda_n \leq 1$, $x_0 \in \operatorname{dom} f$ and $\limsup \lambda_n > 0$.

This theoretical advantage of (BF) over (FB) was not made apparent by our numerical experiments, where all sequences $((f+g)(x_n))_{n\in\mathbf{N}}$, $((f+g)(z_n))_{n\in\mathbf{N}}$ and $((f+g)(v_n))_{n\in\mathbf{N}}$ behaved alike (see Section 5 below).

4.5.5. *Dispensing with convexity?* A natural question is whether or not some nonconvexity can be introduced in function f (obviously g must be convex). The simplest attempt is to suppose f convex up to a square: $f = h - \frac{\varepsilon}{2}\|.\|^2$ with $\varepsilon > 0$ and h convex, lower semicontinuous, proper. The problem is then to find a zero of $A + \nabla g$ with $A = \partial h - \varepsilon I$. If A is ever to meet Assumption 1, the least we can hope for is to find some $\gamma > 0$ such that A_γ is cocoercive. Suppose $\gamma\varepsilon < 1$, a mere computation gives $A_\gamma = \frac{1}{1-\gamma\varepsilon}(\partial h)_\lambda(\frac{\cdot}{1-\gamma\varepsilon}) - \varepsilon I$, where $\lambda = \gamma/(1-\gamma\varepsilon)$. Consider, as an example, the function $h : \mathbf{R} \to \mathbf{R}$ defined by $h(x) = \frac{1}{2}[(x-1)^+]^2 + \frac{1}{2}[(-x-1)^+]^2$ (with $(x)^+ = \max(0,x)$). Then, $(\partial h)_\lambda(x) = [(x-1)^+ + (-x-1)^+]/(1+\lambda)$. Hence for $|x| \leq 1$, we have $A_\gamma(x) = -\varepsilon x$, so A_γ cannot be cocoercive for whatever positive ε and γ.

4.5.6. *Novelty.* Points (i), (ii) and (iii) of Theorem 4.2 are known, while all the rest seem to be new. In turn, Theorems 4.3, 4.9, 4.10 are completely new. Moreover, results about the behavior of the function values $f(x_n)$, $g(x_n)$, $f(v_n)$, $g(v_n)$ and so on, seem unknown in the general case $\lambda_n \not\equiv 1$. Recall that, if $\lambda_n \equiv 1$, then (FB) and (BF) reduce to algorithm ISTA, in which case the objective function values decrease to the minimum at a sublinear rate: $(f+g)(x_n) - \min(f+g) = O(1/n)$ (see [14, 15]).

4.5.7. *Summing up.* To stress the symmetry between (FB) and (BF) we summarize below the algorithms with the main convergence results.

FB(f,g)	**BF**(f,g)
$y_n = x_n - \gamma\nabla g(x_n)$	$v_n = \mathrm{Prox}_{\gamma f} u_n$
$z_n = \mathrm{Prox}_{\gamma f} y_n$	$w_n = v_n - \gamma\nabla g(x_n)$
$x_{n+1} = x_n + \lambda(z_n - x_n)$	$u_{n+1} = u_n + \lambda(w_n - u_n)$
convergence results under Assumption 2	
$\|z_n - x_n\| \to 0$	$\|w_n - u_n\| \to 0$
$x_n \rightharpoonup x \in \mathrm{Argmin}(f+g)$	$u_n \rightharpoonup u \in \mathrm{Argmin}(g_{-\gamma}+f_\gamma)$ (if $\gamma \le \beta$)
$\nabla g(x_n) \to \nabla g(x)$	$\nabla f_\gamma(u_n) \to \nabla f_\gamma(u)$
$y_n \rightharpoonup y \in \mathrm{Argmin}(g_{-\gamma}+f_\gamma)$ (if $\gamma \le \beta$)	$v_n \rightharpoonup v \in \mathrm{Argmin}(f+g)$
$\nabla f_\gamma(y_n) \to \nabla f_\gamma(y)$	$\nabla g(v_n) \to \nabla g(v)$
$(f+g)(z_n) \to \min(f+g)$	$(f+g)(v_n) \to \min(f+g)$
in addition, if $\gamma \le \beta$	
$(g_{-\gamma}+f_\gamma)(y_n) \to \min(g_{-\gamma}+f_\gamma)$	$(g_{-\gamma}+f_\gamma)(u_n) \to \min(g_{-\gamma}+f_\gamma)$
$\min(f+g) = \min(g_{-\gamma}+f_\gamma)$	

5. A THEORETICAL / NUMERICAL ILLUSTRATION

We checked numerically the convergence of $(f+g)(x_n)$, $(f+g)(z_n)$, $(f+g)(v_n)$ towards a common limit for $\beta < \gamma \le 2\beta$, and the convergence of $(f_\gamma + g_{-\gamma})(y_n)$, $(f_\gamma + g_{-\gamma})(u_n)$, $(f_\gamma + g_{-\gamma})(w_n)$ towards the same limit (hopefully $\min\{f_\gamma + g_{-\gamma}\} = \min\{f+g\}$) for $0 < \gamma \le \beta$. We chose an example that allows computing f_γ and $g_{-\gamma}$.

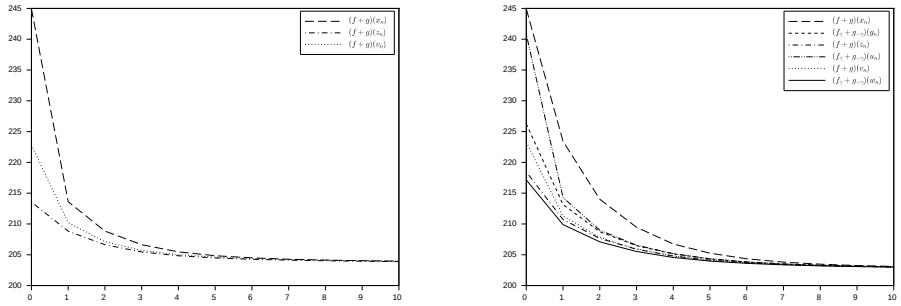

FIGURE 3. Values of $f+g$ and $f_\gamma + g_{-\gamma}$. Left: $(\gamma,\lambda) = (1.99, 1.0025)$. Right: $(\gamma,\lambda) = (0.99, 1.49)$.

Numerical computations were performed on the function

$$f : x \in \mathbf{R}^n \to \rho\|x\|_1, \qquad g : x \in \mathbf{R}^n \to \tfrac{1}{2}\|Ax-b\|_2^2,$$

where $n = 65536$, $\rho = 0.05$, $b \in \mathbf{R}^n$ and A is an ill-conditioned symmetric $n \times n$-matrix (actually b is the observed 256×256 image and A is the corruption operator taken from [10] – but ρ is different; see also [14, 15]). The spectral radius of A is 1, hence $\beta = 1$ and γ may vary in $]0,2[$.

Owing to the simple form of f and g, f_γ (for $\gamma > 0$) and $g_{-\gamma}$ (for $0 < \gamma < \beta$) are computable:
– $f_\gamma(x)$ is evaluated componentwise: $(f_\gamma(x))_i = (|x_i| - \rho\gamma)^+\mathrm{sign}(x_i)$, $1 \le i \le n$;
– with Lemma 4.5 (iii) we have $g_{-\gamma}(x) = \frac{1}{2}\langle R_\gamma\xi_\gamma, \xi_\gamma\rangle + \frac{1}{2}\|b\|_2^2 - \frac{1}{2\gamma}\|x\|_2^2$, with $R_\gamma = (\frac{I}{\gamma} - A^tA)^{-1}$ and $\xi_\gamma = \frac{x}{\gamma} - A^tb$ (computations involving R_γ are approximated by the conjugate gradient method).

For $\gamma = 1.99$ and $\lambda = 1.0025$ ($\delta \simeq 1.0025126$), the values of $(f+g)$ for the ten first iterates x_n, z_n given by (FB) and v_n given by (BF) are reported in Figure 3, Left.

For $\gamma = 0.99$ and $\lambda = 1.49$ ($\delta = 1.5$), the values of $(f+g)$ for the ten first iterates x_n, z_n given by (FB) and v_n given by (BF) and the values of $(f_\gamma + g_{-\gamma})$ for the first ten iterates y_n given by (FB) and u_n, w_n given by (BF) are reported in Figure 3, Right.

All nine sequences decrease and tend rapidly to be nearly identical. At the tenth iteration they differ by less than 0.5%.

As a rule, FB(f,g) and BF(f,g) have the same computational cost (the same gradient and proximal steps, but in different order) and - likely - the same performance. The solution of $\min\{f_\gamma + g_{-\gamma}\}$ appears as a by-product and is likely of purely theoretical interest. Algorithms (FB) and (BF) exposed in this paper enlarge the class of usual forward-backward algorithms.

6. FURTHER REMARKS

6.1. **(BF) as a limiting second-order system.** Let us now show how (BF) can be naturally introduced. First suppose that f is twice differentiable. Newton's method for finding a critical point of f

$$\nabla^2 f(v_n)(v_{n+1} - v_n) + \nabla f(v_n) = 0$$

can be viewed as a gradient method with variable metric induced by the Hessian. It can be generalized to finding a critical point of $f+g$, by using a gradient method with respect to the variable metric induced by the regularization $I/\gamma + \nabla^2 f(v_n)$ of the Hessian of f:

$$(I/\gamma + \nabla^2 f(v_n))(v_{n+1} - v_n) + \lambda_n \nabla(f+g)(v_n) = 0.$$

Using $\nabla^2 f(v_n)(v_{n+1} - v_n) \approx \nabla f(v_{n+1}) - \nabla f(v_n)$, we can approach the previous equation by

$$\frac{1}{\gamma}(v_{n+1} - v_n) + \nabla f(v_{n+1}) - \nabla f(v_n) + \lambda_n \nabla f(v_n) + \lambda_n \nabla g(v_n) = 0.$$

Generalization to a nonsmooth f is obtained by replacing the gradient of f by its subdifferential operator:

$$\begin{cases} w_n \in \partial f(v_n) \\ v_{n+1} + \gamma w_{n+1} - (v_n + \gamma w_n) + \lambda_n \gamma w_n + \lambda_n \gamma \nabla g(v_n) = 0. \end{cases}$$

Setting $u_n = v_n + \gamma w_n$, i.e., $v_n = \mathrm{Prox}_{\gamma f} u_n$, we obtain

$$u_{n+1} - u_n + \lambda_n \left(u_n - \mathrm{Prox}_{\gamma f} u_n + \gamma \nabla g(\mathrm{Prox}_{\gamma f} u_n)\right) = 0,$$

which is precisely (BF).

The discretization of the dynamic considered in [2] would also give (BF).

6.2. **Perspectives.** (i). The introduction of the backward-forward algorithms sheds new light on the class of forward-backward algorithms. Put together with other recent developments (such as inertial forward-backward algorithms, see [10]), this naturally suggests that there is a large class of these splitting algorithms. Our approach to the proof of convergence of these algorithms (Theorems 3.2, 3.4) is essentially based on the convergence of iterations for α-averaged operators. This class of operators has remarkable stability properties regarding the composition, and the convex combination (see [13]), which reinforces our hypothesis of the existence of this large class of algorithms of forward-backward type. A better understanding of this class of operators, could have important consequences. For example, a relaxed version of the forward-backward algorithm is not known for tame optimization (based on Kurdyka-Lojasiewicz property). It would be very interesting to know, if there is a form of these algorithms that fits well this general framework.

(ii). As we already stressed, our approach is related to the time discretization of a regularized Newton continuous dynamic with two potentials recently introduced in [2]:

$$\begin{cases} \upsilon(t) \in A(x(t)) \\ \lambda(t)\dot{x}(t) + \dot{\upsilon}(t) + \upsilon(t) + B(x(t)) = 0. \end{cases}$$

Formally, when $B = 0$, in the above system, the term $\lambda(t)\dot{x}(t)$ plays the role of a Levenberg-Marquardt regularization of the Newton dynamic $\dot{\upsilon}(t) + \upsilon(t) = 0$. In [2], $\lambda(\cdot)$ is asymptotically vanishing, which (when $B = 0$) makes the system asymptotically close to the Newton method for solving $A(x) \ni 0$. Our backward-forward algorithm is directly related to the discretization of this dynamic, in the case $\lambda(\cdot)$ constant. Thus our method is more related to Levenberg-Marquardt than to Newton method. Indeed, for further research, it would be interesting to study the backward-forward algorithms which are obtained by discretization of the regularized Newton continuous dynamic with two potentials, and $\lambda(\cdot)$ asymptotically vanishing. This would meet an active stream of current research concerning proximal based algorithms with large parameters (see [26, 28, 29, 30]).

(iii). The acceleration of the convergence of forward-backward algorithm is an important issue for applications. As described above, our approach relies on the use of a regularized Newton method in a non-smooth framework. This is an analysis of the second-order in space. Another approach, initiated by Nesterov [31, 32], is based on an analysis of the second-order in time, and gives rise to the so-called FISTA method [15]. It would be very interesting to know if the Nesterov approach fits well the backward-forward algorithm.

(iv) When coupling occurs in the constraint, a penalization technique reduces the problem to the situation without constraint. Specifically, the diagonal penalization method developed in [8] gives rise to a splitting process using at each stage, the basic blocks of the forward-backward method. Its complexity is comparable to the forward-backward method. For further research, it would be interesting to study the relaxed version of this method, and the convergence properties of the algorithm which is obtained by reversing the order of operations.

REFERENCES

[1] Abbas, B., Attouch, H.: Dynamical systems and forward-backward algorithms associated with the sum of a convex subdifferential and a monotone cocoercive operator, Optimization, Volume 64, Issue 10, October 2015, pages 2223-2252.

[2] Abbas, B., Attouch, H., Svaiter, B.F.: Newton-like dynamics and forward-backward methods for structured monotone inclusions in Hilbert spaces. J. Optim. Theory Appl. 161, 331-360 (2013).

[3] Alvarez, F., Peypouquet, J.: Asymptotic almost-equivalence of Lipschitz evolution systems in Banach spaces. Nonlinear Anal. 73, 3018-3033 (2010).

[4] Attouch, H., Bolte, J., Redont, P., Soubeyran, A.: Proximal alternating minimization and projection methods for nonconvex problems: an approach based on the Kurdyka-Lojasiewicz inequality. Math. Oper. Res. 35, 438–457 (2010).
[5] Attouch, H., Bolte, J., Svaiter, B.F.: Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward-backward splitting, and regularized Gauss-Seidel methods. Math. Program. 137, 91–129 (2013).
[6] Attouch, H., Briceño-Arias, L.M., Combettes, P.L.: A parallel splitting method for coupled monotone inclusions. SIAM J. Control Optim. 48, 3246–3270 (2010).
[7] Attouch, H., Briceño-Arias, L.M., Combettes, P.L.: A strongly convergent primal-dual method for nonoverlapping domain decomposition, Numerische Mathematik, published online 2015-07-10. arXiv:1410.4531v1 [math.NA]
[8] Attouch, H., Czarnecki, M.-O., Peypouquet, J.: Coupling forward-backward with penalty schemes and parallel splitting for constrained variational inequalities. SIAM J. Optim. 21, 1251-1274 (2011).
[9] Attouch, H., Maingé, P.-E., Redont, P.: A second-order differential system with Hessian-driven damping; application to non-elastic shock laws. Differ. Equ. Appl. 4, 27-65 (2012).
[10] Attouch, H., Peypouquet, J., Redont, P.: A dynamical approach to an inertial forward-backward algorithm for convex minimization. SIAM J. Optim. 24, 232–256 (2014).
[11] Attouch, H., Svaiter, B.F.: A continuous dynamical Newton-Like approach to solving monotone inclusions. SIAM J. Control Optim. 49, 574-598 (2011).
[12] Baillon, J.-B., Haddad, G.: Quelques proprits des oprateurs angle-borns et n-cycliquement monotones. Israel J. Math. 26, 137-150 (1977).
[13] Bauschke, H.H., Combettes, P.L.: Convex Analysis and Monotone Operator Theory in Hilbert Spaces. Springer, New York (2011).
[14] Beck, A., Teboulle, M.: A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems. SIAM J. Imaging Sci. 2, 183–202 (2009).
[15] Beck, A., Teboulle, M.: Gradient-based algorithms with applications in signal recovery problems. In: Palomar, D., Eldar, Y. (eds), Convex Optimization in Signal Processing and Communications, pp. 33–88. Camb. Univ. Press, Camb. (2010).
[16] Bolte, J., Sabach, S., Teboulle, M.: Proximal alternating linearized minimization for nonconvex and nonsmooth problems. Math. Program. A 146, 459-494 (2014).
[17] Chen, .G, Teboulle, M.: A proximal-based decomposition method for convex minimization problems. Math. Program. 64, 81-101 (1994).
[18] Combettes, P.L., Pesquet, J.-C.: Proximal splitting methods in signal processing. In: Bauschke, H.H., Burachik, R.S., Combettes, P.L., Elser, V., Luke, D.R., Wolkowicz, H. (eds), pp. 185-212. Springer, New York (2011).
[19] Combettes, P.L., Pennanen, T.: Proximal methods for cohypomonotone operators. SIAM J. Control Optim. 43, 731-742 (2004).
[20] Combettes, P.L., Wajs, V.R.: Signal recovery by proximal forward-backward splitting. Multiscale Model. Simul. 4, 1168-1200 (2005).
[21] Frankel, P., Garrigos, G., Peypouquet, J.: Splitting methods with variable metric for KL functions and general convergence rates. J. Optim. Theory Appl. 165, no. 3, 874–900 (2015).
[22] Goldstein, A.A.: Convex programming in Hilbert space. Bull. Am. Math. Soc. 70, 709-710 (1964).
[23] Iusem, A.N., Pennanen, T., Svaiter, B.F.: Inexact variants of the proximal point algorithm without monotonicity. SIAM J. Optim. 13, 1080-1097 (2003).
[24] Levitin, E.S., Polyak, B.T.: Constrained minimization methods. USSR Comput. Math. Math. Phys. 6, 1-50 (1966).
[25] Lions, P.L., Mercier, B.: Splitting algorithms for the sum of two nonlinear operators. SIAM J. Numer. Analysis 16, 964–979 (1979).
[26] Marques Alves, M., Monteiro, R.D.C., Svaiter, B.F.: Primal-dual regularized SQP and SQCQP type methods for convex programming and their complexity analysis. Optim. on Line. http://www.optimization-online.org/DB_HTML/2014/05/4353.html (2014).
[27] Minty, G.: Monotone (nonlinear) operators in Hilbert space. Duke Math. J. 29, 341-346 (1962).
[28] Monteiro, R.D.C., Sicre, M.R., Svaiter, B.F.: A hybrid proximal extragradient self-concordant primal barrier method for monotone variational inequalities, SIAM J. Optim. 25-4 (2015), pp. 1965-1996. (2013).
[29] Monteiro, R.D.C., Svaiter, B.F.: Iteration-Complexity of a Newton Proximal Extragradient Method for Monotone Variational Inequalities and Inclusion Problems. SIAM J. Optim. 22, 914-935 (2012).
[30] Monteiro, R.D.C., Svaiter, B.F.: Iteration-Complexity of Block-Decomposition Algorithms and the Alternating Direction Method of Multipliers. SIAM J. Optim. 23, 475-507 (2013).

[31] Nesterov, Y.E.: Introductory Lectures on Convex Optimization. Kluwer, Boston (2004).
[32] Nesterov, Y.E.: A method for solving the convex programming problem with convergence rate $O(1/k^2)$. Dokl. Akad. Nauk SSSR 269, 543-547 (1983).
[33] Noun, N., Peypouquet, J.: Forward-backward-penalty scheme for constrained convex minimization without inf-compactness. J. Optim. Theory Appl. 158, 787-795 (2013).
[34] Passty G.B.: Ergodic convergence to a zero of the sum of monotone operators in Hilbert space. J. Math. Analysis Appl. 72, 383-390 (1979).
[35] Peypouquet, J.: Convex optimization in normed spaces: theory, methods and examples. Springer, Cham (2015).
[36] Peypouquet, J., Sorin, S.: Evolution equations for maximal monotone operators: asymptotic analysis in continuous and discrete time. J. Convex Analysis 17, 1113-1163 (2010).
[37] Rockafellar, R.T.: On the maximal monotonicity of subdifferential mappings. Pac. J. Math. 33, 209-216 (1970).
[38] Rockafellar, R.T., Wets, R.J.-B.: Variational Analysis. Springer-Verlag, New York (2009).
[39] Solodov, M.V., Svaiter, B.F.: A hybrid approximate extragradient-proximal point algorithm using the enlargement of a maximal monotone operator. Set Valued Analysis 7, 323-345 (1999).
[40] Pennanen, T.: Local convergence of the proximal point algorithm and multiplier methods without monotonicity. Math. Oper. Res. 27, 170-191 (2002).
[41] Tseng, P.: A modified forward-backward splitting method for maximal monotone mappings. SIAM J. Control Optim. 38, 431-446 (2000).

HÉDY ATTOUCH. INSTITUT MONTPELLIÉRAIN ALEXANDER GROTHENDIECK, IMAG UMR 5149 CNRS, UNIVERSITÉ MONTPELLIER 2, PLACE EUGÈNE BATAILLON, 34095 MONTPELLIER CEDEX 5, FRANCE. hedy.attouch@univ-montp2.fr

JUAN PEYPOUQUET. DEPARTAMENTO DE MATEMÁTICA, UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA, AVENIDA ESPAÑA 1680, VALPARAÍSO, CHILE. juan.peypouquet@usm.cl

PATRICK REDONT. INSTITUT MONTPELLIÉRAIN ALEXANDRE GROTHENDIECK, IMAG UMR 5149 CNRS, UNIVERSITÉ MONTPELLIER 2, PLACE EUGÈNE BATAILLON, 34095 MONTPELLIER CEDEX 5, FRANCE. patrick.redont@univ-montp2.fr