



HAL
open science

Optimization with First Order Algorithms

Charles Dossal, Samuel Hurault, Nicolas Papadakis

► **To cite this version:**

Charles Dossal, Samuel Hurault, Nicolas Papadakis. Optimization with First Order Algorithms. 2024.
hal-04755998

HAL Id: hal-04755998

<https://hal.science/hal-04755998v1>

Preprint submitted on 28 Oct 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Optimization with First Order Algorithms

Charles Dossal¹, Samuel Hurault², and Nicolas Papadakis³

¹IMT, Univ. Toulouse, INSA Toulouse, Toulouse, France

²ENS Paris - CNRS

³Univ. Bordeaux, CNRS, INRIA, Bordeaux INP, IMB, UMR 5251,
F-33400 Talence, France

Contents

1	Introduction	3
2	Definitions and background	3
2.1	Existence of minimizers for convex functions	4
2.2	L -smoothness	5
3	Optimization of smooth convex functions	6
3.1	Properties of gradient of smooth functions	6
3.2	Gradient descent algorithm	9
3.2.1	Gradient descent operator	9
3.2.2	Gradient descent with fixed stepsize	10
3.2.3	Strongly convex function	11
3.3	Convergence rates of the gradient descent algorithm	12
3.4	Other gradient descent algorithms	14
3.4.1	Gradient descent with optimal stepsize	14
3.4.2	Newton's method	14
3.4.3	Gradient descent with backtracking	15
3.4.4	Implicit gradient descent with fixed stepsize	15
3.4.5	Projected gradient	16

4	Optimization of non smooth convex functions	17
4.1	Properties of subdifferentials	18
4.2	Proximal operator	20
4.2.1	Firm nonexpansiveness	22
4.2.2	Strongly convex functions	24
4.3	Proximal algorithms	25
4.3.1	Proximal Point algorithm	25
4.3.2	Forward-Backward algorithm (FB)	25
4.3.3	Fast Iterative Shrinkage Thresholding Algorithm (FISTA)	27
4.3.4	Douglas-Rachford algorithm (DR)	30
4.3.5	Parallel ProXimal Algorithm (PPXA)	34
4.3.6	Alternating Direction Method of Multipliers (ADMM)	35
5	Duality	37
5.1	Properties of convex conjugates	37
5.2	Primal-Dual Algorithms	40
5.2.1	Saddle-point problem	40
5.2.2	Chambolle-Pock algorithm (CP)	42
5.2.3	Condat algorithm	45
5.3	Equivalence between proximal splitting algorithms	46
5.3.1	Chambolle-Pock and Douglas-Rachford equivalence	46
5.3.2	ADMM and Douglas-Rachford equivalence	47
6	Extension to nonconvex optimization	49
6.1	From convex to nonconvex optimization, the case of gradient descent	49
6.2	Single-point nonconvex convergence of Forward-Backward	50
6.2.1	Nonconvex subdifferential and critical points	51
6.2.2	Kurdyka-Łojasiewicz (KL) property	52
6.2.3	Nonconvex Forward-Backward single-point convergence	54
6.3	Other proximal splitting algorithms	56
6.3.1	Nonconvex Douglas-Rachford and ADMM	56
6.3.2	Nonconvex primal version of the Chambolle-Pock algorithm	57
6.3.3	Nonlinear Chambolle-Pock algorithm	58
7	Various imaging problems, various optimization algorithms	58
A	Convergence of Chambolle-Pock algorithm	62

1 Introduction

These notes focus on the minimization of convex functionals using first-order optimization methods, which are fundamental in many areas of applied mathematics and engineering. The primary goal of this document is to introduce and analyze the most classical first-order optimization algorithms. We aim to provide readers with both a practical and theoretical understanding in how and why these algorithms converge to minimizers of convex functions.

The main algorithms covered in these notes include gradient descent, Forward-Backward splitting, Douglas-Rachford splitting, the Alternating Direction Method of Multipliers (ADMM), and Primal-Dual algorithms. All these algorithms fall into the class of first order methods, as they only involve gradients and subdifferentials, that are first order derivatives of the functions to optimize. For each method, we provide convergence theorems, with precise assumptions and conditions under which the convergence holds, accompanied by complete proofs.

Beyond convex optimization, the final part of this manuscript extends the analysis to nonconvex problems, where we discuss the convergence behavior of these same first-order methods under broader assumptions. To contextualize the theory, we also include a selection of practical examples illustrating how these algorithms are applied in different image processing problems.

We mention the books [4, 5] for more complete results and proofs, as well as the lecture notes [49, 44], the handbook [22] including stochastic aspects and the review paper [17] that contains many details on primal-dual algorithms.

Organization In section 2, we give the necessary background on convex functions, non expansive operators and smooth (differentiable) functions, that will be used all along the document. Section 3 focuses on the minimization of smooth convex functions with gradient descent schemes. Section 4 is dedicated to non smooth convex functions and proximal splitting algorithms. We introduce duality tools and primal dual algorithms in section 5. Section 6 explores the extension of previous algorithms to the optimization of nonconvex functionals. Finally, examples of the application of proximal splitting algorithms to imaging problems are presented in section 7.

2 Definitions and background

We recall that the objective of this document is to present algorithm computing minimizers of functions f . We first introduce the main notations and definitions in section 2.1 in order to give condition for the existence of minimizers of f . As we will consider the optimization of smooth and non smooth functions, we finally give definitions related to differentiable functions with Lipschitz gradient in section 2.2.

2.1 Existence of minimizers for convex functions

In the whole document, E is an Euclidean vectorial space of finite dimension, equipped with an inner product $\langle \cdot, \cdot \rangle$ and a norm $\|\cdot\| = \langle \cdot, \cdot \rangle^{\frac{1}{2}}$. We can typically consider that $E = \mathbb{R}^d$.

Definition 1 (Domain). *Let f be a function defined from E to $\bar{\mathbb{R}} = \mathbb{R} \cup +\infty$, we denote the domain of f as $\text{dom}(f) = \{x \in E \text{ such that } f(x) \neq +\infty\}$.*

Definition 2 (Coercivity). *A function f is coercive if $\lim_{\|x\| \rightarrow +\infty} f(x) = +\infty$.*

Definition 3 (Lower semi-continuity, l.s.c). *A function f defined from E to $\mathbb{R} \cup +\infty$ is lower semi-continuous (l.s.c) if, $\forall x \in E$, $\liminf_{y \rightarrow x} f(y) \geq f(x)$.*

Remark 1. *For later use, we recall interesting properties related to lower semi-continuous functions:*

- *Lower semi-continuity is stable by sum.*
- *Continuous functions are lower semi-continuous.*
- *The characteristic function $\iota_C(x)$ of a closed convex set C is lower semi-continuous.*
- *If f is l.s.c, for all $\alpha \in \mathbb{R}$, the sets*

$$\{x \in E, f(x) \leq \alpha\} \text{ and } \{(x, \alpha) \in E \times \mathbb{R} \text{ such that } f(x) \leq \alpha\}$$

are closed.

Definition 4 (Proper). *A function f from E to $\bar{\mathbb{R}} = \mathbb{R} \cup \{\pm\infty\}$ is proper if $\exists x \in E$ such that $f(x) < +\infty$ and $f(x) > -\infty$, $\forall x \in E$.*

Definition 5 (Convexity). *Let f be a function defined from E to $\bar{\mathbb{R}}$, f is convex iff for all pairs $(x, y) \in E \times E$ and $\forall \lambda \in [0, 1]$, $f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y)$. If the inequality is strict, then f is strictly convex.*

Proposition 1. *Let f be a convex, proper, l.s.c and coercive function defined on E with values in $\bar{\mathbb{R}}$, then f is bounded from below and admits at least one minimizer. If f is strictly convex then the minimizer is unique.*

Proof. For all $r \in \mathbb{R}$, the set $H_r = \{x \in E \text{ such that } f(x) \leq r\}$ is closed since f is l.s.c and bounded since f is coercive. Then for all $r \in \mathbb{R}$, the set H_r is compact. The set $H = \bigcap_{r \in \mathbb{R}} H_r$ is an intersection of nested compact sets. As f is proper, $H \neq \emptyset$ and there exists $r_0 \in \mathbb{R}$ such that H_{r_0} is empty. As a consequence, r_0 is a lower bound of f .

Next, as f is proper, there exists $x_0 \in E$ such that $f(x_0) < +\infty$ and $f(x_0) \neq -\infty$. The set $H = \{x \in E \text{ such that } f(x) \leq f(x_0)\}$ is closed since f is l.s.c and bounded as f is coercive. H is thus compact as the dimension of E is finite. We have

$\inf_{x \in E} f(x) = \inf_{x \in H} f(x) > -\infty$ since f is lower bounded. Let $(x_n)_{n \in \mathbb{N}}$ be a minimizing sequence of elements of H such that $\lim_{n \rightarrow \infty} f(x_n) = \inf_{x \in H} f(x)$. This sequence admits a subsequence converging to an element x^∞ of H . Since f is l.s.c, we get on a

$$\liminf_{n \rightarrow \infty} f(x_n) \geq f(x^\infty).$$

As $\lim_{n \rightarrow \infty} f(x_n) = \inf_{x \in H} f(x)$, we deduce that $f(x^\infty) = \inf_{x \in H} f(x) = \inf_{x \in E} f(x)$. The uniqueness of the minimizer finally comes with the strict convexity. \square

Using the notion of convexity, we can also define *strong and weak convexity*. Strong convexity allows better local control of the function to minimize.

Definition 6 (α -strong and weak convexity). *Let f be a function defined from E to $\mathbb{R} \cup \{+\infty\}$ and $\alpha > 0$. We say that the function f is α -strongly convex, or α -convex (resp. α -weakly convex) if the function g defined by $g(x) = f(x) - \frac{\alpha}{2}\|x\|^2$ (resp. $g(x) = f(x) + \frac{\alpha}{2}\|x\|^2$) is convex.*

By definition, if f is convex and y is an element of E , the function $x \mapsto f(x) + \frac{1}{2\gamma}\|x - y\|^2$ is $\frac{1}{\gamma}$ -strongly convex. Strong convexity involves strict convexity and ensures the uniqueness of the minimizer.

2.2 L -smoothness

In this document, we consider the minimization of smooth and non-smooth functions. We will refer to a smooth function as soon as it is differentiable.

Definition 7 (Differentiability). *A function f defined from E to $\mathbb{R} \cup +\infty$, is differentiable at point $x \in E$ if there exists a unique point $l_x \in E$ such that*

$$\lim_{h \in E, \|h\| \rightarrow 0} \frac{f(x+h) - f(x) - \langle l_x, h \rangle}{\|h\|} = 0.$$

The gradient of f at point x is then denoted as $\nabla f(x) = l_x$.

*A function is said differentiable as soon as it is differentiable $\forall x \in E$. A function f is called **non-smooth** when it is not a differentiable function.*

Definition 8 (Lipschitz continuity). *A function T defined from E to E is said to be L -Lipschitz (or L -Lipschitz continuous) if for all $(x, y) \in E^2$, we have*

$$\|T(x) - T(y)\| \leq L\|x - y\|.$$

Definition 9 (L -smoothness). *A smooth function f with L -Lipschitz gradient ∇f is called a L -smooth function.*

A classical theoretical framework ensuring L -smoothness is the case of twice differentiable functions f . If we are able to bound the operator norm of the Hessian matrix of f by L , then we deduce that the gradient of f is L -Lipschitz (see Remark 3 in the next section).

3 Optimization of smooth convex functions

We consider the general problem:

$$\min_{x \in E} f(x) \tag{1}$$

where f is a real valued, convex, coercive, L -smooth function defined on a closed, convex set E . As f is differentiable, it is lower semi-continuous and proper by definition. Following Proposition 1, these hypotheses ensure that the optimization problem (1) admits at least one solution. Additionally, if f is strictly convex, then the solution is unique.

The minimizers x^* of a convex, differentiable function f defined on E are simply characterized by the Euler equation:

$$\nabla f(x^*) = 0. \tag{2}$$

A different way to formulate this is that all the critical points of a convex function are global minima. To estimate such critical points, we will rely on the gradient descent algorithm:

$$x_{n+1} = x_n - \gamma \nabla f(x_n). \tag{3}$$

This algorithm consists in applying recursively from a point $x_0 \in E$ the gradient descent operator $\text{Id} - \gamma \nabla f$, defined for a strictly positive real number γ called *step-size*.

The organization of this section is as follows. We first give general properties on the gradient of differentiable convex functions f in section 3.1. Then we study in section 3.2 the properties of the gradient descent operator $\text{Id} - \gamma \nabla f$ and exploit this operator to find fixed points of (3) satisfying the optimality condition (2) and thus minimizing problem (1). We provide an analysis of the convergence rate of the gradient descent scheme in section 3.3 and finally discuss several variants and accelerations of this algorithm in section 3.4.

3.1 Properties of gradient of smooth functions

We here recall important properties related to differentiable and L -smooth functions.

Proposition 2. *Let f be a convex, differentiable function defined from E to \mathbb{R} . Then for all $(x, y) \in E^2$, we have*

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle \tag{4}$$

Proof. By convexity of f , we have for all $\lambda \in]0, 1[$,

$$f(x + \lambda(y - x)) \leq (1 - \lambda)f(x) + \lambda f(y).$$

which can be rewritten

$$\frac{f(x + \lambda(y - x)) - f(x)}{\lambda} \leq f(y) - f(x).$$

By letting λ go to zero we obtain

$$\langle \nabla f(x), y - x \rangle \leq f(y) - f(x)$$

□

By applying this proposition to two different points x and y , we deduce that the gradient of f is monotone.

Definition 10 (Monotonicity). *Let T be a function defined from E to E . We say that T is monotone if for all $(x, y) \in E^2$,*

$$\langle T(x) - T(y), x - y \rangle \geq 0.$$

Corollary 1. *Let f be a convex, differentiable function defined from E to \mathbb{R} . Then the gradient of f is monotone, for all $(x, y) \in E^2$:*

$$\langle \nabla f(x) - \nabla f(y), x - y \rangle \geq 0.$$

Proof. The result is given by summing the inequality in Proposition 2 taken in (x, y) and (y, x) . □

Remark 2. *Note that Corollary 1 is in fact an equivalence [4, Proposition 17.10]. As the monotonicity result is directly obtained by the inequality in Proposition 2, we get that, for $f : E \rightarrow \mathbb{R}$ differentiable, the following points are equivalent:*

- (i) f is convex.
- (ii) $\forall (x, y) \in E^2$, we have $f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle$
- (iii) ∇f is monotone.

Moreover, if f is twice-differentiable, this is also equivalent to [4, Proposition 17.10]

- (iv) for all $(x, y) \in E^2$, we have $\langle \nabla^2 f(x)y, y \rangle \geq 0$

We just saw that convex and differentiable functions are bounded from below by their affine approximations. In addition, if the gradient of f is L -Lipschitz, we can obtain an upper bound of f regardless of its convexity.

Lemma 1 (Descent Lemma). *Let f be a L -smooth function (i.e. differentiable with L -Lipschitz gradient) defined from E to E . Then for all $(y, z) \in E^2$, we have*

$$f(z) \leq f(y) + \langle \nabla f(y), z - y \rangle + \frac{L}{2} \|z - y\|^2. \quad (5)$$

Proof. Let g be a differentiable function on \mathbb{R} such that g' is K -Lipschitz (see Definition 8). Then we have

$$g(1) = g(0) + \int_0^1 g'(t) dt = g(0) + g'(0) + \int_0^1 (g'(t) - g'(0)) dt \leq g(0) + g'(0) + \frac{K}{2}.$$

Let $(y, z) \in E^2$. For all $t \in [0, 1]$, we set $v = z - y$, $y_t = y + t(z - y)$ and $g(t) = f(y_t)$. The function g is differentiable and $g'(t) = \langle \nabla f(y_t), v \rangle$. According to the hypotheses on f , g' is K -Lipschitz with $K = L\|v\|^2$.

From this we deduce that for all $(y, z) \in E^2$,

$$f(z) \leq f(y) + \langle \nabla f(y), z - y \rangle + \frac{L}{2} \|z - y\|^2.$$

□

Remark 3. Using Remark 2, we have that the descent lemma equation (5) is equivalent to $x \rightarrow \frac{L}{2}\|x\|^2 - f$ being convex. And if f is twice differentiable, this is also equivalent to: for all $(x, y) \in E^2$, $\langle \nabla^2 f(x)y, y \rangle \leq L$.

Next we recall a useful inequality for convex and L -smooth functions.

Lemma 2 (Co-coercivity). *Let f be a convex L -smooth function, then*

$$\forall (x, y) \in E^2, \quad \langle \nabla f(x) - \nabla f(y), x - y \rangle \geq \frac{1}{L} \|\nabla f(x) - \nabla f(y)\|^2. \quad (6)$$

This property is called co-coercivity of the function ∇f .

Proof. Using inequality (5) with $z = x - \frac{1}{L}\nabla f(x)$ and $y = x$, we obtain

$$\frac{1}{2L} \|\nabla f(x)\|^2 \leq f(x) - f\left(x - \frac{1}{L}\nabla f(x)\right) \leq f(x) - f(x^*).$$

where x^* is a minimizer of f . We also have, for all $x \in E$,

$$\frac{1}{2L} \|\nabla f(x)\|^2 \leq f(x) - f(x^*). \quad (7)$$

Let $(x, y) \in E^2$ and introduce the functions $h_1(z) = f(z) - \langle \nabla f(x), z \rangle$ and $h_2(z) = f(z) - \langle \nabla f(y), z \rangle$. These two functions are convex and respectively admit the minimizers x and y . Applying inequality (7) to these two functions and obtain

$$h_1(x) \leq h_1(y) - \frac{1}{2L} \|\nabla h_1(y)\|^2 \text{ and } h_2(y) \leq h_2(x) - \frac{1}{2L} \|\nabla h_2(x)\|^2.$$

By adding these two inequalities we obtain the result of the lemma. □

Let us finally mention that we can obtain a lower bound for f when the function f is also α -strongly (or weakly) convex,

Lemma 3. *Let $\alpha \in \mathbb{R}$, let f be a smooth and α -convex function defined from E to E . Then for all $(y, z) \in E^2$, we have*

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + \frac{\alpha}{2} \|y - x\|^2 \quad (8)$$

so that

$$\langle \nabla f(x) - \nabla f(y), y - x \rangle \geq \alpha \|y - x\|^2. \quad (9)$$

Proof. The function $f(x) - \frac{\alpha}{2}\|x\|^2$ is smooth and convex, from the Definition 6 of α -strong convexity. Using Proposition 2, we have for all $(x, y) \in E^2$

$$f(y) - \frac{\alpha}{2}\|y\|^2 \geq f(x) - \frac{\alpha}{2}\|x\|^2 + \langle \nabla f(x) - \alpha x, y - x \rangle, \quad (10)$$

which gives (8). Switching the roles of x and y in (8) and summing both relations, we obtain (9). □

3.2 Gradient descent algorithm

In this section we describe the most classical method using the gradient of a convex and differentiable function f , to approach a solution of problem (1):

$$\min_{x \in E} f(x). \quad (11)$$

3.2.1 Gradient descent operator

In order to compute minimizers of problem (11), we now introduce the gradient descent operator $\text{Id} - \gamma \nabla f$, defined for a strictly positive real number γ called *stepsize*. In this section, we analyze the properties of this operator using the characteristics of L -smooth functions studied in section 3.1. First of all, notice that the gradient descent operator allows to characterize minimizers of (1).

Proposition 3. *Let f be a convex, proper and differentiable function defined in E , then for all $\gamma \neq 0$:*

$$\text{Fix}(\text{Id} - \gamma \nabla f) = \text{argmin } f.$$

Proof. $x = (\text{Id} - \gamma \nabla f)(x) \Leftrightarrow \nabla f(x) = 0 \Leftrightarrow x \in \text{argmin } f.$ □

The idea behind gradient descent algorithms is to find fixed points of the gradient descent operator, as these fixed points correspond to minimizers of f . In that perspective, it is interesting to note that for a suitable choice of the stepsize γ , the gradient descent operator enjoys non expansive properties.

Definition 11 (Non expansive operators and contractions). *We recall that a function T defined from E to E is said to be L -Lipschitz (or L -Lipschitz continuous) if for all $(x, y) \in E^2$, we have*

$$\|T(x) - T(y)\| \leq L\|x - y\|.$$

If T is 1-Lipschitz we say that T is non expansive and if T is L -Lipschitz with $L < 1$ we say that T is L -contracting or just a contraction.

Proposition 4. *Let f be a convex, differentiable function with a L -Lipschitz gradient. For $\gamma > 0$ we note $T = \text{Id} - \gamma \nabla f$. If $\gamma \leq \frac{2}{L}$, the operator T is 1-Lipschitz.*

Proof. Taking $(x, y) \in E^2$ and using Lemma 2 on the co-coercivity of ∇f , we get

$$\begin{aligned} & \|x - \gamma \nabla f(x) - (y - \gamma \nabla f(y))\|^2 \\ &= \|x - y\|^2 + \gamma^2 \|\nabla f(x) - \nabla f(y)\|^2 - 2\gamma \langle x - y, \nabla f(x) - \nabla f(y) \rangle \\ &\leq \|x - y\|^2 + \gamma \left(\gamma - \frac{2}{L} \right) \|\nabla f(x) - \nabla f(y)\|^2. \end{aligned}$$

We deduce that $\text{Id} - \nabla f$ is 1-Lipschitz for $\gamma \leq \frac{2}{L}$. \square

3.2.2 Gradient descent with fixed stepsize

The gradient descent method with fixed stepsize consists in applying recursively the gradient descent operator $\text{Id} - \gamma \nabla f$. Considering a strictly positive real number γ called *stepsize* and an element $x_0 \in E$, the gradient descent algorithm writes $\forall n \in \mathbb{N}$:

$$x_{n+1} = x_n - \gamma \nabla f(x_n). \quad (12)$$

If we note T the operator defined from E to E by

$$Tx = x - \gamma \nabla f(x) = (\text{Id} - \gamma \nabla f)(x),$$

the sequence $(x_n)_{n \in \mathbb{N}}$ is simply defined by $x_{n+1} = Tx_n$.

If no specific hypothesis is made on the gradient of f , the method (12) may not converge to a minimizer of f . The algorithm may indeed oscillate and diverge if γ is chosen too large. On the other hand, if the gradient of f is L -Lipschitz, we now show that the algorithm (12) converges to a minimizer of f for a correct choice of γ .

Theorem 1 (Gradient descent algorithm). *Let f be a convex, differentiable function with a L -Lipschitz gradient, admitting a minimizer. If $\gamma < \frac{2}{L}$ then for all $x_0 \in E$, the sequence defined $\forall n \in \mathbb{N}$ by $x_{n+1} = (\text{Id} - \gamma \nabla f)x_n$ converges to a minimizer of f .*

There exist several proofs for this convergence result. The one we propose is based on the nonexpansiveness of the operator T and the following lemma. Such a sketch of proof will be useful to prove the convergence of other algorithms in this document.

Lemma 4. *Let T be a 1-Lipschitz operator defined from E to E admitting a fixed point. Let $x_0 \in E$ and let $(x_n)_{n \in \mathbb{N}}$ be the sequenced defined by $x_{n+1} = Tx_n$. If $\lim_{n \rightarrow \infty} \|x_{n+1} - x_n\| = 0$ then the sequence $(x_n)_{n \in \mathbb{N}}$ converges to a fixed point of T .*

Proof. Let y be a fixed point of T . As T is 1-Lipschitz,

$$\|x_{n+1} - y\| = \|T(x_n) - T(y)\| \leq \|x_n - y\|,$$

the sequence $(\|x_n - y\|)_{n \in \mathbb{N}}$ is decreasing and therefore bounded. The sequence $(x_n)_{n \in \mathbb{N}}$ is thus bounded. As E is of finite dimension, the sequence $(x_n)_{n \in \mathbb{N}}$ admits an adherent point (or closure point) $z \in E$. As $\lim_{n \rightarrow \infty} \|x_{n+1} - x_n\| = 0$, this adherent point satisfies $Tz = z$ and it is a fixed point of T . The sequence $\|z - x_n\|$ is thus decreasing and admits a subsequence that goes to zero. Hence the sequence converges to zero. \square

Proof of Theorem 1. We just have to show that the operator $T = \text{Id} - \gamma \nabla f$ satisfies all the hypotheses of Lemma 4. The fact that T is 1-Lipschitz is the result of Proposition 4. Furthermore, we may note from Proposition 3 that there is an equivalence between being a fixed point of T and being a minimizer of f , as $Tx = x$ is equivalent to $\nabla f(x) = 0$. Theorem 1 assumes the existence of such a minimizer and thus the existence of a fixed point of T . To finish the proof, it is sufficient to show that $\lim_{n \rightarrow \infty} \|x_{n+1} - x_n\| = 0$. This result is obtained by applying the inequality (5) satisfied by L -smooth functions (Lemma 1) to the points $y = x_n$ and $z = x_{n+1}$. We then have $z - y = \gamma \nabla f(x_n)$ and thus

$$f(x_{n+1}) \leq f(x_n) - \frac{1}{\gamma} \|x_{n+1} - x_n\|^2 + \frac{L}{2} \|x_{n+1} - x_n\|^2. \quad (13)$$

If $\gamma < \frac{2}{L}$, we get

$$f(x_{n+1}) + \left(\frac{2 - \gamma L}{2\gamma} \right) \|x_{n+1} - x_n\|^2 \leq f(x_n). \quad (14)$$

As the sequence $(f(x_n))_{n \in \mathbb{N}}$ is bounded from below, we deduce that $\lim_{n \rightarrow \infty} \|x_{n+1} - x_n\| = 0$, which concludes the proof of the theorem. \square

The gradient descent method with fixed stepsize is a *descent method*, i.e. the value of $f(x_n)$ decreases. In the next section, we will see that it is possible to control the rate at which $f(x_n) - f(x^*)$ decreases towards zero, where x^* refers to an arbitrary minimizer of f .

3.2.3 Strongly convex function

If the function f is L -smooth and α -strongly convex, the following result shows that the gradient descent operator is a contraction for $\tau < \frac{1}{L}$, so that Banach's Theorem directly ensures the convergence of the gradient descent method (12) to a fixed point of the gradient descent operator, which is a global minimizer of problem (11).

Proposition 5. *Let f be a α -strongly convex function. If f is L -smooth, then $\text{Id} - \gamma \nabla f$ is a $\sqrt{1 - \gamma\alpha}$ -Lipschitz application for $\tau < \frac{1}{L}$.*

Proof. We here rely on Lemmas 1 and 3. If f is α -strongly convex, then relation (8) states that $\forall x, y \in E^2$

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + \frac{\alpha}{2} \|y - x\|^2. \quad (15)$$

If f is L -smooth, we also have from relation (5) that

$$f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{L}{2} \|y - x\|^2,$$

so that $\alpha < L$. Combining the co-coercivity property (6) with the strong convexity proper (9), we get:

$$\begin{aligned}
& \|x - y\|^2 - \|(\text{Id} - \gamma \nabla f)(x) - (\text{Id} - \gamma \nabla f)(y)\|^2 \\
&= 2\gamma \langle \nabla f(x) - \nabla f(y), x - y \rangle - \gamma^2 \|\nabla f(x) - \nabla f(y)\|^2 \\
&= \gamma (\langle \nabla f(x) - \nabla f(y), x - y \rangle - \gamma \|\nabla f(x) - \nabla f(y)\|^2 + \langle \nabla f(x) - \nabla f(y), x - y \rangle) \quad (16) \\
&\geq \gamma \left(\frac{1}{L} - \gamma \right) \|\nabla f(x) - \nabla f(y)\|^2 + \gamma \alpha \|y - x\|^2.
\end{aligned}$$

We conclude that for all $\gamma < \frac{1}{L}$:

$$\|(\text{Id} - \gamma \nabla f)(x) - (\text{Id} - \gamma \nabla f)(y)\|^2 \leq (1 - \gamma \alpha) \|x - y\|^2,$$

so that $\text{Id} - \gamma \nabla f$ is a $\sqrt{1 - \gamma \alpha}$ -Lipschitz application. One can also check that since $\alpha < L$, it implies that $\gamma \alpha < 1$ for $\gamma < \frac{1}{L}$. \square

3.3 Convergence rates of the gradient descent algorithm

The decay rate on f of the gradient descent depends on the hypothesis made on f . If f is convex and differentiable, the decay is $O(\frac{1}{n})$. If f is strongly convex or even if f satisfies a quadratic growth condition, then the decay may be exponential. We give here two theorems based on the inequality (14) with $\gamma = \frac{1}{L}$ proved in section 3.2.1:

$$f(x_{n+1}) + \frac{L}{2} \|x_{n+1} - x_n\|^2 \leq f(x_n), \quad (17)$$

that is satisfied by the sequence defined by $x_{n+1} = x_n - \frac{1}{L} \nabla f(x_n)$. The first theorem provides a decay of $O(\frac{1}{n})$ for differentiable convex functions.

Theorem 2. *If f is convex differentiable and ∇f is L -Lipschitz, having at least one minimizer x^* . The sequence x_n defined by $x_0 \in E$ and*

$$x_{n+1} = x_n - \frac{1}{L} \nabla f(x_n) \quad (18)$$

satisfies

$$f(x_n) - f(x^*) \leq \frac{L}{2n} \|x_0 - x^*\|^2. \quad (19)$$

Proof. The proof uses a Lyapunov analysis, i.e. a sequence S_n is defined such that it is bounded along the trajectory. As it is usually the case for Lyapunov analysis, the sequence we define is non-negative and non-increasing:

$$S_n := n(f(x_n) - f(x^*)) + \frac{L}{2} \|x_n - x^*\|^2. \quad (20)$$

Notice that in practice, x^* and $f(x^*)$ may be unknown and the Lyapunov sequence can not be computed. We first show that there exists n_0 such that $(S_n)_{n \geq n_0}$ is non-increasing :

$$S_{n+1} - S_n = (n+1)(f(x_{n+1}) - f(x_n)) + f(x_n) - f(x^*) + \frac{L}{2} \|x_{n+1} - x^*\|^2 - \frac{L}{2} \|x_n - x^*\|^2.$$

Using (17) we get

$$S_{n+1} - S_n \leq -(n+1) \frac{L}{2} \|x_{n+1} - x_n\|^2 + f(x_n) - f(x^*) + \frac{L}{2} \langle x_{n+1} - x_n, x_{n+1} + x_n - 2x^* \rangle$$

and then

$$S_{n+1} - S_n \leq -n \frac{L}{2} \|x_{n+1} - x_n\|^2 + f(x_n) - f(x^*) + L \langle x_{n+1} - x_n, x_n - x^* \rangle.$$

Using the fact that $x_{n+1} - x_n = -\frac{1}{L} \nabla f(x_n)$, we get

$$S_{n+1} - S_n \leq -n \frac{L}{2} \|x_{n+1} - x_n\|^2 + f(x_n) - f(x^*) - \langle \nabla f(x_n), x_n - x^* \rangle.$$

Using the convexity of f we finally obtain that $S_{n+1} - S_n \leq 0$. Then we deduce that for all $n \in \mathbb{N}$, $S_n \leq S_0$ and thus

$$f(x_n) - f(x^*) \leq \frac{L}{2n} \|x_0 - x^*\|^2. \quad (21)$$

□

If additional assumptions, such as strong convexity, are made on f , a better convergence can be achieved. To prove this, we need the following lemma.

Lemma 5. *If f is α -strongly convex and differentiable, we have for all $x \in E$*

$$f(x) - f(x^*) \leq \frac{1}{2\alpha} \|\nabla f(x)\|^2, \quad (22)$$

where x^* is the unique minimizer of f .

Proof. For any $x \in E$, we define the function

$$\phi_x(y) := f(x) + \langle \nabla f(x), y - x \rangle + \frac{\alpha}{2} \|y - x\|^2.$$

Lemma 3 ensures that for any $(x, y) \in E^2$,

$$f(y) \geq \phi_x(y).$$

For any x , ϕ_x is a quadratic function whose minimizer is $x - \frac{1}{\alpha} \nabla f(x)$. Taking the minimum value in the above inequality, we get

$$f(x^*) \geq \phi_x(x - \frac{1}{\alpha} \nabla f(x)) = f(x) - \frac{1}{2\alpha} \|\nabla f(x)\|^2, \quad (23)$$

which ends the proof of the Lemma. □

With this lemma we can now state the following Theorem.

Theorem 3. *If f is a L -smooth and α -strongly convex function, then the sequence generated by the gradient descend with $\gamma = \frac{1}{L}$ satisfies*

$$f(x_n) - f(x^*) \leq \left(1 - \frac{\alpha}{L}\right)^n (f(x_0) - f(x^*)). \quad (24)$$

Proof. The inequality (17) can be written

$$f(x_{n+1}) - f(x^*) \leq f(x_n) - f(x^*) - \frac{1}{2L} \|\nabla f(x_n)\|^2. \quad (25)$$

Using Lemma 5 we have

$$f(x_{n+1}) - f(x^*) \leq f(x_n) - f(x^*) - \frac{\alpha}{L} (f(x_n) - f(x^*)). \quad (26)$$

which concludes the proof of the Theorem. \square

Notice that the strong convexity assumption does not improve the convergence rate already given by the so-called quadratic growth condition (17).

3.4 Other gradient descent algorithms

We now describe some variants and accelerations of the gradient descent algorithm.

3.4.1 Gradient descent with optimal stepsize

The gradient descent with optimal stepsize consists in adapting the descent stepsize γ at each iteration in order to have a maximal decrease of the value of the functional f . We define a sequence for $x_0 \in E$ and for all $n \in \mathbb{N}$:

$$x_{n+1} = x_n - \gamma_n \nabla f(x_n) \text{ with } \gamma_n = \arg \min_{\gamma > 0} f(x_n - \gamma \nabla f(x_n)) \quad (27)$$

This method converges to the unique minimizer of f if f is α -strongly convex. It has the advantage of converging faster than the gradient descent method with fixed stepsize in terms of number of iterations, but each iteration requires the resolution of an optimization problem that is not simple to solve in general. A close form expression of the optimal parameter γ is available in few cases, including quadratic functions f .

3.4.2 Newton's method

Newton's method was originally intended for approaching a regular zero of a function F of class C^2 defined from E to E . We say that x is a *regular zero* of F if

$$F(x) = 0 \text{ and } F'(x) \text{ is invertible.} \quad (28)$$

Newton's method is based on a first order Taylor development in the neighbourhood of a zero of F : $F(x_{n+1}) = F(x_n) + \langle F'(x_n), x_{n+1} - x_n \rangle$. In order to find x_{n+1} that is a zero of F , it builds, from an element $x_0 \in E$, a sequence defined for all $n \in \mathbb{N}$ by

$$x_{n+1} = x_n - (F'(x_n))^{-1} F(x_n). \quad (29)$$

In order for this sequence to be defined the matrix $F'(x_n)$ has to be invertible. This is guaranteed as soon as the departure point x_0 is close enough to the zero of F .

In the case of the minimization of a smooth function f

$$\min_x f(x), \tag{30}$$

the Newton's method corresponds to a gradient descent algorithm with an optimized (non scalar) step. Recalling that an optimum x^* of problem (30) is characterized by $\nabla f(x^*) = 0$, Newton's method can be applied to find a zero of ∇f . Assuming that f is \mathcal{C}^2 , the algorithm writes

$$x_{n+1} = x_n - H_f^{-1}(x_n)\nabla f(x_n),$$

where $H_f(x_n)$ is the Hessian matrix of f taken at point x_n . If f is convex, such a sequence is guaranteed to converge to a global minima of f .

3.4.3 Gradient descent with backtracking

We showed in Theorem 1 that the gradient descent algorithm $x_{n+1} = x_n - \gamma\nabla f(x_n)$ applied to the minimization of a convex L -smooth function f converges for $\gamma < 2/L$. This global upper bound may be suboptimal locally: there exists points x_n for which larger stepsizes γ can be safely considered. Moreover, the constant L may be unknown in real use cases. The idea of backtracking is to consider a gradient descent with a potentially too large stepsize, and to reduce it until a sufficient decrease condition is met. The decrease condition is built from relation (5) that writes, for $x = x_n$ and $y = x_n - \gamma\nabla f(x_n)$:

$$f(x_n) - f(x_n - \gamma\nabla f(x_n)) \geq \gamma \left(1 - \frac{\gamma L}{2}\right) \|\nabla f(x_n)\|^2. \tag{31}$$

Given a parameter a large stepsize $\gamma^0 > 2/L$ and a parameter $\beta \in]0; 1[$, backtracking consists in performing, at each point x_n a gradient descent with stepsize γ^0 and to update it with $\gamma^{\ell+1} = \gamma^\ell \beta$ until

$$f(x_n) - f(x_n - \gamma^\ell \nabla f(x_n)) > \frac{\gamma^\ell}{2} \|\nabla f(x_n)\|^2.$$

In the worst case, we obtain $\gamma^\ell < 2/L$ and backtracking gets back to standard gradient descent. This simple approach generally performs well. When the constant L is large, gradient descent relies on a small parameter $\gamma < 2/L$. With backtracking, the number of iterations n required by x_n to reach the neighborhood of a global minimizer x^* may be drastically reduced. Backtracking may also be considered to optimize smooth functions whose gradients are not Lipschitz. In this case, the choice of the initial point is essential.

3.4.4 Implicit gradient descent with fixed stepsize

The implicit gradient descent with fixed stepsize can be expressed as the explicit method, but it raises a practical problem of solving an implicit problem at each step.

We consider a strictly positive real number γ called *stepsize* and an element $x_0 \in E$. For all $n \in \mathbb{N}$ we define:

$$x_{n+1} = x_n - \gamma \nabla f(x_{n+1}). \quad (32)$$

Without hypothesis on f this relation does not guarantee the existence and uniqueness of x_{n+1} , and the resolution of this implicit equation can be practically complicated. The convexity of f ensures the existence and uniqueness of x_{n+1} , as we may note that x_{n+1} is the unique minimizer of a strongly convex and coercive function:

$$x_{n+1} = \arg \min_{x \in E} f(x) + \frac{1}{2} \|x_n - x\|^2 \quad (33)$$

Thus x_{n+1} is the unique element of E such that $x_{n+1} + \gamma \nabla f(x_{n+1}) = x_n$, which allows us to write $x_{n+1} = (\text{Id} + \gamma \nabla f)^{-1}(x_n)$, becoming $x_{n+1} = Tx_n$ by introducing $T = (\text{Id} + \gamma \nabla f)^{-1}$.

We will later see that this second definition of x_{n+1} s that does not take into account the gradient at all, can be generalized to any convex, proper, lower semi-continuous function without making any hypothesis of differentiability.

We will show that the implicit gradient descent method constructs a minimizing sequence, regardless of the choice of x_0 and $\gamma > 0$. This method has the advantage over the explicit method of not having a condition on γ , but the disadvantage of requiring the resolution of a possibly difficult problem at each iteration.

Theorem 4. *Let f be a convex, differentiable function, admitting a minimizer. Then for all $x_0 \in E$, the sequence defined for all $n \in \mathbb{N}$ by $x_{n+1} = (\text{Id} + \gamma \nabla f)^{-1}x_n$ converges to a minimizer of f .*

We do not prove this result as it is a particular case of more general algorithms, called Proximal Point and Forward-Backward algorithms, that will be treated in sections 4.3.1 and 4.3.2.

3.4.5 Projected gradient

We now consider the following constrained optimization problem:

$$\min_{x \in C} f(x), \quad (34)$$

where K is a closed, convex set and f is differentiable with an L -Lipschitz gradient. In this case, we characterize the minimizers in the following way.

Theorem 5. *Let f be a convex and smooth function defined from E to \mathbb{R} and $C \subset E$ a closed, convex set. Then x^* is a solution to (34) if and only if for all $y \in C$,*

$$\langle \nabla f(x^*), y - x^* \rangle \geq 0. \quad (35)$$

Proof. \Rightarrow Let $(x^*, y) \in C^2$. For all $t \in [0; 1]$, $x^* + t(y - x^*) \in C$ as C is convex. If x^* minimizes f on C , then

$$\begin{aligned} f(x^* + t(y - x^*)) - f(x^*) &\geq 0 \\ \lim_{t \rightarrow 0} \frac{f(x^* + t(y - x^*)) - f(x^*)}{t} &\geq 0 \\ \langle \nabla f(x^*), y - x^* \rangle &\geq 0. \end{aligned}$$

\Leftarrow From Proposition 2, we know that for a convex function

$$f(y) \geq f(x^*) + \langle \nabla f(x^*), y - x^* \rangle.$$

Assuming that $\langle \nabla f(x^*), y - x^* \rangle \geq 0$, we deduce that for all $y \in C$,

$$f(y) \geq f(x^*) + \langle \nabla f(x^*), y - x^* \rangle \geq f(x^*),$$

which implies that $x^* \in K$ solves problem (34). \square

The result below states that it is possible to adapt the gradient descent method with fixed step (12) to solve the constrained problem (34).

Theorem 6. *Let f be a convex, differentiable function such that ∇f is L -Lipschitz. Let K be a closed, convex set and $\gamma < \frac{2}{L}$. Then for all $x_0 \in E$, the sequence defined by*

$$x_{n+1} = \text{Proj}_K(x_n - \gamma \nabla f(x_n)) \quad (36)$$

converges to a solution of problem (34).

We do not prove this result for two reasons. First, it is a simple adaptation of the convergence proof for the gradient descent method with fixed step. Second, the projected gradient method is a particular case of the more general Forward-Backward algorithm, that will be treated in section 4.3.2.

4 Optimization of non smooth convex functions

In this part we consider non smooth convex optimization problems, i.e. the function that we want to minimize is non-differentiable. A particular case that we will consider is the case of the sum of two convex functions where at least one (let say g) is non-differentiable:

$$\min_{x \in E} f(x) + g(x). \quad (37)$$

The problem of minimizing a differentiable function f under convex constraints (34) can be rewritten in this form, by using the indicator function of the convex set K :

$$\min_{x \in C} f(x) \iff \min_{x \in E} f(x) + i_K(x), \quad (38)$$

where

$$i_K(x) = \begin{cases} 0 & \text{if } x \in K \\ +\infty & \text{otherwise.} \end{cases}$$

However, this is not the only situation we will consider. For example, in image processing, it is common to minimize functionals of the form of a sum of two terms; a *data fidelity* part and a *regularization* part, the latter being chosen to promote certain characteristics to the image to be reconstructed. Such a regularity prior is often non-differentiable, as in the case of an ℓ_1 regularization, that promotes sparsity, or the total variation regularization, that promotes piecewise constant images.

In this section, we will introduce *proximal splitting algorithms*, that are designed to solve problem (37). We first recall the key concept of subdifferentials ∂f of convex functions f in section 4.1. In section 4.2, we introduce and study the properties of the proximal operator $(\text{Id} + \gamma \partial f)^{-1}$, which can be seen as the implicit equivalent of the explicit gradient descent operator. In section 4.3, we finally present different proximal splitting algorithms to solve problem (37). Such algorithms minimize the sum of (non-differentiable) functions by alternating elementary explicit $(\text{Id} + \gamma \nabla f)$ and/or implicit $(\text{Id} + \partial f)^{-1}$ gradient operations on each function separately. In particular, we will analyze the Forward-Backward algorithm and its acceleration, the Alternating Direction Method of Multipliers and the Douglas-Rachford algorithm.

4.1 Properties of subdifferentials

Definition 12 (Subdifferential). *Let f be a function from E to $\mathbb{R} \cup \{+\infty\}$. The subdifferential of f is the multivalued operator that to each $x \in E$ associates the set of slopes*

$$\partial f(x) = \{u \in E \text{ such that } \forall y \in E, \langle y - x, u \rangle + f(x) \leq f(y)\}. \quad (39)$$

Remark 4. *We recall the following properties related to subdifferentials:*

- *An element $p \in \partial f$ of the subdifferential is called a **subgradient**.*
- *The subdifferential of a function at a point x can be either empty or a convex set, possibly reduced to a singleton.*
- *If f is convex and differentiable in each point $x \in E$, the subdifferential is reduced to the singleton $\{\nabla f(x)\}$, the gradient of f in x .*
- *The subdifferential of the function $x \mapsto -x^2$ is empty in each point of E , even though f is differentiable in each point. In this example f is not convex.*
- *It can be shown that in each point x of the relative interior of the domain of a **convex** function f , the subdifferential $\partial f(x)$ is non-empty.*
- *On the boundary of the domain things are a somewhat more complicated. The function f defined from \mathbb{R} to $\mathbb{R} \cup +\infty$ by*

$$f(x) = \begin{cases} -\sqrt{x} & \text{si } x \geq 0 \\ +\infty & \text{si } x < 0 \end{cases}$$

is convex, takes a finite value in 0 but does not admit a subdifferential in zero.

The importance of the subdifferential is made clear through Fermat's rule, as defined by the following theorem.

Theorem 7. *Let f be a proper function defined from E to $\mathbb{R} \cup +\infty$. Then*

$$\operatorname{argmin} f = \operatorname{zer} \partial f = \{x \in E \text{ such that } 0 \in \partial f(x)\}.$$

Proof.

$$x \in \operatorname{argmin} f \Leftrightarrow \forall y \in E, \langle y - x, 0 \rangle + f(x) \leq f(y) \Leftrightarrow 0 \in \partial f(x).$$

□

In order to minimize a functional we thus have to find a zero of its subdifferential. If by lower semi-continuity and coercivity, we know that a functional admits at least one minimum, then we know that in this minimum, the subdifferential is non-empty and contains 0. This rule is an extension of the well known fact that in the global minimum of a differentiable function, the gradient is zero.

We end this paragraph by some rules on the sum of subdifferentials.

Lemma 6 (Proof in Rockafellar [39]). *Let $(f_i)_{i \leq p}$ be a family of proper, lower semi-continuous functions and let $x \in E$. Then*

$$\partial\left(\sum_{i=1}^p f_i\right)(x) \supset \sum_{i=1}^p \partial f_i(x).$$

Additionally, if

$$\bigcap_{1 \leq i \leq p} \operatorname{inter}(\operatorname{dom}(f_i)) \neq \emptyset$$

then we have equality between the subdifferential of the sum and the sum of subdifferentials.

The hypothesis ensuring equality is not very constraining. In the majority of interesting practical cases, it is satisfied.

Remark 5. *If J is the sum of a convex differentiable function f and a convex function g , then under the hypotheses of Lemma 6,*

$$\partial J = \nabla f + \partial g.$$

Remark 6. *The constrained optimization problem (34) can be reformulated as*

$$\min_x f(x) + \iota_C(x),$$

where f is a smooth and convex function and ι_C is the characteristic function of a close convex set C , that is to say $\iota_C(x) = 0$ if $x \in C$ and $+\infty$ otherwise. The optimality condition thus gives

$$0 \in \partial(f + \iota_C)(x^*) \Leftrightarrow -\nabla f(x^*) \in \partial \iota_C(x^*).$$

We deduce from the subdifferential definition (39) that $\langle \nabla f(x^), x - x^* \rangle \geq 0 \forall x \in C$, which corresponds to the characterization of minimizers shown in Theorem 5.*

4.2 Proximal operator

In order to minimize non smooth convex functions, we now define the proximal (or proximity) operator of a convex function.

Definition 13 (Proximal operator). *Let g be a convex, lower semi-continuous and proper function from E to $\mathbb{R} \cup +\infty$. The proximity operator of g , denoted prox_g and also called "prox of g " is the operator defined from E to E by*

$$\text{prox}_g(x) = \arg \min_{z \in E} g(z) + \frac{1}{2} \|x - z\|_2^2 \quad (40)$$

Remark 7. *The function $z \mapsto g(z) + \frac{1}{2} \|x - z\|_2^2$ that defines the prox is convex, proper, lower semi-continuous and coercive, and thus admits at least one minimizer. The uniqueness of the minimizer comes from the strict convexity of the function $y \mapsto \frac{1}{2} \|x - z\|^2$.*

The proximal operator can be interpreted as a generalization of the concept of projecting onto a convex set in the sense that if f is the indicator of a closed convex set C then $\text{prox}_f(x)$ is the projection of x onto C . From now on we will say that **a function f is simple when its proximity operator can be computed easily.**

If f is not convex, the proximity operator may not be defined or it may be multi-valued. The convexity of f is a sufficient condition for the existence and uniqueness of the proximity operator, but it is not a necessary condition.

Example 1. *We here present the proximal operator of functions used in many problems coming from statistics and image processing.*

- *Separable functions. If a function f is separable, the proximity operator may be computed by blocks or even component by component. Hence if $x = (x_1, x_2)$ and if $f(x) = f_1(x_1) + f_2(x_2)$ then*

$$\begin{aligned} \text{prox}_f(x) &= \arg \min_{z=(z_1, z_2) \in \mathbb{R}^n} f_1(z_1) + f_2(z_2) + \frac{1}{2} \|z - x\|_2^2 \\ &= \arg \min_{z=(z_1, z_2) \in \mathbb{R}^n} f_1(z_1) + \frac{1}{2} \|z_1 - x_1\|_2^2 + f_2(z_2) + \frac{1}{2} \|z_2 - x_2\|_2^2 \end{aligned}$$

it follows that

$$\text{prox}_f(x) = (\text{prox}_{f_1}(x_1), \text{prox}_{f_2}(x_2))$$

This remark also applies to a sum of M functions $f(x) = \sum_{i=1}^M f_i(x_i)$, the proximity operator of f at the point x is a vector whose components p_i are $\text{prox}_{f_i}(x_i)$. A classical application of this rule is the proximal operator of $f(x) = \lambda \|x\|_1$ which is given by the component-wise "soft thresholding" operation with threshold λ :

$$\text{prox}_f(x) = \begin{cases} 0 & \text{if } |x| \leq \lambda \\ x - \lambda & \text{if } x > \lambda \\ x + \lambda & \text{if } x < -\lambda \end{cases}$$

- *Quadratic functions.* Standard data fidelity terms in imaging write $f(x) = \frac{\lambda}{2} \|Ax - b\|_2^2$. There is an explicit form for the proximal operator of such functions:

$$\text{prox}_f(x) = (\text{Id} + \lambda A^* A)^{-1}(x + \lambda A^* b). \quad (41)$$

Thus the computation of this proximity operator amounts to invert of a system of linear equations. In any cases the matrix $\text{Id} + \lambda A^* A$ is real and symmetric and a conjugate gradient may be used to solve such a problem. It can also happen that the matrix A is not available, and we may only be able to apply A and A^* . It also exist situations in imaging where the linear problem can be solved exactly:

- If A is a circular convolution by a filter h . In the Fourier domain $(\text{Id} + \lambda A^* A)^{-1}$ is a simple multiplication by $(1 + \lambda |\hat{h}(k)|^2)^{-1}$.
 - If A is a derivative operator like a discrete gradient, then if the derivative is circular, that may create strange boundary effect, it actually is a convolution and one may apply the previous formula. If the derivative is not shift invariant, the inversion in the Fourier domain is only an approximation and a conjugate gradient may provide more precise results.
 - If A is a masking operator the operator $(\text{Id} + \lambda A^* A)$ is diagonal in the domain and the inversion of the system is a simple pointwise division.
- *Indicator functions of vectorial subspaces* Let A a linear operator defined from \mathbb{R}^n to \mathbb{R}^d and $K = \{(x_1, x_2) \text{ such that } Ax_1 = x_2\}$ a vectorial subspace of \mathbb{R}^{n+d} . The proximity operator of the indicator function ι_K of K is defined by

$$(p_1, p_2) = \text{prox}_{\iota_K}(x_1, x_2) = \arg \min_{(z_1, z_2)} \frac{1}{2} \|z_1 - x_1\|_2^2 + \frac{1}{2} \|z_2 - x_2\|_2^2 + \iota_K(z_1, z_2) \quad (42)$$

It follows that $p_2 = Ap_1$ with

$$p_1 = \arg \min_{z_1} \frac{1}{2} \|z_1 - x_1\|_2^2 + \frac{1}{2} \|Az_1 - x_2\|_2^2 = (\text{Id} + A^* A)^{-1}(x_1 + A^* x_2) \quad (43)$$

Once again, the computation of this proximity operator may be fast or not depending on A . For example if $A = \text{Id}$ then $p_1 = \frac{1}{2}(x_1 + x_2)$ and $p_2 = p_1$. If we are able to diagonalize the matrix $I + A^* A$, for circular convolutions or circular derivative operators, the computation may be fast. If not, one needs to use an algorithm to solve this linear system, like conjugate gradient.

The proximal operator has many remarkable properties that make it useful when minimizing non-differentiable convex functions.

Proposition 6. *Let f be a convex function on E . Then for all $(x, p) \in E^2$, we have*

$$p = \text{prox}_f(x) \Leftrightarrow \forall y \in E, \langle y - p, x - p \rangle + f(p) \leq f(y). \quad (44)$$

Thus $p = \text{prox}_f(x)$ is the unique vector $p \in E$ such that $x - p \in \partial f(p)$.

This last characterization implies that the decomposition of $x = p + z$, as the sum of one element p from E and one element from the subdifferential of f at the point p , is unique. This property is the origin of the notation

$$\text{prox}_f(x) = (\text{Id} + \partial f)^{-1}(x)$$

to which we can give a sense even though the map ∂f is multivalued.

Proof. \Rightarrow Suppose that $p = \text{prox}_f(x)$. Let $\alpha \in]0, 1[$ and $p_\alpha = \alpha y + (1 - \alpha)p$. Then, by using the definition of $\text{prox}_f(x)$ and the convexity of f , we obtain

$$\begin{aligned} f(p) &\leq f(p_\alpha) + \frac{1}{2}\|x - p_\alpha\|^2 - \frac{1}{2}\|x - p\|^2 \\ &\leq \alpha f(y) + (1 - \alpha)f(p) - \alpha\langle y - p, x - p \rangle + \frac{\alpha^2}{2}\|y - p\|^2 \end{aligned}$$

and thus

$$\langle y - p, x - p \rangle + f(p) \leq f(y) + \frac{\alpha}{2}\|y - p\|^2.$$

By letting α go to zero we obtain the direct implication.

\Leftarrow Conversely, suppose that $\forall y$ we have $\langle y - p, x - p \rangle + f(p) \leq f(y)$. Then

$$\begin{aligned} f(p) + \frac{1}{2}\|x - p\|^2 &\leq f(y) + \frac{1}{2}\|x - p\|^2 + \langle x - p, p - y \rangle + \frac{1}{2}\|p - y\|^2 \\ &\leq f(y) + \frac{1}{2}\|x - y\|^2 \end{aligned}$$

from which we deduce that $p = \text{prox}_f(x)$ by using the definition of $\text{prox}_f(x)$. \square

An important property of fixed points of the proximity operator is the following:

Proposition 7. *Let f be a proper, convex function defined on E . Then*

$$\text{Fix prox}_f = \text{argmin } f.$$

Proof.

$$x = \text{prox}_f(x) \Leftrightarrow \forall y \in E, \langle y - x, x - x \rangle + f(x) \leq f(y) \Leftrightarrow x \in \text{argmin } f.$$

\square

Thanks to this characterization, one can consider an iterative algorithm involving the proximity operator of f to minimize a convex functional f .

4.2.1 Firm nonexpansiveness

To establish the convergence of proximal algorithms, we first show that the proximity operator of convex functions is 1-Lipschitz. To do so, we rely on the notion of firm nonexpansiveness, which implies 1-Lipschitzity.

Definition 14 (Firm nonexpansiveness). A function T is said to be firmly nonexpansive if for all $(x, y) \in E^2$,

$$\|T(x) - T(y)\|^2 + \|x - T(x) - y + T(y)\|^2 \leq \|x - y\|^2.$$

A classic example of a firmly nonexpansive function is the case of a projection onto a closed convex set. nonexpansiveness will be a key ingredient to show the convergence of the Douglas-Rachford algorithm.

Proposition 8. If f is a proper, convex function on E then the maps prox_f and $\text{Id} - \text{prox}_f$ are firmly nonexpansive; that is, for all $(x, y) \in E^2$,

$$\|\text{prox}_f(x) - \text{prox}_f(y)\|^2 + \|(x - \text{prox}_f(x)) - (y - \text{prox}_f(y))\|^2 \leq \|x - y\|^2$$

In particular, these two operators are nonexpansive (1-Lipschitz).

Proof. Let $(x, y) \in E^2$. By noting $p = \text{prox}_f(x)$ and $q = \text{prox}_f(y)$, we have, by definition of p and q :

$$\langle q - p, x - p \rangle + f(p) \leq f(q) \text{ and } \langle p - q, y - q \rangle + f(q) \leq f(p).$$

Adding these two inequalities gives

$$0 \leq \langle p - q, (x - p) - (y - q) \rangle$$

and we conclude by observing that

$$\begin{aligned} \|x - y\|^2 &= \|p - q + (x - p) - (y - q)\|^2 \\ &= \|p - q\|^2 + \|(x - p) - (y - q)\|^2 + 2\langle p - q, (x - p) - (y - q) \rangle. \end{aligned}$$

□

We underline that the gradient descent operator of a smooth convex function, which is 1-Lipschitz for $\gamma < \frac{2}{L}$ (see Proposition 4), is also nonexpansive with an additional restriction on the stepsize γ .

Proposition 9. Let f be a differentiable, convex function defined on E with an L -Lipschitz gradient. If $\gamma < \frac{1}{L}$, then the map $\text{Id} - \gamma \nabla f$ is firmly nonexpansive.

Proof. The proof is based on determining the sign of a scalar product. Let $(x, y) \in E^2$. We set $u = \gamma(\nabla f(x) - \nabla f(y))$ and $v = x - \gamma \nabla f(x) - (y - \gamma \nabla f(y))$. Hence we have

$$\|x - y\|^2 = \|u + v\|^2 = \|u\|^2 + \|v\|^2 + 2\langle u, v \rangle.$$

Showing that $\text{Id} - \gamma \nabla f$ is firmly nonexpansive amounts to demonstrate that $\langle u, v \rangle$ is non negative. The result directly comes from the co-coercivity of ∇f shown in Lemma 2 and the fact that $\gamma < \frac{1}{L}$:

$$\begin{aligned} \langle u, v \rangle &= \gamma \langle x - y, \nabla f(x) - \nabla f(y) \rangle - \gamma^2 \|\nabla f(x) - \nabla f(y)\|^2 \\ &\geq \gamma \left(\frac{1}{L} - \gamma \right) \|\nabla f(x) - \nabla f(y)\|^2. \end{aligned}$$

□

4.2.2 Strongly convex functions

The proximal operator of a strongly convex function is a contraction.

Proposition 10. *Let f be a α -strongly convex function, then $\text{prox}_{\gamma f}$ is a $\frac{1}{1+\alpha\gamma}$ -lipschitz application.*

Proof. We recall that a α -strongly convex function f satisfies for all x, y and $\alpha \in [0; 1]$:

$$f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha)f(y) - \frac{\alpha(1 - \alpha)\alpha}{2} \|x - y\|^2.$$

Let $p = \text{prox}_{\gamma f}(x)$, $\alpha \in (0, 1)$ and denote $p_\alpha = \alpha z + (1 - \alpha)p$ for some $z \in \mathbb{R}^n$. From the definition (40) of $\text{prox}_f(x)$ and the α -strong convexity of f , we have:

$$\begin{aligned} & f(p) \\ & \leq f(p_\alpha) + \frac{1}{2\gamma} \|x - p_\alpha\|^2 - \frac{1}{2\gamma} \|x - p\|^2 \\ & \leq \alpha f(z) + (1 - \alpha)f(p - \frac{\alpha(1 - \alpha)\alpha}{2} \|z - p\|^2 - \frac{\alpha}{\gamma} \langle z - p, x - p \rangle + \frac{\alpha^2}{2\gamma} \|z - p\|^2, \end{aligned}$$

and thus

$$\frac{1}{\gamma} \langle z - p, x - p \rangle + f(p) \leq f(z) + \frac{1}{2} \left(\frac{\alpha}{\gamma} - (1 - \alpha)\alpha \right) \|z - p\|^2.$$

For $\alpha \rightarrow 0$, we deduce that

$$\frac{1}{\gamma} \langle z - p, x - p \rangle + f(p) \leq f(z) - \frac{\alpha}{2} \|z - p\|^2.$$

Taking $z = q := \text{prox}_{\gamma f}(y)$ we get:

$$\frac{1}{\gamma} \langle q - p, x - p \rangle + f(p) \leq f(q) - \frac{\alpha}{2} \|q - p\|^2$$

Switching the roles of x and y and summing both expressions we obtain:

$$\langle p - q, (x - p) - (y - q) \rangle \geq \gamma\alpha \|q - p\|^2, \quad (45)$$

and we get the co-coercivity property :

$$\langle p - q, x - y \rangle \geq (1 + \gamma\alpha) \|p - q\|^2 \quad (46)$$

From (46), and recalling that $p = \text{prox}_f(x)$ and $q = \text{prox}_f(y)$, we get that $\text{prox}_{\gamma f}$ is $\frac{1}{1+\gamma\alpha}$ -Lipschitz and thus a contraction. \square

4.3 Proximal algorithms

We now present proximal optimization algorithms to solve nonsmooth convex minimization problems.

4.3.1 Proximal Point algorithm

In order to solve the problem

$$\min_{x \in E} g(x),$$

for a (nonsmooth) proper convex function g , the proximal point algorithm applies recursively the proximity operator of the function γg . Choosing $\gamma > 0$ and $x_0 \in E$, this algorithm defines the sequence $(x_n)_{n \in \mathbb{N}}$ as

$$x_{n+1} = Tx_n = \text{prox}_{\gamma g}(x_n). \quad (47)$$

The fixed points of the proximity operator are minimizers of g (Proposition 7) and this operator is 1-Lipschitz (Proposition 8). By construction, (47) gives that

$$g(x_{n+1}) + \frac{1}{2\gamma} \|x_n - x_{n+1}\|^2 \leq g(x_n),$$

which implies that the sequence $\|x_n - x_{n+1}\|^2$ goes to zero. Applying Lemma 4, we get that the sequence generated by the algorithm (47) converges to a minimizer of g .

In practice, this algorithm is barely used, in the sense that computing the proximity operator may be just as difficult as minimizing g directly (by taking $\gamma \rightarrow 0$). In what follows, we will rather use the proximity operator as an ingredient in more sophisticated *splitting* algorithms.

4.3.2 Forward-Backward algorithm (FB)

The Forward-Backward, also known as proximal Gradient Descent (PGD) is an algorithm designed to solve the following optimization problem:

$$\min_{x \in E} \mathcal{J}(x) = \min_{x \in E} f(x) + g(x) \quad (48)$$

where f is a convex, differentiable function with an L -Lipschitz gradient and g is a convex function. This algorithm consists in alternating an explicit gradient descent step on f and a proximity operator on g .

The Forward-Backward algorithm is defined by an initial point $x_0 \in E$ and a parameter $\gamma > 0$ as follows:

$$\forall n \in \mathbb{N} \quad x_{n+1} = Tx_n = \text{prox}_{\gamma g}(x_n - \gamma \nabla f(x_n)) \quad (49)$$

The algorithm results from the following proposition:

Proposition 11. *Let $\mathcal{J} = f + g$ be a functional defined from E to $\mathbb{R} \cup +\infty$ having the form of a sum of two convex, proper, lower semi-continuous functions satisfying the hypotheses of Lemma 6 (on the sum of subdifferentials), such that f is differentiable. Let $\gamma > 0$, then*

$$\text{zeros}(\partial \mathcal{J}) = \text{Fix}(\text{prox}_{\gamma g}(\text{Id} - \gamma \nabla f)). \quad (50)$$

Proof.

$$\begin{aligned}
0 \in \partial \mathcal{J}(x) &\Leftrightarrow 0 \in \partial \gamma \mathcal{J}(x) \\
&\Leftrightarrow 0 \in \nabla \gamma f(x) + \partial \gamma g(x) \\
&\Leftrightarrow -\gamma \nabla f(x) \in \partial \gamma g(x) \\
&\Leftrightarrow x - \gamma \nabla f(x) \in (\text{Id} + \partial \gamma g)(x) \\
&\Leftrightarrow x = \text{prox}_{\gamma g}(x - \gamma \nabla f(x)).
\end{aligned}$$

□

Having an operator T whose fixed points are the minimizers of \mathcal{J} does not guarantee that the sequence $(x_n)_{n \in \mathbb{N}}$ converges to one of these minimizers if T is not a contraction (take for instance $x_{n+1} = -x_n$ as a counterexample).

For convex functions f and g , the Forward-Backward operator is only nonexpansive, and thus not a contraction, as the composition of 1-Lipschitz operators. We indeed know from Propositions 4 and 8 that $\text{Id} - \tau \nabla f$ is 1-Lipschitz for $\tau < \frac{2}{L}$ and $\text{prox}_{\tau g}$ is 1-Lipschitz for all $\tau > 0$.

In the strongly convex case, convergence may directly be ensured. If f (resp. g) is strongly convex, then Proposition 5 (resp. Proposition 10) states that $\text{Id} - \tau \nabla f$ (resp. $\text{prox}_{\tau g}$) is a contraction for $\tau < \frac{1}{L}$ (resp. $\tau > 0$). Combining a contraction with a 1-Lipschitz operator preserves the contraction. Hence Banach's Theorem directly gives the convergence of the sequence generated by the Forward-Backward algorithm in case f or g are strongly convex.

To show convergence of the iterates in the general case, we introduce the concept of *surrogate functions*. The following lemma shows that at each iteration of the algorithm, the value of the functional \mathcal{J} decreases. Additionally, this decrease ensures that the sequence with the general term $\|x_{n+1} - x_n\|^2$ is summable and thus goes to zero when n goes to $+\infty$.

Lemma 7. *The sequence $(x_n)_{n \in \mathbb{N}}$ defined by $x_{n+1} = \text{prox}_{\gamma g}(x_n - \gamma \nabla f(x_n))$ satisfies the following relation:*

$$\mathcal{J}(x_{n+1}) + \left(\frac{1}{\gamma} - \frac{L}{2} \right) \|x_{n+1} - x_n\|^2 \leq \mathcal{J}(x_n). \quad (51)$$

We underline that relation (51), ensured by the Forward-Backward algorithm when minimizing $\mathcal{J} = f + g$, is the extension of the property (14) satisfied by the iterates of the gradient descent algorithm when minimizing a differentiable function f .

Proof. By definition of x_{n+1} , we have

$$x_{n+1} = \underset{x \in E}{\text{argmin}} \gamma g(x) + \frac{1}{2} \|x - x_n + \gamma \nabla f(x_n)\|^2.$$

Note that x_{n+1} is also the minimizer of the following functional:

$$x_{n+1} = \underset{x \in E}{\text{argmin}} g(x) + f(x_n) + \langle \nabla f(x_n), x - x_n \rangle + \frac{1}{2\gamma} \|x - x_n\|^2. \quad (52)$$

According to the inequality (5), we have, for all $x \in E$,

$$f(x) \leq f(x_n) + \langle \nabla f(x_n), x - x_n \rangle + \frac{L}{2} \|x - x_n\|^2. \quad (53)$$

The right hand-side is a function that bounds \mathcal{J} from above, and it is equal to \mathcal{J} in the point x_n . We call this function a *surrogate function*. The function in (52) being $\frac{1}{\gamma}$ -strongly convex, we deduce from (52) that

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

By applying (53) to x_{n+1} and adding the previous inequality we obtain

$$\mathcal{J}(x_{n+1}) + \frac{1}{\gamma} \|x_{n+1} - x_n\|^2 \leq \mathcal{J}(x_n) + \frac{L}{2} \|x_{n+1} - x_n\|^2. \quad \square$$

With all these elements, we can now show the convergence of the Forward-Backward algorithm, also known in the literature as the Proximal Gradient Descent (PGD) algorithm.

Theorem 8 (Forward-Backward algorithm). *Let $\mathcal{J} = f + g$ be a sum of two convex, coercive, lower semi-continuous functions that are bounded from below. We suppose that f is differentiable with an L -Lipschitz gradient. Let $\gamma < \frac{2}{L}$ and $x_0 \in E$, and let $(x_n)_{n \in \mathbb{N}}$ be the sequence defined for all $n \in \mathbb{N}$ by*

$$x_{n+1} = \text{prox}_{\gamma g}(\text{Id} - \gamma \nabla f)(x_n). \quad (54)$$

Then the sequence $(x_n)_{n \in \mathbb{N}}$ converges to a minimizer of \mathcal{J} .

Proof. As for the proof of the gradient descent method, we use Lemma 4 to prove the theorem. We first show that the operator T is 1-Lipschitz and then demonstrate that the sequence $\|x_n - x_{n+1}\|$ goes to zero.

The operator T is 1-Lipschitz as a composition of a proximity operator that is 1-Lipschitz (Proposition 8) and an operator of the form $\text{Id} - \gamma \nabla f$ that is also 1-Lipschitz under the condition that $\gamma \leq \frac{2}{L}$ (Proposition 4). The second point is then given by Lemma 7. \square

In the following section we will see that it is possible to control the speed of the functional's decrease towards its minimum.

4.3.3 Fast Iterative Shrinkage Thresholding Algorithm (FISTA)

In the early 60' Polyak introduced inertial methods in [38] to improve the convergence rate of gradient descent algorithm for C^2 strongly convex functions. The goal of inertial methods is to accelerate the gradient descent algorithm without any second order information on f , that is the Hessian of f .

Such an inertial algorithm uses the memory of the past trajectory at points x_n and x_{n-1} to compute x_{n+1} . It is also known as Heavy Ball because the optimization scheme can be seen as a discretization of an ODE describing the position of a ball submitted to a force field ∇f with a friction proportional to its speed:

$$\ddot{x}(t) + a\dot{x}(t) + \nabla f(x(t)) = 0. \quad (55)$$

If f is L -smooth and α -strongly convex, we have seen in Theorem 3 (relation (24)) that the sequence $(x_n)_{n \in \mathbb{N}}$ provided by the gradient descent satisfies

$$f(x_n) - f(x^*) = O(e^{-\frac{\alpha}{L}n}).$$

If we also assume that f is C^2 , the Heavy Ball method provides a sequence $(x_n)_{n \in \mathbb{N}}$ ensuring the following rate, that is better if $\frac{\alpha}{L} < 1$, which is often the case for large scale problems:

$$f(x_n) - f(x^*) = O(e^{-4\sqrt{\frac{\alpha}{L}}n}).$$

In this section we focus on two inertial algorithms inspired by the work of Polyak to minimize composite functions:

$$\min_{x \in E} \mathcal{J}(x) = \min_{x \in E} f(x) + g(x). \quad (56)$$

The first one is FISTA by Beck and Teboulle [7]. This algorithm, based on an original idea by Nesterov [35], is dedicated to convex functions. The second one, called V-FISTA [6], can be seen as an adaptation of the original Heavy ball algorithm to composite strongly convex functions.

FISTA for convex functions To minimize a convex function $\mathcal{J} = f+g$ where f is convex and L -smooth and g is convex, the FISTA (Fast Iterative Soft Thresholding Algorithm) algorithm [7] writes

$$\begin{cases} y_n &= x_n + \frac{t_n-1}{t_{n+1}}(x_n - x_{n-1}) \\ x_{n+1} &= \text{prox}_{\gamma g}(y_n - \gamma \nabla f(y_n)), \end{cases} \quad (57)$$

where $\gamma \leq \frac{1}{L}$, $t_1 = 1$ and $t_{n+1} = \frac{1+\sqrt{1+4t_n^2}}{2}$. The following result shows that this sequence generally performs better than Forward-Backward.

Theorem 9. *The sequence generated by (57) satisfies*

$$\mathcal{J}(x_n) - \mathcal{J}(x^*) \leq \frac{2}{\gamma(n+1)^2} \|x_0 - x^*\|^2. \quad (58)$$

Several remarks can be done before going any further.

- The convergence rate of FISTA, $O(\frac{1}{n^2})$ is better than the rate $O(\frac{1}{n})$ of Forward-Backward (FB). Numerically it performs better, try it.

- FISTA should have been called Fast Forward Backward, but the authors prefer a reference to ISTA (Iterative Soft Thresholding Algorithm) which is a particular case of FB to get a better name...
- Y. Nesterov proposed the same scheme with $T = Id - \gamma \nabla f$ to minimize a differentiable function in 1983 with the same rate [35]. For this reason, FISTA is also called a Nesterov's acceleration of FB.
- The decay is optimal in the sense that no first order method (using only gradient or subgradient) may provide a rate $O(\frac{1}{n^\delta})$ with $\delta > 2$ [35]. Using extra previous iterates x_k with $k < n - 1$ does not help to get a better rate.
- With the original choice of the sequence $(t_n)_{n \in \mathbb{N}^*}$, there is no known proof of the convergence of the sequence $(x_n)_{n \in \mathbb{N}}$ generated by FISTA.

The analysis of this algorithm with the sequence $(t_n)_{n \in \mathbb{N}^*}$ defined in (57) was not understood until the works of [11] and [45] who proposed a complete proof of convergence for a modified sequence $(t_n)_{n \in \mathbb{N}^*}$, while noticing that FISTA can be seen as a discretization of the following ODE

$$\ddot{x}(t) + \frac{3}{t}\dot{x}(t) + \nabla f(x(t)) = 0. \quad (59)$$

More precisely, the proof of Theorem 9 can be obtained by relying on a Lyapunov analysis, by showing that the following sequence

$$S_n = t_n^2(f(x_n) - f(x^*)) + \frac{1}{2\gamma} \|t_n(x_n - x_{n-1}) + 2(x_n - x^*)\|^2 \quad (60)$$

is non-increasing if

$$t_{n+1}^2 - t_{n+1} + t_n^2 \leq 0. \quad (61)$$

The choice in [35, 7] is to maximize t_n at each step, to get the best decay. Other definition of t_n satisfying (61) can nevertheless be considered, to ensure both similar asymptotic decay and weak convergence of the sequence $(x_n)_{n \in \mathbb{N}^*}$. As proposed in [11], the following version of FISTA

$$\begin{cases} y_n &= x_n + \frac{n}{n+\beta}(x_n - x_{n-1}) \\ x_{n+1} &= \text{prox}_{\gamma g}(y_n - \gamma \nabla f(y_n)) \end{cases} \quad (62)$$

ensures that

$$\mathcal{J}(x_n) - \mathcal{J}(x^*) \leq \frac{\beta + 1}{2\gamma(n+1)^2} \|x_0 - x^*\|^2, \quad (63)$$

for $\gamma \leq \frac{1}{L}$ and $\beta > 3$. Unfortunately, there is no rule to choose β . In practice, if no additional assumptions are made on \mathcal{J} , i.e. \mathcal{J} is only assumed convex, any value $\beta > 3$ seems to be a good choice. When \mathcal{J} is strongly convex, β must be chosen depending on ε if ε is the numerical precision we want to reach [3].

V-FISTA for strongly convex functions. When the function $\mathcal{J} = f + g$ is α -strongly convex and α is known, there exist several ways to improve the convergence reached by the Forward-Backward algorithm. To avoid many complex definitions, we only focus here on the strongly convex hypothesis, but this assumption could be weakened. The analysis of Polyak in his seminal work [38] deals with C^2 functions. The Heavy Ball has been extended to composite functions in [6], with an inertial scheme that can be seen as a discretization of a Heavy Ball ODE (59) with a different friction parameter a and a different stepsize $\gamma = \frac{1}{L}$:

$$\begin{cases} y_n &= x_n + \frac{\sqrt{L}-\sqrt{\alpha}}{\sqrt{L}+\sqrt{\alpha}}(x_n - x_{n-1}) \\ x_{n+1} &= T(y_n) := \text{prox}_{\gamma g}(y_n - \gamma \nabla f(y_n)). \end{cases} \quad (64)$$

This scheme provides a sequence such that

$$\mathcal{J}(x_n) - \mathcal{J}(x^*) \leq \left(1 - \sqrt{\frac{\alpha}{L}}\right)^n (\mathcal{J}(x_0) - \mathcal{J}(x^*)). \quad (65)$$

We finally make two remarks on this result.

- The strong convexity parameter α must be known to fix the inertial parameter, which is a difference between FISTA and V-FISTA.
- If $\frac{\alpha}{L} \ll 1$, this rate is much better than Forward-Backward.

4.3.4 Douglas-Rachford algorithm (DR)

We now target the problem

$$\min_{x \in E} \mathcal{J}(x) = \min_{x \in E} f(x) + g(x),$$

where the functions f and g are convex, proper, lower semi-continuous and where \mathcal{J} is bounded from below. The difference with the previous setting is that we do not make any assumptions on the differentiability of f . We also assume that we are able to compute the proximity operators of both f and g . The first thing to notice is that under these assumptions, there exists a minimizer of \mathcal{J} .

As before, we can identify an operator that is 1-Lipschitz and whose fixed points are associated to the minimizers of \mathcal{J} . In the case of Forward-Backward, these fixed points were the minimizers of \mathcal{J} , whereas for the Douglas-Rachford algorithm, the minimizers are the images of these fixed points through an operator. To introduce this operator, we first need to define the *reflected proximity operator*.

Definition 15. *The reflected prox or Rprox of a proper and convex function f is*

$$\text{Rprox}_f = 2 \text{prox}_f - \text{Id}.$$

The Douglas-Rachford algorithm is then based on the following proposition.

Proposition 12. *Let $\mathcal{J} = f + g$ be a functional defined from E to $\mathbb{R} \cup +\infty$, where the two functions f and g are both convex, proper, lower semi-continuous and satisfying the hypotheses of Lemma 6 (on the sum of subdifferentials), and let γ be a strictly positive real number. Then we have*

$$\text{zeros}(\partial\mathcal{J}) = \text{prox}_{\gamma g}(\text{Fix}(\text{Rprox}_{\gamma f} \text{Rprox}_{\gamma g})). \quad (66)$$

Proof. Under the hypotheses of Lemma 6, we have

$$\begin{aligned} 0 \in \partial\mathcal{J}(x) &\Leftrightarrow 0 \in \partial\gamma\mathcal{J}(x) \\ &\Leftrightarrow 0 \in \partial\gamma f(x) + \partial\gamma g(x) \\ &\Leftrightarrow \exists z \in E \text{ such that } -z \in \partial\gamma f(x) \text{ and } z \in \partial\gamma g(x) \\ &\Leftrightarrow \exists y \in E \text{ such that } x - y \in \partial\gamma f(x) \text{ and } y - x \in \partial\gamma g(x) \end{aligned}$$

We rewrite $x - y \in \partial\gamma f(x)$ as $2x - y \in (\text{Id} + \partial\gamma f)(x)$. The relation $y - x \in \partial\gamma g(x)$ can also be rewritten as $y \in (\text{Id} + \gamma\partial g)(x)$ and thus $x = \text{prox}_{\gamma g}(y)$, which gives

$$0 \in \partial\mathcal{J}(x) \Leftrightarrow \exists y \in E \text{ such that } 2x - y \in (\text{Id} + \partial\gamma f)(x) \text{ and } x = \text{prox}_{\gamma g}(y).$$

By using the definition of the operator Rprox we obtain:

$$\begin{aligned} 0 \in \partial\mathcal{J}(x) &\Leftrightarrow \exists y \in E \text{ such that } x = \text{prox}_{\gamma f}(\text{Rprox}_{\gamma g} y) \text{ and } x = \text{prox}_{\gamma g}(y) \\ &\Leftrightarrow \exists y \in E \text{ such that } y = 2x - \text{Rprox}_{\gamma g} y = \text{Rprox}_{\gamma f}(\text{Rprox}_{\gamma g} y) \text{ and } x = \text{prox}_{\gamma g}(y) \\ &\Leftrightarrow \exists y \in E \text{ such that } y \in \text{Fix}(\text{Rprox}_{\gamma f} \text{Rprox}_{\gamma g}) \text{ and } x = \text{prox}_{\gamma g}(y). \end{aligned}$$

□

From this proposition we can define a minimization algorithm for \mathcal{J} that estimates a fixed point of the operator $T = \text{Rprox}_{\gamma f} \text{Rprox}_{\gamma g}$ and then applies $\text{prox}_{\gamma f}$ to this fixed point to get a minimizer of \mathcal{J} . In order to estimate such a fixed point, we first give a key property of Rprox operators related to firm nonexpansiveness (see Definition 14).

Lemma 8. *Let T be an operator defined from E to E . The following two statements are equivalent:*

- T is firmly nonexpansive
- $R = 2T - \text{Id}$ is nonexpansive (1-Lipschitz)

Proof. We show that these two properties are equivalent to the nonnegativeness of a scalar product. Let $(x, y) \in E^2$. By setting $u = Tx - Ty$ and $v = Tx - x - (Ty - y)$, we have

$$\|x - y\|^2 = \|u - v\|^2 = \|u\|^2 + \|v\|^2 - 2\langle u, v \rangle$$

thus T is firmly nonexpansive if and only if $\langle u, v \rangle \geq 0$. Additionally,

$$\|Rx - Ry\|^2 = \|u + v\|^2 = \|u\|^2 + \|v\|^2 + 2\langle u, v \rangle = \|x - y\|^2 + 4\langle u, v \rangle$$

which shows that the nonexpansiveness of R is equivalent to $\langle u, v \rangle \leq 0$. □

Corollary 2. *For any convex function f and parameter $\tau > 0$, the operator $\text{Rprox}_{\tau f}$ is nonexpansive.*

Proof. The result follows from Lemma 8, given the definition of the Rprox operator (Definition 15) and the fact that the proximal operator of a convex function is firmly nonexpansive (Proposition 8). \square

The Rprox operator is thus nonexpansive (1-Lipschitz), but this is not sufficient to show the convergence of $x_{n+1} = \text{Rprox}_{\gamma f} \text{Rprox}_{\gamma g}(x_n)$. Indeed, contrary to gradient descent and Forward-Backward algorithms, we do not have a descent Lemma ensuring that $\lim_{n \rightarrow \infty} \|x_{n+1} - x_n\| = 0$ to apply Lemma 4. Hence we rather rely on the Krasnosel'skii-Mann Algorithm that considers an averaging process to target fixed-points of nonexpansive operators.

Theorem 10 (Krasnosel'skii-Mann Algorithm). *Let D be a non-empty, closed and convex subset of E and let $T, D \rightarrow D$ be a 1-Lipschitz operator such that the set of fixed points of T is non-empty. Let $(\lambda_n)_{n \in \mathbb{N}}$ be a sequence of real numbers in $[0, 1]$ such that $\sum_{n \in \mathbb{N}} \lambda_n(1 - \lambda_n) = +\infty$, and take $x_0 \in D$. We define*

$$\forall n \in \mathbb{N}, \quad x_{n+1} = x_n + \lambda_n(Tx_n - x_n).$$

Proof. Let y be a fixed point of T (i.e. $Ty = y$) and $n \in \mathbb{N}$, we have

$$\begin{aligned} \|x_{n+1} - y\|^2 &= \|(1 - \lambda_n)(x_n - y) + \lambda_n(Tx_n - y)\|^2 \\ &= (1 - \lambda_n)\|x_n - y\|^2 + \lambda_n\|Tx_n - Ty\|^2 - \lambda_n(1 - \lambda_n)\|Tx_n - x_n\|^2 \\ &\leq \|x_n - y\|^2 - \lambda_n(1 - \lambda_n)\|Tx_n - x_n\|^2, \end{aligned}$$

where we used the following relation:

$$2\langle x_n - y, Tx_n - y \rangle = \|Tx_n - y\|^2 - \|Tx_n - x_n\|^2 + \|x_n - y\|^2,$$

as well as the fact that T is non expansive for the last inequality. We deduce that

$$\sum_{n \in \mathbb{N}} \lambda_n(1 - \lambda_n)\|Tx_n - x_n\|^2 \leq \|x_0 - y\|^2. \quad (67)$$

Also observe that

$$\begin{aligned} \|Tx_{n+1} - x_{n+1}\| &= \|Tx_{n+1} - Tx_n + (1 - \lambda_n)(Tx_n - x_n)\| \\ &\leq \|x_{n+1} - x_n\| + (1 - \lambda_n)\|Tx_n - x_n\| \\ &\leq \|Tx_n - x_n\| \end{aligned}$$

As $\sum_{n \in \mathbb{N}} \lambda_n(1 - \lambda_n) = +\infty$ and as the sequence $(\|Tx_n - x_n\|)_{n \in \mathbb{N}}$ is non increasing we deduce from (67) that $(Tx_n - x_n)_{n \in \mathbb{N}}$ converges to 0. The sequence $\|x_n - y\|$ being non increasing, we get that all elements from the sequence $(x_n)_{n \in \mathbb{N}}$ belong to a close ball of center y and radius $\|y - x_0\|$, which is a compact since E is finite-dimensional. Hence we can extract a subsequence $(x_{n_k})_{k \in \mathbb{N}}$ that converges to a point $x \in E$.

As $Tx_n - x_n$ converges to 0, the sequence $(Tx_{n_k} - x_{n_k})_{k \in \mathbb{N}}$ goes to 0 too. We deduce that the sequence $(Tx_{n_k})_{k \in \mathbb{N}}$ also converges to x . Since this sequence also converges to Tx , we get that $x = Tx$ is a fixed point of T . Finally, since x is a fixed point of T , the sequence $(\|x_n - x\|)_{n \in \mathbb{N}}$ is non increasing. We can extract a subsequence that converges to 0, which implies that $(x_n)_{n \in \mathbb{N}}$ converges to x a fixed point of T . \square

We can now show the convergence of the Douglas-Rachford algorithm.

Theorem 11 (Douglas-Rachford algorithm). *Let f and g be two convex, proper, lower semi-continuous functions, bounded from below. Let $(\mu_n)_{n \in \mathbb{N}}$ be a sequence of elements in $[0, 2]$ such that $\sum_{n \in \mathbb{N}} \mu_n(2 - \mu_n) = +\infty$. Let $\gamma > 0$ and $x_0 \in E$. Let $(x_n)_{n \in \mathbb{N}}$, $(y_n)_{n \in \mathbb{N}}$ and $(z_n)_{n \in \mathbb{N}}$ be the sequences defined by*

$$\forall n \in \mathbb{N} \begin{cases} y_n &= \text{prox}_{\gamma g}(x_n), \\ z_n &= \text{prox}_{\gamma f}(2y_n - x_n), \\ x_{n+1} &= x_n + \mu_n(z_n - y_n), \end{cases} \quad (68)$$

which is equivalent to

$$x_{n+1} = x_n + \frac{\mu_n}{2} (\text{Rprox}_{\gamma f} \text{Rprox}_{\gamma g}(x_n) - x_n).$$

Then there exists $x \in E$ minimizing $(f + g)$ such that $(x_n)_{n \in \mathbb{N}}$ converges to x .

Proof. First observe that

$$\begin{aligned} x_{n+1} &= x_n + \mu_n(z_n - y_n) \\ &= x_n + \mu_n (\text{prox}_{\gamma f}(2y_n - x_n) - \text{prox}_{\gamma g}(x_n)) \\ &= x_n + \mu_n (\text{prox}_{\gamma f}(\text{Rprox}_{\gamma g}(x_n)) - \text{prox}_{\gamma g}(x_n)) \\ &= x_n + \frac{\mu_n}{2} (2 \text{prox}_{\gamma f}(\text{Rprox}_{\gamma g}(x_n)) - 2 \text{prox}_{\gamma g}(x_n) + x_n - x_n) \\ &= x_n + \frac{\mu_n}{2} (\text{Rprox}_{\gamma f}(\text{Rprox}_{\gamma g}(x_n)) - \text{Rprox}_{\gamma g}(x_n) - x_n) \\ &= x_n + \frac{\mu_n}{2} (\text{Rprox}_{\gamma f}(\text{Rprox}_{\gamma g}(x_n)) - x_n). \end{aligned}$$

Corollary 2 states that the Rprox operator is 1-Lipschitz. We now define the operator $T = \text{Rprox}_{\gamma f} \text{Rprox}_{\gamma g}$. We know that this operator is 1-Lipschitz as a composition of two operators that are 1-Lipschitz. From Proposition 12, we also know that the set of fixed points of T is equal to the image by the 1-Lipschitz operator $\text{prox}_{\gamma f}$ of the set of minimizers of $f + g$, which is non-empty according to the hypotheses on f and g . We note that $x_{n+1} = x_n + \frac{\mu_n}{2}(Tx_n - x_n)$. We conclude by applying Krasnoselsky-Mann's algorithm presented in Theorem 10. \square

Other formulations of Douglas-Rachford The Douglas-Rachford algorithm can be expressed in many forms in the literature. For example, it is frequent for the parameters μ_n to be fixed to 1. The algorithm is then expressed in the following way:

$$\forall n \in \mathbb{N} \begin{cases} y_n &= \text{prox}_{\gamma g}(x_n), \\ z_n &= \text{prox}_{\gamma f}(2y_n - x_n), \\ x_{n+1} &= x_n + z_n - y_n \end{cases} \quad (69)$$

By omitting the variable z_n we obtain the following description of the sequences $(x_n)_{n \in \mathbb{N}}$ and $(y_n)_{n \in \mathbb{N}}$:

$$\forall n \geq 1 \begin{cases} x_{n+1} = x_n + \text{prox}_{\gamma f}(2y_n - x_n) - y_n, \\ y_{n+1} = \text{prox}_{\gamma g}(x_{n+1}). \end{cases} \quad (70)$$

We also mention the change of variables $w_n = y_n - x_n$ that appears in the literature:

$$\forall n \geq 1 \begin{cases} u_{n+1} = \text{prox}_{\gamma f}(y_n + w_n) \\ y_{n+1} = \text{prox}_{\gamma g}(u_{n+1} - w_n), \\ w_{n+1} = w_n + y_{n+1} - u_{n+1}. \end{cases} \quad (71)$$

This algorithm can also be adapted to the case where the minimization problem is of the form:

$$\min_{x \in E} f(Ax) + g(x)$$

where A is a linear operator from E to F . Indeed, one can consider the equivalent problem:

$$\min_{x, y \in E \times F} f(y) + g(x) + \iota_{Ax=y}(x, y),$$

and apply the Douglas-Rachford algorithm to minimize the sum of the two convex functions $f(y) + g(x)$ and $\iota_{Ax=y}(x, y)$, with respect to the augmented variable $v = (x, y)$.

4.3.5 Parallel ProXimal Algorithm (PPXA)

The PPXA algorithm (see Chapter 10 in [15]) adapts the Douglas-Rachford algorithm to solve:

$$\min_{x \in E} \sum_{i=1}^M f_i(x), \quad (72)$$

where the M function f_i are convex proper and l.s.c. The PPXA algorithm increases the dimension of the problem and it allows for the parallel computation of the proximal operators prox_{f_i} . Let us denote as X a vector in E^M that writes $X = (x_1, x_2, \dots, x_M)$ with $x_i \in E$ for all $i \leq M$. PPXA then consists in reformulating the problem (72) as an optimization problem in E^M :

$$\min_{X=(x_i)_{i \leq n} \in E^M} \sum_{i=1}^M f_i(x_i) + \iota_D(X) \quad (73)$$

where $D = \{X = (x_n)_{k \leq M} \in E^M \text{ such that } x_n = x_1, \forall k \leq M\}$.

Assuming that the optimization problem (72) has at least one solution, a minimizer of (72) can be obtained by applying the Douglas-Rachford algorithm to the problem $\min_{X \in E^M} \tilde{f}(X) + \tilde{g}(X)$ with $\tilde{f}(X) = \sum_{i=1}^M f_i(x_i)$ and $\tilde{g}(X) = \iota_D(X)$. The proximity operator of \tilde{f} can be computed in a parallel way using the separability of \tilde{f} , following Example (1). One can also show that the proximity operator of the indicator of the set D is a simple mean.

We finally provide the following observations about PPXA:

- Even when only $M = 2$ functions are involved, applying PPXA is different from applying Douglas-Rachford. The computation of proximal operators is parallel with PPXA, and it may be faster to use PPXA than Douglas-Rachford for $M = 2$ functions.
- If the functions f_i for $i \geq 2$ are composed with linear operators L_i :

$$\min_{x \in E} f_1(x) + \sum_{i=2}^M f_i(L_i x) \quad (74)$$

a similar reformulation can be considered:

$$\min_{X=(x_i)_{i \leq M} \in E^M} \sum_{i=1}^M f_i(x_i) + \iota_{D_2}(X) \quad (75)$$

where $D_2 = \{X = (x_i)_{i \leq M} \in E^M \text{ such that } L_i x_1 = x_i, \forall i \leq M\}$. To solve problem (75) with the Douglas-Rachford algorithm, one needs to perform the projection on D_2 . It exists a closed form for this projection, which is similar to (43). If we denote $(p_1, p_2, \dots, p_n) = Proj_{D_2}(X)$ then $\forall k \geq 2$ we have $p_k = L_k p_1$ and

$$p_1 = \left(\text{Id} + \sum_{i=2}^M L_i^* L_i \right)^{-1} \left(x_1 + \sum_{i=2}^M L_i^* x_i \right).$$

To perform the inversion in a general case (without assumptions on the operators L_i), a conjugate gradient method can be considered.

4.3.6 Alternating Direction Method of Multipliers (ADMM)

The ADMM algorithm [21, 23] is designed to solve optimization problems of the form:

$$\min_{(x,y) \in E \times F, Ax+By=b} f(x) + g(y) \quad (76)$$

where A and B are two linear operators taking values from E and F to G , b is a vector in G and f and g are two convex, proper, lower semi-continuous functions. It is a general formulation, containing cases such as $y = x$ or $y = Ax$.

The Lagrangian associated to (76) to account for the constraint $Ax + By = b$ with a multiplier variable $z \in G$ writes

$$L(x, y, z) = f(x) + g(y) + \langle z, Ax + By - b \rangle, \quad (77)$$

and the augmented Lagrangian for $\gamma > 0$ is

$$L_\gamma(x, y, z) = L(x, y, z) + \frac{\gamma}{2} \|Ax + By - b\|^2. \quad (78)$$

Let $(x_0, y_0) \in E \times F$, $\gamma > 0$ and $z^0 \in G$. The ADMM introduces the sequences $(x_n)_{n \in \mathbb{N}}$, $(y_n)_{n \in \mathbb{N}}$ and $(z_n)_{n \in \mathbb{N}}$ defined $\forall n \geq 0$ as:

$$\begin{cases} x_{n+1} &= \operatorname{argmin}_x f(x) + \langle z_n, Ax \rangle + \frac{\gamma}{2} \|Ax + By_n - b\|^2 \\ y_{n+1} &= \operatorname{argmin}_y g(y) + \langle z_n, By \rangle + \frac{\gamma}{2} \|Ax_{n+1} + By - b\|^2 \\ z_{n+1} &= z_n + \gamma(Ax_{n+1} + By_{n+1} - b) \end{cases} \quad (79)$$

The ADMM is so called because it can be seen as a variant of an algorithm known as the Augmented Lagrangian Method. If we replace the iterative updates of x and y by a joint update step:

$$(x_{n+1}, y_{n+1}) = \operatorname{arg} \min_{(x,y) \in E \times F} f(x) + g(y) + \langle z_n, Ax + By \rangle + \frac{\gamma}{2} \|Ax + By - b\|^2,$$

we obtain the Augmented Lagrangian method that consists in penalizing the constraints with a Lagrange multiplier z and a quadratic term. One of the problems with this method is that joint minimization is generally difficult to perform. The ADMM separates this problem by optimizing with respect to the first variable x first, and then the second variable y , while the variable z can be interpreted as a Lagrange multiplier that is updated at each iteration.

Theorem 12 (ADMM algorithm). *Let f and g be two proper, convex, coercive, lower semi-continuous functions. Let the sequences $(x_n)_{n \in \mathbb{N}}$, $(y_n)_{n \in \mathbb{N}}$ and $(z_n)_{n \in \mathbb{N}}$ be defined as in the ADMM algorithm (79), then*

1. *The sequence $(f(x_n) + g(y_n))_{n \in \mathbb{N}}$ converges to the minimum value of $f + g$.*
2. *The sequences $(x_n)_{n \in \mathbb{N}}$ and $(y_n)_{n \in \mathbb{N}}$ converge.*
3. *The sequence $(Ax_n + By_n - b)_{n \in \mathbb{N}}$ goes to zero.*

In the specific case where $A = \operatorname{Id}$, $B = -\operatorname{Id}$ and $b = 0$ (i.e. $x = y$), the ADMM algorithm (79) can be expressed using proximity operators:

$$\begin{cases} x_{n+1} &= \operatorname{prox}_{f/\gamma}(y_n - z_n/\gamma) \\ y_{n+1} &= \operatorname{prox}_{g/\gamma}(x_{n+1} + z_n/\gamma) \\ z_{n+1} &= z_n + \gamma(x_{n+1} - y_{n+1}), \end{cases} \quad (80)$$

which, for $\gamma = 1$, is one of the forms of Douglas-Rachford, see (71).

To prove Theorem 12 in the general case, it is possible to show that applying ADMM is equivalent to applying the Douglas-Rachford algorithm on a different problem, called the dual problem of (76), and this independently of the choice of the operators A and B . We will show an equivalence between both algorithms in section 5.3.2, once the adequate material on duality has been introduced.

5 Duality

In this section we present optimization algorithms that rely on the dual formulations of convex problems. In section 5.1, we first introduce the key dual notion of convex conjugate f^* of a function f , and detail relevant properties of this transform. Then, we study primal-dual algorithms in section 5.2. In section 5.3, we finally show equivalence, in particular settings, between the Chambolle-Pock primal dual algorithm, the Douglas-Rachford algorithm and the ADMM.

5.1 Properties of convex conjugates

There exist close relationships between convex conjugate, subdifferential and proximal operators. These links will be useful to show the equivalence between primal and dual problems.

Definition 16 (Convex conjugate). *Let f be a function defined from E to $\bar{\mathbb{R}}$, the convex (or Fenchel or Legendre-Fenchel) conjugate of f is the function defined from E to $\bar{\mathbb{R}}$ by*

$$f^*(u) = \sup_{x \in E} (\langle x, u \rangle - f(x)). \quad (81)$$

Remark 8. *The function $\langle x, u \rangle - f(x)$ is linear (and thus convex) in u . The supremum of convex functions being convex, f^* is necessarily a convex function.*

Conjugation offers a geometric interpretation of convex, proper and l.s.c functions: at any point, f is equal to the supremum of all affine functions that are lower bounds of f .

Example 2. *Let us give examples of convex conjugates*

- *If $f(x) = \langle a, x \rangle - b$ then*

$$f^*(u) = \begin{cases} b & \text{if } u = a \\ +\infty & \text{if } u \neq a. \end{cases}$$

- *If $f(x) = \|x\|_p$ where $\|\cdot\|_p$ is the ℓ_p norm, then f^* is the indicator function of the unit ball for the dual norm ℓ_q (such that $1/p + 1/q = 1$), i.e $f^*(u) = i_{\mathcal{B}_q}(u)$, for $\mathcal{B}_q = \{u, \|u\|_q \leq 1\}$. Thus, for $f(x) = \|x\|_1$, we have*

$$f^*(u) = \begin{cases} 0 & \text{if } \|u\|_\infty \leq 1 \\ +\infty & \text{if } \|u\|_\infty > 1. \end{cases}$$

Next lemma gives a main relation between subdifferentials and convex conjugates.

Lemma 9. *Let f be a function defined on E and $x \in E$, then*

$$u \in \partial f(x) \Leftrightarrow f^*(u) + f(x) = \langle u, x \rangle \quad (82)$$

Proof.

$$\begin{aligned}
u \in \partial f(x) &\Leftrightarrow \forall y \in E, f(y) \geq f(x) + \langle y - x, u \rangle. \\
&\Leftrightarrow \forall y \in E, \langle x, u \rangle - f(x) \geq \langle y, u \rangle - f(y). \\
&\Leftrightarrow \langle x, u \rangle - f(x) = f^*(u) \\
&\Leftrightarrow f^*(u) + f(x) = \langle x, u \rangle.
\end{aligned}$$

□

One essential property of convex functions is equality to its biconjugate. First we show a property of the biconjugate for a general, nonconvex function, and then prove the equality between f and f^{**} if f is convex.

Lemma 10. *Let f be a function defined from E to $[-\infty, +\infty]$, for all $x \in E$, $f^{**}(x) \leq f(x)$. Moreover, f^{**} is the largest convex lower semi-continuous envelope of the function satisfying $f^{**}(x) \leq f(x)$, for all $x \in E$*

Proof. Let $\Sigma \subset E \times \mathbb{R}$ the set of pairs (u, α) representing affine functions $x \mapsto \langle u, x \rangle - \alpha$ that are upper bounded by f :

$$\begin{aligned}
(u, \alpha) \in \Sigma &\Leftrightarrow f(x) \geq \langle u, x \rangle - \alpha, \forall x \in E \\
&\Leftrightarrow \alpha \geq \sup_{x \in E} \langle u, x \rangle - f(x) \\
&\Leftrightarrow \alpha \geq f^*(u), \text{ and } u \in \text{dom}(f^*).
\end{aligned}$$

Then we obtain, for all $x \in E$:

$$\begin{aligned}
\sup_{(u, \alpha) \in \Sigma} \langle u, x \rangle - \alpha &= \sup_{\substack{u \in \text{dom}(f^*) \\ -\alpha \leq -f^*(u)}} \langle u, x \rangle - \alpha \\
&= \sup_{u \in \text{dom}(f^*)} \langle u, x \rangle - f^*(u) = f^{**}(x).
\end{aligned}$$

□

Theorem 13 (Biconjugation). *If f and is a proper l.s.c convex function defined from E to $[-\infty, +\infty]$, then $f^{**} = f$.*

Proof. If f is convex proper and l.s.c then Lemma 10 ensures that f is equal to its largest convex lower semi-continuous envelope, that is to say $f^{**} = f$. □

We deduce from this theorem that a convex, proper and l.s.c function f satisfies $\forall x \in E$

$$f(x) = \sup_{u \in E} (\langle x, u \rangle - f^*(u)) \quad (83)$$

Remark 9. *Computations in the proof of Lemma 10 show that the supremum in (81) is reached at all points $u \in \partial f(x)$. Hence, if the subdifferential $\partial f(x)$ is not empty, which is the case in most practical applications, the supremum (sup) is a maximum (max) in relation (81).*

Remark 10. Relation (83) is useful when f is composed with an operator. For instance, if K is a linear application from E to another space F , we have $\forall x \in E$

$$f(Kx) = \sup_{u \in F} (\langle Kx, u \rangle - f^*(u)) = \sup_{u \in F} (\langle x, K^*u \rangle - f^*(u)). \quad (84)$$

With such an expression, the operator K can be decoupled from the convex function f . This is a key ingredient of the primal-dual algorithms detailed below.

Let us give a relevant properties between subgradients of conjugate functions.

Proposition 13. *If f is a convex, proper and l.s.c function, then*

$$u \in \partial f(x) \Leftrightarrow x \in \partial f^*(u). \quad (85)$$

Proof. Let $(x, u) \in E^2$ such that $u \in \partial f(x)$. Lemma 9 ensures that

$$u \in \partial f(x) \Leftrightarrow f^*(u) + f(x) = \langle u, x \rangle.$$

As f is convex, proper and l.s.c, we have $f = f^{**}$ so that

$$u \in \partial f(x) \Leftrightarrow f^*(u) + f^{**}(x) = \langle u, x \rangle$$

which implies $u \in \partial f(x) \Leftrightarrow x \in \partial f^*(u)$. □

We now give a useful theorem involving convex conjugates and proximal operators.

Corollary 3 (Moreau's identity [33]). *If f is a convex, proper and l.s.c function, then for all $\gamma > 0$*

$$\text{prox}_{\gamma f}(x) + \gamma \text{prox}_{f^*/\gamma}(x/\gamma) = x. \quad (86)$$

that is to say, for $\gamma = 1$, $\text{prox}_f + \text{prox}_{f^} = \text{Id}$.*

Proof. Using Proposition 6 and relation (85), we have

$$\begin{aligned} p = \text{prox}_{\gamma f}(x) &\Leftrightarrow (x - p) \in \gamma \partial f(p) \\ &\Leftrightarrow p \in \partial f^*((x - p)/\gamma) \\ &\Leftrightarrow p/\gamma \in \frac{1}{\gamma} \partial f^*((x - p)/\gamma) \\ &\Leftrightarrow x/\gamma - (x - p)/\gamma \in \partial(f^*/\gamma)((x - p)/\gamma) \\ &\Leftrightarrow (x - p)/\gamma = \text{prox}_{f^*/\gamma}(x/\gamma) \\ &\Leftrightarrow \text{prox}_{\gamma f} + \gamma \text{prox}_{f^*/\gamma}(\cdot/\gamma) = \text{Id}. \end{aligned}$$

□

We finally give an equivalence between L -smoothness and strong convexity of conjugate functions.

Proposition 14. *Let f be a convex proper and l.s.c function, then the two following assertions are equivalent: (i) f is $1/L$ -strongly ; (ii) f^* is L -smooth.*

Proof. (i) \Rightarrow (ii): If f is $1/L$ -strongly convex, then we can extend relation (9) from Proposition (5) to non smooth function using the definition of subdifferential (39). For all $(x, y) \in E^2$, $p \in \partial f(x)$ and $q \in \partial f(y)$, we obtain:

$$\|p - q\| \geq \frac{1}{L}\|x - y\|.$$

Using relation (85) it implies:

$$\|p - q\| \geq \frac{1}{L}\|\partial f^*(p) - \partial f^*(q)\|,$$

so that ∂f^* has a L -Lipschitz continuous gradient.

(ii) \Rightarrow (i): If f^* is L -smooth, then the co-coercivity property (6) gives $\forall (p, q) \in E^2$:

$$\langle \nabla f^*(p) - \nabla f^*(q), p - q \rangle \geq \frac{1}{L}\|\nabla f^*(p) - \nabla f^*(q)\|^2.$$

Denoting as $x = \nabla f^*(p)$ and $y = \nabla f^*(q)$, relation (85) leads to $p \in \partial f(x)$ and $q \in \partial f(y)$. Hence we have

$$\begin{aligned} \langle x - y, \partial f(x) - \partial f(y) \rangle &\geq \frac{1}{L}\|x - y\|^2 \\ \|\partial f(x) - \partial f(y)\| &\geq \frac{1}{L}\|x - y\|, \end{aligned}$$

and f^* is $1/L$ -strongly convex. □

5.2 Primal-Dual Algorithms

We now consider the optimization problem:

$$\min_{x \in E} f(Kx) + g(x), \tag{87}$$

where K is a linear operator from E to F and f and g are two convex proper l.s.c functions defined from F (resp. E) to $[-\infty, +\infty]$. In the following, we assume that the proximal operators of both f and g are simple to compute, but the proximal operator of $f \circ K$ is not easily accessible. To tackle this issue, we consider the *primal dual* formulation of problem (87). We first detail the corresponding saddle point problem and then present algorithms dedicated to its resolution.

5.2.1 Saddle-point problem

If f is convex proper and l.s.c, we use conjugation

$$\forall z \in F, \quad f(z) = \sup_{y \in F} (\langle y, z \rangle - f^*(y)) \tag{88}$$

to rewrite the term $f(Kx)$ as

$$\forall x \in E, \quad f(Kx) = \sup_{y \in F} (\langle y, Kx \rangle - f^*(y)) = \sup_{y \in F} (\langle K^*y, x \rangle - f^*(y)), \tag{89}$$

where K^* denotes the adjoint operator of K defined from F to E .

Introducing this reformulation into the initial problem (87), we obtain the primal-dual problem

$$\min_{x \in E} \sup_{y \in F} \langle K^*y, x \rangle - f^*(y) + g(x), \quad (90)$$

defined for the primal unknown x and the dual variable y .

In the following, we assume that the above supremum is a maximum. As soon as the subdifferential of f at point z is non empty, the supremum is reached at a point y of the subdifferential $\partial f(z)$ and the supremum is indeed a maximum. We refer the reader to the book of R. Rockafellar [40] for further details. Having made this assumption, we return to the following **saddle point** problem:

$$\min_{x \in E} \max_{y \in F} h(x, y) := \langle K^*y, x \rangle - f^*(y) + g(x), \quad (91)$$

whose saddle-points (x^*, y^*) provide minimizers x^* of problem (87). In the following we assume that such saddle-points exist. Problems (87) and (91) are equivalent in the following sense: if (x, y) solves (91) then x is a minimizer of problem (87).

Proposition 15. *Let f and g be convex and proper functions respectively defined from F and E to \mathbb{R} , and K be an operator from E to F . If (x^*, y^*) is a saddle point of (91), then x^* is a solution of the primal problem (87). Moreover, the saddle points (x^*, y^*) of (91) satisfy*

$$\begin{cases} \forall \tau > 0, & x^* = \text{prox}_{\tau g}(x^* - \tau K^*y^*) \\ \forall \sigma > 0, & y^* = \text{prox}_{\sigma f^*}(y^* + \sigma Kx^*). \end{cases} \quad (92)$$

Proof. Let us recall that h is the function defined over $E \times F$ by

$$h(x, y) = \langle Kx, y \rangle + g(x) - f^*(y). \quad (93)$$

For $(x^*, y^*) \in E \times F$, we introduce h_1 and h_2 the partial functions respectively defined on E and F by $h_1(x) = h(x, y^*)$ and $h_2(y) = h(x^*, y)$. For any x^* and y^* , h_1 is convex in x and h_2 is concave in y (i.e. $-h_2$ is convex).

Hence, one can observe that

$$\begin{aligned} (x^*, y^*) \text{ is a saddle-point of } h &\Leftrightarrow \begin{cases} \forall x \in E, & h(x^*, y^*) \leq h(x, y^*) \\ \forall y \in F, & h(x^*, y) \leq h(x^*, y^*) \end{cases} \\ &\Leftrightarrow \begin{cases} \forall x \in E, & h_1(x^*) \leq h_1(x) \\ \forall y \in F, & -h_2(y^*) \leq -h_2(y) \end{cases} \\ &\Leftrightarrow \begin{cases} 0 \in \partial h_1(x^*) \\ 0 \in \partial h_2(y^*) \end{cases} \\ &\Leftrightarrow \begin{cases} 0 \in (K^*y^* + \partial g(x^*)) \\ 0 \in (-Kx^* + \partial f^*(y^*)) \end{cases} \\ &\Leftrightarrow \begin{cases} -K^*y^* \in \partial g(x^*) \\ Kx^* \in \partial f^*(y^*) \end{cases} \\ &\Leftrightarrow \begin{cases} \forall \tau > 0, & x^* = \text{prox}_{\tau g}(x^* - \tau K^*y^*) \\ \forall \sigma > 0, & y^* = \text{prox}_{\sigma f^*}(y^* + \sigma Kx^*). \end{cases} \end{aligned} \quad (94)$$

To conclude on the fact that x^* is a minimizer of problem (87), we first provide the following lemma.

Lemma 11. *Let f be a convex and close function defined from F to \mathbb{R} and K be an operator from E to F , then $\forall y \in \partial f(Kx)$, $K^*y \in \partial(f \circ K)(x)$.*

Proof. Let $y \in \partial f(Kx)$, then from definition (39), we have $\forall x' \in E$:

$$\begin{aligned} f(Kx') &\geq \langle y, Kx' - Kx \rangle + f(Kx) \\ (f \circ K)(x') &\geq \langle K^*y, x' - x \rangle + (f \circ K)(x), \end{aligned}$$

which is also equivalent to $K^*y \in \partial(f \circ K)(x)$ from definition (39). \square

Using relation (85) to switch from f^* to f , as well as the previous lemma, we obtain from (94):

$$\begin{aligned} &\exists y^* \in F \text{ s. t. } \begin{cases} -K^*y^* &\in \partial g(x^*) \\ Kx^* &\in \partial f^*(y^*) \end{cases} \\ \Leftrightarrow &\exists y^* \in F \text{ s. t. } \begin{cases} -K^*y^* &\in \partial g(x^*) \\ y^* &\in \partial f(Kx^*) \end{cases} \\ \Rightarrow &\exists y^* \in F \text{ s. t. } \begin{cases} -K^*y^* &\in \partial g(x^*) \\ K^*y^* &\in \partial(f \circ K)(x^*) \end{cases} \\ \Leftrightarrow &0 \in \partial(f \circ K)(x^*) + \partial g(x^*) \\ \Leftrightarrow &x^* \in \operatorname{argmin}_{x \in E} f(Kx) + g(x). \end{aligned}$$

\square

As a consequence from Proposition (15), algorithms dedicated to primal-dual problems are looking for pairs of variables (x, y) satisfying relations (92). In particular, the Arrow-Hurwicz algorithm computes, for a given $x_0 \in E$, the following pair of sequences $(x_n)_{n \in \mathbb{N}}$ and $(y_n)_{n \in \mathbb{N}}$:

$$\begin{cases} y_{n+1} &= \operatorname{prox}_{\sigma f^*}(y_n + \sigma Kx_n) \\ x_{n+1} &= \operatorname{prox}_{\tau g}(x_n - \tau K^*y_{n+1}). \end{cases} \quad (95)$$

The proof of convergence of this algorithm to a saddle-point of (91) requires additional assumptions, such as the strong convexity of f or g to make the proximal operators contractions (see Proposition 10). We now focus on generalizations of this algorithm that involve minimal conditions.

5.2.2 Chambolle-Pock algorithm (CP)

Before presenting the algorithm proposed by Chambolle and Pock in [12], we first give a fundamental property of saddle-points of problem (91), by introducing the *partial primal-dual gap*:

$$\mathcal{G}_{B_1 \times B_2}(x, y) = \max_{y' \in B_2} \langle y', Kx \rangle - f^*(y') + g(x) - \min_{x' \in B_1} \langle y, Kx' \rangle + g(x') - f^*(y), \quad (96)$$

that is defined on bounded sets $(B_1 \times B_2) \subset E \times F$.

Proposition 16. *Let $(x^*, y^*) \in B_1 \times B_2$ be a solution to problem (91), then, $\forall (x, y) \in B_1 \times B_2$, we have $\mathcal{G}_{B_1 \times B_2}(x, y) \geq 0$, and this gap is zero if and only if (x, y) is a solution of (91).*

Proof. We recall that the function h is defined in (91) as $h(x, y) = \langle Kx, y \rangle + g(x) - f^*(y)$. A saddle point (x^*, y^*) of problem (91) thus satisfies: $h(x^*, y^*) = \min_{x' \in B_1} h(x', y^*) = \max_{y' \in B_2} h(x^*, y')$, which implies that

$$\mathcal{G}_{B_1 \times B_2}(x^*, y^*) = \max_{y' \in B_2} h(x^*, y') - \min_{x' \in B_1} h(x', y^*) = h(x^*, y^*) - h(x^*, y^*) = 0. \quad (97)$$

Next we have

$$\begin{aligned} \mathcal{G}_{B_1 \times B_2}(x, y) &= \max_{y' \in B_2} h(x, y') - \min_{x' \in B_1} h(x', y) \\ &\geq h(x, y^*) - h(x^*, y) \\ &\geq h(x, y^*) - h(x^*, y^*) + h(x^*, y^*) - h(x^*, y) \\ &\geq h(x, y^*) - \min_{x' \in B_1} h(x', y^*) + \max_{y' \in B_2} h(x^*, y') - h(x^*, y) \\ &\geq 0. \end{aligned} \quad (98)$$

□

We can now present the Chambolle-Pock algorithm [12], that is based on a modification of the scheme (95) with a nonconvex update of x_n .

Theorem 14 (Chambolle-Pock algorithm). *Let f and g be two convex, proper, lower semi-continuous functions, with f defined from F to \mathbb{R} , g defined from E to \mathbb{R} and K a linear operator from E to F . Assume that the problem (91) admits a solution (x^*, y^*) . We denote $L = \|K\| = \sqrt{\|K^*K\|}$, and take σ and τ such that $\tau\sigma L^2 < 1$. We choose $(x_0, y_0) \in E \times F$ and set $\bar{x}_0 = x_0$ and define the sequences $(x_n)_{n \in \mathbb{N}}$, $(y_n)_{n \in \mathbb{N}}$, and $(\bar{x}_n)_{n \in \mathbb{N}}$ by*

$$\begin{cases} y_{n+1} &= \text{prox}_{\sigma f^*}(y_n + \sigma K \bar{x}_n) \\ x_{n+1} &= \text{prox}_{\tau g}(x_n - \tau K^* y_{n+1}) \\ \bar{x}_{n+1} &= 2x_{n+1} - x_n. \end{cases} \quad (99)$$

Setting $x^N = (\sum_{n=1}^N x_n)/N$ and $y^N = (\sum_{n=1}^N y_n)/N$, then for each bounded set $(B_1 \times B_2) \subset E \times F$, the partial primal-dual gap satisfies

$$0 \leq \mathcal{G}_{B_1 \times B_2}(x^N, y^N) \leq \frac{1}{N} \left(\frac{\|x_0 - x^*\|^2}{2\tau} + \frac{\|y_0 - y^*\|^2}{2\sigma} - \langle K(x_0 - x^*), y_0 - y^* \rangle \right).$$

This theorem illustrates the ergodic convergence rate of the primal-dual algorithm [13]. For completeness, we provide in Appendix A the proof of [12], that shows the convergence of the iterates (x_n, y_n) of algorithm (99) to a saddle-point (x^*, y^*) of problem (91).

Proof. This demonstration is based on the works of [24, 32]. Let us first switch the step order in algorithm (99):

$$\begin{cases} x_{n+1} &= \text{prox}_{\tau g}(x_n - \tau K^* y_{n+1}) \\ \bar{x}_{n+1} &= 2x_{n+1} - x_n \\ y_{n+1} &= \text{prox}_{\sigma f^*}(y_n + \sigma K \bar{x}_{n+1}). \end{cases} \quad (100)$$

Using the characterization of proximal operators in Proposition 6, we have:

$$\begin{aligned} & \begin{cases} -\left(\frac{x_{n+1}-x_n}{\tau} + K^* y_n\right) & \in \partial g(x_{n+1}) \\ -\left(\frac{y_{n+1}-y_n}{\sigma} - K(2x_{n+1} - x_n)\right) & \in \partial f^*(x_{n+1}) \end{cases} \\ \Leftrightarrow 0 & \in \begin{cases} \frac{x_{n+1}-x_n}{\tau} + K^* y_n + \partial g(x_{n+1}) \\ \frac{y_{n+1}-y_n}{\sigma} - K(2x_{n+1} - x_n) + \partial f^*(x_{n+1}) \end{cases} \\ \Leftrightarrow 0 & \in \begin{bmatrix} \frac{\text{Id}}{\tau} & 0 \\ 0 & \text{Id} \end{bmatrix} \begin{bmatrix} x_{n+1} - x_n \\ y_{n+1} - y_n \end{bmatrix} - \begin{bmatrix} 0 & K^* \\ K & 0 \end{bmatrix} \begin{bmatrix} x_{n+1} - x_n \\ y_{n+1} - y_n \end{bmatrix} + \begin{bmatrix} \partial g & K^* \\ -K & \partial f^* \end{bmatrix} \begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} \\ \Leftrightarrow 0 & \in \begin{bmatrix} \frac{\text{Id}}{\tau} & -K^* \\ -K & \frac{\text{Id}}{\sigma} \end{bmatrix} \begin{bmatrix} x_{n+1} - x_n \\ y_{n+1} - y_n \end{bmatrix} + \begin{bmatrix} \partial g & K^* \\ -K & \partial f^* \end{bmatrix} \begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix}. \end{aligned} \quad (101)$$

Recalling that

$$h(x, y) = \langle Kx, y \rangle + g(x) - f^*(y),$$

and introducing the matrix

$$M = \begin{bmatrix} \frac{\text{Id}}{\tau} & -K^* \\ -K & \frac{\text{Id}}{\sigma} \end{bmatrix},$$

relation (101) can be written as

$$M \begin{bmatrix} x_n - x_{n+1} \\ y_n - y_{n+1} \end{bmatrix} \in \begin{bmatrix} \partial_x h(x_{n+1}, y_{n+1}) \\ -\partial_y h(x_{n+1}, y_{n+1}) \end{bmatrix}. \quad (102)$$

In the following, we consider $\sigma\tau\|K^*K\| = \sigma\tau L^2 < 1$ to ensure that the matrix M is positive-definite. Since h is convex (resp. concave) with respect to x (resp. y), we get from the definition of subdifferential (12) that:

$$\begin{aligned} h(x_{n+1}, y_{n+1}) + \langle \partial_x h(x_{n+1}, y_{n+1}), x - x_{n+1} \rangle &\leq h(x, y_{n+1}) \\ -h(x_{n+1}, y_{n+1}) - \langle \partial_y h(x_{n+1}, y_{n+1}), y - y_{n+1} \rangle &\leq -h(x_{n+1}, y). \end{aligned}$$

Summing both expressions and introducing $w_{n+1} = M(z_n - z_{n+1})$, we get from relation (102)

$$\begin{aligned} h(x_{n+1}, y) - h(x, y_{n+1}) &\leq \langle w_{n+1}, z_{n+1} - z \rangle = \langle z_n - z_{n+1}, z_{n+1} - z \rangle_M \\ &\leq \frac{1}{2} (\|z_n - z\|_M^2 - \|z_{n+1} - z\|_M^2 - \|z_{n+1} - z_n\|_M^2) \\ &\leq \frac{1}{2} (\|z_n - z\|_M^2 - \|z_{n+1} - z\|_M^2), \end{aligned}$$

where we used the fact that $\|\cdot\|_M^2 = \langle M, \cdot \rangle$ is a semi-norm since M is positive-definite. Summing the previous expression from $n = 1$ to N and introducing $x^N = (\sum_{n=1}^N x_n)/N$ and $y^N = (\sum_{n=1}^N y_n)/N$, we obtain:

$$h(x^N, y) - h(x, y^N) \leq \frac{1}{N} \sum_{n=1}^N (h(x_n, y) - h(x, y_n)) \leq \frac{1}{2N} \|z_0 - z\|_M^2.$$

Considering previous expression for a saddle point $z^* = (x^*, y^*)$, we get from Proposition 16 that

$$0 \leq \mathcal{G}_{B_1 \times B_2}(x^N, y^N) \leq \frac{1}{2N} \|z_0 - z^*\|_M^2. \quad \square$$

Remark 11. *The work in [24] offers an alternative to the proof of convergence of the primal-dual algorithm provided in Appendix A. The convergence of primal-dual can indeed be obtained from the fact that the scheme (101) is an instance of the proximal point algorithm of [41] applied to the variable $z = (x, y)$ in which $\begin{bmatrix} \partial g & K^* \\ -K & \partial f^* \end{bmatrix}$ corresponds to a maximal monotone operator (a monotone operator $T : \mathcal{H} \rightarrow \mathcal{H}$ satisfies $\langle T(z) - T(z'), z - z' \rangle \geq 0$ for all $z, z' \in \mathcal{H}$ and it is maximal if and only if $\text{range}(T + \text{Id}) = \mathcal{H}$, with $\text{range}(T) = \{z \in \mathcal{H}, \text{ such that } \exists w \in \mathcal{H}, \text{ satisfying } z = T(w)\}$). The iterates $z_n = (x_n, y_n)$ then converge if the matrix M is positive-definite, which is the case as soon as $\sigma\tau\|K^*K\| = \sigma\tau L^2 < 1$.*

5.2.3 Condat algorithm

The Condat algorithm generalizes the primal dual one to the minimization of problems involving more convex functions:

$$\min_{x \in E} f(x) + g(x) + \sum_{i=1}^M h_i(L_i x). \quad (103)$$

where f is a convex differentiable function whose gradient is L -Lipschitz and the functions g and h_i are convex, and L_k are linear operators. For any initial points $x_0 \in E$ and $(u_0^i)_{i \leq M}$ initial dual variables, the Condat algorithm introduced in [16] defines the sequences $(\tilde{x}_n)_{n \in \mathbb{N}}$, $(x_n)_{n \in \mathbb{N}}$, $(\tilde{u}_n)_{n \in \mathbb{N}}$, $(u_n)_{n \in \mathbb{N}}$ as

$$\begin{cases} \tilde{x}_{n+1} &= \text{prox}_{\tau g}(x_n - \tau \nabla f(x_n) - \tau \sum_{i=1}^M L_m^* u_n^i) \\ x_{n+1} &= \rho \tilde{x}_{n+1} + (1 - \rho)x_n \\ \tilde{u}_{n+1}^i &= \text{prox}_{\sigma h_i^*}(u_k^i + \sigma L_i(2\tilde{x}_{n+1} - x_n)) & \text{for } i = 1 \dots M \\ u_{n+1}^i &= \rho \tilde{u}_{n+1}^i + (1 - \rho)u_k^i & \text{for } i = 1 \dots M. \end{cases} \quad (104)$$

For $\rho \in (0, 1]$ and $\sigma > 0$, $\tau > 0$ such that

$$\tau \left(\frac{L}{2} + \sigma \left\| \left\| \sum_{i=1}^M L_i^* L_i \right\| \right\| \right) < 1, \quad (105)$$

it is shown in [16] (the proof is not reproduced here) that the sequence $(x_n)_{n \in \mathbb{N}}$ defined in (104) weakly converges to a minimizer of (103).

The Condat algorithm have the following properties:

- It allows to use an explicit gradient descend on the differentiable part, as the Forward-Backward algorithm;
- It separates functions and operators, as done in the Chambolle-Pock algorithm;
- The computations of $(u_n^i)_{i \leq M}$ are independent and can be done separately and allow parallel computing as with PPXA.

5.3 Equivalence between proximal splitting algorithms

We now show that in particular cases, some proximal splitting algorithms are equivalent. To that end, we consider the general problem

$$\min_{x \in E} f(Kx) + g(x), \quad (106)$$

where the linear operator K is defined from E to F . For convex proper and l.s.c functions f and g , this problem can be solved with the Douglas-Rachford algorithm (70):

$$\begin{cases} w_{n+1} &= w_n + \text{prox}_{\gamma f \circ K}(2v_n - w_n) - v_n, \\ v_{n+1} &= \text{prox}_{\gamma g}(w_{n+1}). \end{cases} \quad (107)$$

We now detail the equivalence of this algorithm with the Chambolle-Pock and the ADMM ones.

5.3.1 Chambolle-Pock and Douglas-Rachford equivalence

We show the equivalence between Chambolle-Pock and Douglas-Rachford algorithms when the linear operator is the identity.

Proposition 17. *Considering the primal problem (106), in the case $K = Id$ and setting*

$$\sigma = \frac{1}{\gamma} \text{ and } \tau = \gamma, \quad (108)$$

the Douglas-Rachford iterations (107) are recovered from the Chambolle-Pock iterations (99) using

$$\begin{cases} x_{n+1} &= v_{n+1}, \\ \gamma y_{n+1} &= v_n - w_{n+1}. \end{cases} \quad (109)$$

Proof. The Chambolle-Pock algorithm with the above parameter reads:

$$\begin{cases} y_{n+1} &= \text{prox}_{f^*/\gamma}(y_n + \bar{x}_n/\gamma) = y_n + \bar{x}_n/\gamma - \frac{1}{\gamma} \text{prox}_{\gamma f}(\gamma y_n + \bar{x}_n) \\ x_{n+1} &= \text{prox}_{\tau g}(x_n - \gamma y_{n+1}), \\ \bar{x}_{n+1} &= 2x_{n+1} - x_n, \end{cases} \\ \Leftrightarrow \begin{cases} -\gamma y_{n+1} + \gamma y_n + 2x_n - x_{n-1} &= \text{prox}_{\gamma f}(\gamma y_n + 2x_n - x_{n-1}) \\ x_{n+1} &= \text{prox}_{\tau g}(x_n - \gamma y_{n+1}) \end{cases}$$

Incorporating the change of variables (109), we obtain

$$\begin{cases} w_{n+1} - w_n + v_n &= \text{prox}_{\gamma f}(2v_n - w_n) \\ v_{n+1} &= \text{prox}_{\tau g}(w_{n+1}), \end{cases}$$

that corresponds to the Douglas-Rachford iterations (107). \square

5.3.2 ADMM and Douglas-Rachford equivalence

To show the equivalence between Douglas-Rachford and ADMM, we consider the Fenchel-Rockafeller dual to the primal problem (106) that writes:

$$\max_{z \in F} -(g^*(-K^*z) + f^*(z)) \Leftrightarrow \min_{z \in F} f^*(z) + g^*(-K^*z), \quad (110)$$

where we assume that the adjoint operator K^* defined from F to E is injective. This problem can be reformulated as

$$\min_{(x,y) \in F \times E, K^*x+y=0} f^*(x) + g^*(y). \quad (111)$$

This corresponds to a particular form of the problem (76) for the two convex conjugate f^* and g^* defined on F and E , the operators $A = K^*$, $B = \text{Id}$ and the vector $b = 0$. It can be solved with the Alternating Direction Method of Multipliers (ADMM) algorithm presented in (79):

$$\begin{cases} x_{n+1} &= \underset{x}{\text{argmin}} f^*(x) + \langle z_n, K^*x \rangle + \frac{\gamma}{2} \|y_n + K^*x\|^2 \\ y_{n+1} &= \underset{y}{\text{argmin}} g^*(y) + \langle z_n, y \rangle + \frac{\gamma}{2} \|K^*x_{n+1} + y\|^2 \\ z_{n+1} &= z_n + \gamma(y_{n+1} + K^*x_{n+1}). \end{cases} \quad (112)$$

Introducing the proximal operator $\text{prox}_{\gamma f}^A$ with a metric induced by an injective map A

$$\text{prox}_{\gamma f}^A(x) = \underset{s}{\text{argmin}} \frac{1}{2\gamma} \|As - x\|^2 + f(s). \quad (113)$$

the ADMM iterations (112) can be rewritten using proximal maps

$$\begin{cases} x_{n+1} &= \text{prox}_{f^*/\gamma}^{K^*}(-y_n - z_n/\gamma), \\ y_{n+1} &= \text{prox}_{g^*/\gamma}(-K^*x_{n+1} - z_n/\gamma), \\ z_{n+1} &= z_n + \gamma(y_{n+1} + K^*x_{n+1}). \end{cases} \quad (114)$$

The following proposition, which was initially proved in [20], shows that applying the ADMM algorithm to the dual problem (110) is equivalent to solving the primal (106) using the Douglas-Rachford algorithm. This result was further extended by [19] which shows the equivalence of ADMM with the proximal point algorithm and propose several generalizations.

Proposition 18. *For f and g convex, the Douglas-Rachford iterations (107) are recovered from the ADMM iterations (114) using*

$$\begin{cases} z_n &= -v_n, \\ \gamma y_n &= w_n - v_n, \\ -\gamma K^*x_{n+1} &= w_{n+1} - v_n. \end{cases} \quad (115)$$

To show this proposition, we first require the following result [36].

Proposition 19. *One has*

$$\text{prox}_{f/\gamma}^A(x) = A^+ \left(x - \frac{1}{\gamma} \text{prox}_{\gamma f^* \circ A^*}(\gamma x) \right), \quad (116)$$

where $A^+ = (A^*A)^{-1}A^*$ is the pseudo-inverse of A . Note that in the case $A = \text{Id}$, one recovers Moreau's identity (86).

Proof. We denote $\mathcal{U} = \partial f$ the set-valued maximal monotone operator. One has $\partial f^* = \mathcal{U}^{-1}$, where \mathcal{U}^{-1} is the set-valued inverse operator. We thus have $\partial(f^* \circ A^*) = A \circ \mathcal{U}^{-1} \circ A^* = \mathcal{V}$ and $\text{prox}_{\gamma f^* \circ A^*} = (\text{Id} + \gamma \mathcal{V})^{-1}$, which is a single-valued operator. Denoting $x^* = \text{prox}_{f/\gamma}^A(x)$, the optimality condition of (113) leads to

$$\begin{aligned} 0 &\in A^*(Ax^* - x) + \frac{1}{\gamma} \mathcal{U}(x^*) \Leftrightarrow x^* \in \mathcal{U}^{-1}(\gamma A^* Ax^* - \gamma A^* x) \\ &\Leftrightarrow \gamma Ax^* \in \gamma \mathcal{V}(\gamma x - \gamma Ax^*) \Leftrightarrow \gamma Ax^* \in (\text{Id} + \gamma \mathcal{V})(\gamma x - \gamma Ax^*) + \gamma Ax^* - \gamma x \\ &\Leftrightarrow \gamma x \in (\text{Id} + \gamma \mathcal{V})(\gamma x - \gamma Ax^*) \Leftrightarrow \gamma x - \gamma Ax^* = (\text{Id} + \gamma \mathcal{V})^{-1}(\gamma x) \\ &\Leftrightarrow x^* = A^+ \left(x - \frac{1}{\gamma} (\text{Id} + \gamma \mathcal{V})^{-1}(\gamma x) \right) \end{aligned}$$

where we used the fact that A is injective. \square

Proof of Proposition 18. Denoting $\bar{x}_n = -K^*x_n \in \text{Im}(-K^*)$ (recall that K^* is injective) and using the result of Proposition 19, the iterates (114) write

$$\begin{cases} \bar{x}_{n+1} &= y_n + z_n/\gamma + \frac{1}{\gamma} \text{prox}_{\gamma f \circ K}(-\gamma y_n - z_n), \\ y_{n+1} &= \bar{x}_{n+1} - z_n/\gamma - \frac{1}{\gamma} \text{prox}_{\gamma g}(\gamma \bar{x}_{n+1} - z_n), \\ z_{n+1} &= z_n + \gamma(y_{n+1} - \bar{x}_{n+1}), \end{cases} \quad (117)$$

so that

$$\begin{cases} -z_n + \gamma(\bar{x}_{n+1} - y_n) &= \text{prox}_{\gamma f \circ K}(-\gamma y_n - z_n), \\ -z_n + \gamma(\bar{x}_{n+1} - y_{n+1}) &= \text{prox}_{\gamma g}(\gamma \bar{x}_{n+1} - z_n), \\ z_{n+1} &= z_n + \gamma(y_{n+1} - \bar{x}_{n+1}). \end{cases} \quad (118)$$

Considering the following change of variables (115):

$$\begin{cases} z_n &= -v_n, \\ \gamma y_n &= w_n - v_n, \\ \gamma \bar{x}_{n+1} &= w_{n+1} - v_n, \end{cases}$$

relations (118) become

$$\begin{cases} w_{n+1} &= w_n + \text{prox}_{\gamma f \circ K}(2v_n - w_n) - v_n, \\ v_{n+1} &= \text{prox}_{\gamma g}(w_{n+1}), \\ 0 &= 0, \end{cases} \quad (119)$$

that gives the Douglas-Rachford iterations (107). \square

Remark 12. *For $K = \text{Id}$, the previous result remains true for f and g nonconvex functions. Indeed, as already mentioned in Section 4.3.6, for $K = \text{Id}$, there is a primal equivalence between ADMM (80) and DRS (71).*

6 Extension to nonconvex optimization

We now study the application of previously introduced gradient based algorithms to the optimization of nonconvex functionals. Without convexity, a lot of issues arise

- A local minimum is no longer necessarily global. The convergence to a (local) minimum of the nonconvex function is not ensured. Iterative optimization algorithms rather target the convergence to critical points of the function.
- The subdifferential may be reduced to an empty set at many points.
- The proximal operator may be undefined or multivalued.
- The identity between a function f and its biconjugate f^{**} is no longer valid and can not be exploited in primal-dual formulations.

Despite these limits, we can still prove some convergence results. In particular, by finding decreasing Lyapunov functions, we can show convergence to zero of the norm between consecutive iterates. Then, assuming additional regularity on the optimized function around its critical points, such as the Kurdyka-Łojasewicz property, one can prove convergence of the iterates towards a critical point of the function.

In this section, we first provide preliminary results on the convergence of the gradient descent scheme in the nonconvex setting in section 6.1. Next a detailed analysis of the Forward-Backward algorithm for nonconvex problems is given in section 6.2. In section 6.3, we present existing results relative to the use of Douglas-Rachford, ADMM and Chambolle-Pock algorithms in the nonconvex setting.

6.1 From convex to nonconvex optimization, the case of gradient descent

A part of the convergence results that were proved in the convex setting are still valid on the nonconvex setting. As in Section 3, we consider

$$\min_{x \in E} f(x) \tag{120}$$

with a nonconvex, L -smooth function $f : E \subseteq \mathbb{R}^d \rightarrow \mathbb{R} \cup \{+\infty\}$. The main difference with the convex case is that for minimizing a nonconvex function, there is usually no hope to target a global minimum. The gradient of f is still defined and we instead look for a stationary point of f , i.e. a point x such that $\nabla f(x) = 0$. Notice that in the nonconvex case, such a stationary point is not necessarily a global minimum.

We consider again the gradient-descent algorithm with stepsize $\gamma > 0$,

$$x_{n+1} = x_n - \gamma \nabla f(x_n). \tag{121}$$

In the proof of convergence of the gradient-descent algorithm in the convex setting in Theorem 1, without making use of the convexity of f and using the fact that f is L -smooth, we proved the following sufficient decrease property:

$$f(x_{n+1}) + \left(\frac{2 - \gamma L}{2\gamma} \right) \|x_{n+1} - x_n\|^2 \leq f(x_n). \tag{122}$$

Therefore, as soon as $\gamma < \frac{2}{L}$, the function f decreases along the iterates. Assuming that f is lower-bounded, the sequence of function values $f(x_n)$ then converges. Moreover, by summing (122), between $n = 0$ and $n = N > 0$, we get that $\|x_{n+1} - x_n\| = \|\nabla f(x_n)\|$ converges to 0 with rate $\mathcal{O}(1/\sqrt{N})$. Eventually, assuming f of class \mathcal{C}^1 , we get that any cluster point x^* of the sequence (x_n) verifies $\nabla f(x^*) = 0$, i.e. x^* is a critical points of the objective function f . These results are summarized in the following Proposition.

Proposition 20 (Nonconvex Gradient Descent). *Assume f proper, lsc, bounded from below and L -smooth. Then, for $\gamma < 2/L$, the iterates (x_n) given by the Gradient Descent algorithm (121) verify*

(i) $f(x_n)$ is non-increasing and converges.

(ii) The sequence $\|x_{n+1} - x_n\|$ converges to 0 at rate $\min_{n < N} \|x_{n+1} - x_n\| = \mathcal{O}(1/\sqrt{N})$

(iii) All cluster points of the sequence (x_n) are critical points of f .

It is important to note that this result does not guarantee the convergence of the sequence $(x_n)_{n \in \mathbb{N}}$ itself, which could infinitely turn around a level set of $f + g$ that is not a local minima. In the following section, we will explain that under an additional assumption on the geometry of the objective function around its critical points, we can prove single-point convergence of the sequence (x_n) towards a critical point of f .

6.2 Single-point nonconvex convergence of Forward-Backward

Let's consider again the more general problem

$$\min_{x \in E} f(x) + g(x) \tag{123}$$

under the main assumption of a L -smooth function f . Note that even if the functions f and/or g are nonconvex, the following Forward-Backward algorithm is well defined:

$$x_{n+1} \in \text{prox}_{\gamma g}(x_n - \gamma \nabla f(x_n)). \tag{124}$$

Note that, due to the nonconvexity of g , the output of the proximal operator may not be unique. We consider this algorithm because it is more general than the Gradient Descent algorithm. In particular, it allows to treat non-differentiable term g .

In this section, we study the point-wise convergence of the iterates of the Forward-Backward algorithm (124) in the nonconvex setting. As the objective function is not necessarily differentiable, we look for a zero of its subdifferential. We first need to define the right notion of nonconvex subdifferential. Then, we introduce the Kurdyka-Łojasiewicz property, a local property that characterizes the shape of a function around its critical points. Finally, we prove iterate convergence of the Forward-Backward algorithm (124) under this property.

6.2.1 Nonconvex subdifferential and critical points

We first define the notion of optimum for the nonconvex problem (123). For a nonconvex function f , the usual notion of subdifferential from Definition 12 may not be informative enough. Indeed, with this notion, with Fermat's rule (Theorem 7), the critical points of f , i.e. the zeros of the subdifferential ∂f corresponds to the minimizers of f . Therefore, targeting a critical point comes back to looking for a global minimum. We explained that this is out of reach when minimizing a general nonconvex function. Instead, one can hope to reach a point that is critical for some generalized notion of subdifferential.

To study the convergence of proximal iterative schemes for minimizing a nonconvex function, an adequate notion of subdifferential [2] is the *limiting subdifferential*, also called *general subgradient* in [42]:

$$\partial^{lim} f(x) = \left\{ \omega \in E, \exists x_n \rightarrow x, f(x_n) \rightarrow f(x), \omega_n \rightarrow \omega, \omega_n \in \hat{\partial} f(x_n) \right\} \quad (125)$$

with $\hat{\partial} f$ the *Fréchet subdifferential* of f (also called *regular subgradient* in [42]) defined as

$$\hat{\partial} f(x) = \left\{ \omega \in E, \liminf_{y \rightarrow x} \frac{f(y) - f(x) - \langle \omega, y - x \rangle}{\|x - y\|} \geq 0 \right\}. \quad (126)$$

The three introduced notions of subdifferential verify for $x \in \text{dom } f$

$$\partial f(x) \subset \hat{\partial} f(x) \subset \partial^{lim} f(x). \quad (127)$$

For f *convex*, the three subdifferentials coincide [42, Proposition 8.12]. For f of class \mathcal{C}^1 they also coincide with the usual concept of gradient $\hat{\partial} f(x) = \partial^{lim} f(x) = \{\nabla f(x)\}$. This generalized notion of subdifferential gives birth to generalized notions of *critical point* or *stationary point* i.e. $x \in E$ such that

$$0 \in \partial^{lim} f(x). \quad (128)$$

A necessary (but not sufficient) condition for $x \in E$ to be a local minimizer of a nonconvex function f is $0 \in \hat{\partial} f(x)$ and thus $0 \in \partial^{lim} f(x)$. Last but not least, the limiting subdifferential also verifies the sum rule.

Proposition 21 (Sum rule [42, 8.8(c)]). *If $\mathcal{J} = f + g$ with f of class \mathcal{C}^1 and g proper. Then for $x \in \text{dom}(g)$,*

$$\partial^{lim} \mathcal{J}(x) = \nabla f(x) + \partial^{lim} g(x). \quad (129)$$

In particular, thanks to this property, a fixed point of the Forward-Backward algorithm (124) is a critical point of $f + g$

$$x^* \in \text{prox}_{\gamma g}(x^* - \gamma \nabla f(x^*)) \Rightarrow 0 \in \gamma \partial^{lim} g(x^*) + (x^* - (x^* - \gamma \nabla f(x^*))) \quad (130)$$

$$\Rightarrow -\nabla f(x^*) \in \partial^{lim} g(x^*) \quad (131)$$

$$\Rightarrow 0 \in \partial^{lim}(f + g)(x^*) \quad (132)$$

For additional details on the notion of limiting subdifferential, we refer to [42, Chapter 8]. In the rest of this section, we will denote for simplicity ∂f as the limiting subdifferential $\partial^{lim} f$.

6.2.2 Kurdyka-Łojasiewicz (KL) property

To ensure the single-point convergence of the Forward-Backward algorithm towards a critical point of $\mathcal{J} = f + g$ (i.e. a zero of the limiting subdifferential of \mathcal{J}), we introduce the Kurdyka-Łojasiewicz (KL) property [8]. It is a local property that characterizes the shape of the function \mathcal{J} around its critical points. Before entering into technical details, we first present a simpler condition given by Łojasiewicz for smooth functions [30, 31].

Definition 17 (Łojasiewicz condition). *Let $f : E \rightarrow \mathbb{R} \cup \{+\infty\}$. We say that the function f satisfies the Łojasiewicz condition if, for any critical point x^* of f , there exists a neighborhood V of x^* , $C > 0$, and $\alpha \in [0, 1[$ such that for all $x \in V$*

$$\|\nabla f(x)\| \geq C|f(x) - f(x^*)|^\alpha. \quad (133)$$

For a smooth function f , this condition is verified at points x such that $\nabla f(x) \neq 0$. The Łojasiewicz condition thus ensures that a function is not too flat around its critical points. It is an essential property to ensure convergence of optimization algorithms in the nonconvex setting. The Łojasiewicz property is verified in particular by *real analytic functions*.

The Kurdyka-Łojasiewicz condition [28] generalizes the previous Łojasiewicz condition to nonsmooth functions (see also [2] for more details).

Definition 18 (Kurdyka-Łojasiewicz property). *The function $f : E \rightarrow \mathbb{R} \cup \{+\infty\}$ satisfies the Kurdyka-Łojasiewicz property at point $x^* \in \text{dom } \partial f$ if there exists $\eta > 0$, a neighborhood V of x^* and a function $\varphi : [0, \eta[\rightarrow \mathbb{R}_+$ such that:*

1. $\varphi(0) = 0$,
2. φ is C^1 on $]0, \eta[$,
3. For all $s \in]0, \eta[$, $\varphi(s) > 0$,
4. For all $x \in V$ such that $f(x) \in]f(x^*), f(x^*) + \eta[$, the KL property is verified:

$$\varphi'(f(x) - f(x^*)) \text{dist}(0, \partial f(x)) \geq 1. \quad (134)$$

Proper and l.s.c functions satisfying the KL property $\forall x \in \text{dom } \partial f$ are called KL functions.

The KL condition can be interpreted as the fact that, up to a reparameterization, the function is locally sharp around its critical points, i.e. we can bound its subgradients away from 0. As before, this condition is verified at all points x such that $0 \notin \partial f(x)$.

How to verify that a function verifies the Kurdyka-Łojasiewicz property?

In the next section, we will make use of the KL property to study the convergence of iterative algorithms for minimizing the sum of two functions $F = f + g$ with f or g nonconvex.

The first condition will be to have F KŁ. However, the KŁ condition is not stable by sum. Therefore, we need to introduce conditions on f and g such that $f+g$ is KŁ. Moreover, directly verifying that a function verifies the KŁ inequality (134) is not easy. We need to choose a subclass of KŁ functions that, on the one hand, is large enough to encompass most functions of interest and, on the other hand, has minimal stability properties so that inclusion to that set is easy to verify. *Semialgebraic functions* [18] is such a convenient class of functions and is used in most works on nonconvex optimization [1, 2, 9].

Definition 19 (Semialgebraic function).

- A subset S of \mathbb{R}^n is a real semialgebraic set if there exists a finite number of real polynomial functions $P_{i,j}, Q_{i,j} : \mathbb{R}^n \rightarrow \mathbb{R}$ such that

$$S = \cup_{j=1}^p \cap_{i=1}^q \{x \in \mathbb{R}^n, P_{i,j} = 0, Q_{i,j} < 0\}. \quad (135)$$

- A function $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$ (resp. $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$) is called semialgebraic if its graph $\{(x, y) \in \mathbb{R}^n \times \mathbb{R}, y = f(x)\}$ (resp. $\{(x, y) \in \mathbb{R}^n \times \mathbb{R}^m, y = f(x)\}$) is a semialgebraic subset of $\mathbb{R}^n \times \mathbb{R}$ (resp. $\mathbb{R}^n \times \mathbb{R}^m$).

A semialgebraic function satisfies the Kurdyka-Łojasiewicz property [8]. From Definition 19, we first verify that polynomial functions are semialgebraic functions. Some other typical semialgebraic maps are the indicator function of a semialgebraic set or the Euclidean norm. What makes the semialgebraic set of functions useful is that it has strong stability properties. The main ones follow from the *Tarski-Seidenberg principle*, which states that the image of a semialgebraic subset of \mathbb{R}^{n+m} by projection on the first n coordinates is a semialgebraic subset of \mathbb{R}^n . From this theorem, the sum, product, composition or derivative of a semialgebraic functions are semialgebraic. These stability properties are very useful to prove that a given function is KŁ.

Remark 13 (On the KŁ properties of neural networks). *One can prove that neural networks with ReLU activations are semialgebraic functions, and thus KŁ. Indeed neural networks basically consists of composition and sums of linear maps and activation functions. A linear map being semialgebraic, the main difficulty is to show that activation functions are semialgebraic. This is the case of the ReLU which can be expressed with a system of polynomial (in)equalities:*

for $x, y \in \mathbb{R}$, $y = \text{ReLU}(x) = \max(0, x)$ if and only if $y(y - x) = 0$, $y \geq x$, $y \geq 0$.

Nonconvex convergence to a critical point with Kurdyka-Łojasiewicz property We now explain why Kurdyka-Łojasiewicz functions are useful for convergence in the nonconvex setting. Given a KŁ function f , we want to show convergence of a sequence $(x_n)_{n \in \mathbb{N}}$, produced by a certain iterative procedure, towards a critical point of f . The following lemma from [2] gives sufficient conditions on (x_n) such that this convergence is verified. This comprehensive lemma will be the basis of the nonconvex convergence analysis of first-order optimization algorithms like Forward-Backward.

Lemma 12 ([2, Theorem 2.9]). *Let $f : E \rightarrow \mathbb{R} \cup \{+\infty\}$ be a proper lsc function. Consider a sequence $(x_n)_{n \in \mathbb{N}}$ satisfying the following conditions:*

- **H1: Sufficient decrease condition** $\forall n \in \mathbb{N}$,

$$f(x_{n+1}) + a\|x_{n+1} - x_n\|^2 \leq f(x_n). \quad (136)$$

- **H2: Relative error condition** $\forall n \in \mathbb{N}$, there exists $\omega_{n+1} \in \partial f(x_{n+1})$ such that

$$\|\omega_{n+1}\| \leq b\|x_{n+1} - x_n\|. \quad (137)$$

- **H3: Continuity condition** There exists a subsequence $(x_{k_i})_{i \in \mathbb{N}}$ and $\hat{x} \in E$ such that

$$x_{k_i} \rightarrow \hat{x} \text{ and } f(x_{k_i}) \rightarrow f(\hat{x}) \text{ as } i \rightarrow +\infty. \quad (138)$$

If f verifies the Kurdyka–Łojasiewicz property at the cluster point \hat{x} specified in H3, then (x_n) converges to \hat{x} as $k \rightarrow +\infty$ and \hat{x} is a critical point of f .

The first condition H1 represents the descent of the objective function f along the iterates. Note that if f is lower-bounded, with H1, as $f(x_k)$ decreases, we have convergence of the function values $f(x_n)$, as well as, by telescopic sum, $\sum \|x_{n+1} - x_n\|^2 < +\infty$. If f is also coercive, then $\{f(x) < f(x_0)\}$ is necessarily bounded and the iterates remain bounded. We can then extract from (x_n) a subsequence converging towards \hat{x} . With condition H2 and H3, we obtain $0 \in \partial f(\hat{x})$ i.e. the limit point is a critical point of f .

When dealing with a nonconvex function f , a sequence (x_n) satisfying the above conditions is not guaranteed to converge to a single point. It can be the case if f is flat or highly oscillating around its critical points. The KL property is a general and flexible condition that prevents the above cases and ensures single-point convergence of any sequence satisfying H1, H2 and H3.

6.2.3 Nonconvex Forward-Backward single-point convergence

We remind the Forward-Backward algorithm

$$x_{n+1} \in \text{prox}_{\gamma g}(x_n - \gamma \nabla f(x_n)). \quad (139)$$

We can use the abstract nonconvex convergence result of Lemma 12, with the Kurdyka–Łojasiewicz (KL) property, to prove that if $\mathcal{J} = f + g$ is KL and the sequence generated by the Forward-Backward algorithm is bounded, the sequence converges towards a critical point of the objective function.

Theorem 15 (Single point convergence of nonconvex Forward-Backward.). *Assume f and g proper, lsc, lower-bounded, with f L -smooth and $\mathcal{J} = f + g$ KL. Then, for $\gamma < 1/L$, if the iterates (x_n) given by the Forward-Backward algorithm (139) are bounded, then they converge towards a critical point of $\mathcal{J} = f + g$.*

Proof. This result is a direct application of the general nonconvex convergence result from Lemma 12. We need to verify its assumptions **H1**, **H2** and **H3**.

- **H1: Sufficient decrease condition**

We first reformulate the Forward-Backward iterates as

$$x_{n+1} \in \text{prox}_{\gamma g} \circ (x_n - \gamma \nabla f(x_n)) \quad (140)$$

$$\Leftrightarrow x_{n+1} \in \underset{x}{\text{argmin}} g(x) + \frac{1}{2\gamma} \|x - (x_n - \gamma \nabla f(x_n))\|^2 \quad (141)$$

$$\Leftrightarrow x_{n+1} \in \underset{x}{\text{argmin}} g(x) + \langle x - x_n, \nabla f(x_n) \rangle + \frac{1}{2\gamma} \|x - x_n\|^2. \quad (142)$$

Hence by evaluating the right-hand side at x_{n+1} and x_n , and adding $f(x_n)$ on both sides, we get

$$f(x_n) + g(x_{n+1}) + \langle x_{n+1} - x_n, \nabla f(x_n) \rangle + \frac{1}{2\gamma} \|x_{n+1} - x_n\|^2 \leq f(x_n) + g(x_n) = \mathcal{J}(x_n). \quad (143)$$

Then, using the descent lemma (Lemma 1), f being L -smooth, we obtain

$$\begin{aligned} & f(x_n) + \langle x_{n+1} - x_n, \nabla f(x_n) \rangle + \frac{1}{2\gamma} \|x_{n+1} - x_n\|^2 \\ &= f(x_n) + \langle x_{n+1} - x_n, \nabla f(x_n) \rangle + \frac{L}{2} \|x_{n+1} - x_n\|^2 + \left(\frac{1}{2\gamma} - \frac{L}{2} \right) \|x_{n+1} - x_n\|^2 \\ &\geq f(x_{n+1}) + \left(\frac{1}{2\gamma} - \frac{L}{2} \right) \|x_{n+1} - x_n\|^2, \end{aligned} \quad (144)$$

leading to the sufficient decrease relation

$$\mathcal{J}(x_n) \geq \mathcal{J}(x_{n+1}) + \left(\frac{1}{2\gamma} - \frac{L}{2} \right) \|x_{n+1} - x_n\|^2. \quad (145)$$

- **H2: Relative error condition**

By optimality of the proximal operator of g , we have, for all $n \geq 0$

$$\frac{x_{n+1} - x_n}{\gamma} - \nabla f(x_{n+1}) \in \partial g(x_{n+1}), \quad (146)$$

and thus $\omega_n = \frac{x_{n+1} - x_n}{\gamma} \in \partial g(x_{n+1}) + \nabla f(x_{n+1}) = \partial \mathcal{J}(x_{n+1})$.

- **H3: Continuity condition**

The iterates (x_n) are assumed bounded. Thus, there exists a subsequence $(x_{n_i})_{i \in \mathbb{N}}$ converging towards $\hat{x} \in E$ as $i \rightarrow +\infty$. If we can show that for such a subsequence $\lim_{i \rightarrow \infty} g(x_{n_i}) = g(\hat{x})$, by continuity of f , we get $\lim_{i \rightarrow \infty} \mathcal{J}(x_{n_i}) = \mathcal{J}(\hat{x})$ and the continuity condition is verified. Using the fact that g is l.s.c we first have

$$\liminf_{i \rightarrow \infty} g(x_{n_i}) \geq g(\hat{x}). \quad (147)$$

On the other hand, by optimality of (142), we obtain

$$\begin{aligned} & g(x_{n_i+1}) + \langle x_{n_i+1} - x_{n_i}, \nabla f(x_{n_i}) \rangle + \frac{1}{2\gamma} \|x_{n_i+1} - x_{n_i}\|^2 \\ & \leq g(x) + \langle x - x_{n_i}, \nabla f(x_{n_i}) \rangle + \frac{1}{2\gamma} \|x - x_{n_i}\|^2. \end{aligned} \quad (148)$$

Using that $x_{n_i} \rightarrow x$ and $x_{n_i+1} - x_{n_i} \rightarrow 0$ when $i \rightarrow +\infty$, we get

$$\limsup_{i \rightarrow \infty} g(x_{n_i}) \leq g(x), \quad (149)$$

and

$$\lim_{i \rightarrow \infty} g(x_{n_i}) = g(x). \quad (150)$$

□

Remark 14. *The boundedness of the iterates is verified as soon as the objective \mathcal{J} is **coercive**. Indeed, it ensures that $\{\mathcal{J}(x) \leq \mathcal{J}(x_0)\}$ is bounded and, since $\mathcal{J}(x_k)$ is non-increasing, the iterates remain bounded.*

Remark 15. *When the function under the proximal operator (here g) is assumed α -weakly convex (i.e. $g(x) + \frac{\alpha}{2}\|x\|^2$ is convex), one can make the inequality (143) sharper and finally relax the condition on the stepsize in Theorem 15 from $\gamma < 1/L$ to $\gamma < 2/(L + \alpha)$. In particular, if g is convex, the condition on the stepsize becomes $\gamma < 2/L$.*

6.3 Other proximal splitting algorithms

We finally present some generalizations of Douglas-Rachford, ADMM and Primal-dual algorithms to the nonconvex setting.

6.3.1 Nonconvex Douglas-Rachford and ADMM

We remind the Douglas-Rachford (DR) algorithm introduced in Sections 4.3.4

$$\forall n \in \mathbb{N} \begin{cases} y_n & \in \text{prox}_{\gamma f}(x_n), \\ z_n & \in \text{prox}_{\gamma g}(2y_n - x_n), \\ x_{n+1} & = x_n + \mu(z_n - y_n). \end{cases} \quad (151)$$

As shown in Proposition 18, for $\mu = 1$, the DR algorithm is equivalent to the ADMM algorithm introduced in Section 4.3.6. In [29] and [47], convergence proofs of the DR algorithm are proposed for the minimization of the sum of two nonconvex functions $\mathcal{J} = f + g$, one of the two functions (here f) being L -smooth. The authors of [47] generalize the result from [29] with a less restrictive stepsize condition. Both consider as Lyapunov function the *Douglas-Rachford envelope* [47] (or *Douglas-Rachford merit function* [29])

$$\mathcal{J}_\gamma^{DR}(x, y, z) = f(y) + g(z) + \frac{1}{\gamma} \langle y - x, y - z \rangle + \frac{1}{2\gamma} \|y - z\|^2. \quad (152)$$

Similar to the previous result on Forward-Backward convergence, two convergence results can be derived in the nonconvex setting. First, without the KL hypothesis, one can show convergence of the Douglas-Rachford envelope $\mathcal{J}_\gamma^{DR}(x_n, y_n, z_n)$ along the iterates and convergence to 0 of $\|x_{n+1} - x_n\|$. Second, invoking the Kurdyka-Łojasiewicz (KL) property, we get convergence towards a critical point of the objective function. Both results are encompassed in the following theorem.

Theorem 16 ([29, 47]). *Assume that f and g are proper, lsc, lower-bounded and that f is L -smooth and α -weakly convex. Then, for a stepsize*

$$0 < \gamma < \min\left(\frac{2 - \mu}{2\alpha}, \frac{1}{L}\right), \quad (153)$$

the sequence (x_n, y_n, z_n) generated by the DR algorithm (151) verifies

- (i) $\mathcal{J}_\gamma^{DR}(x_{n-1}, y_n, z_n)$ is non-increasing and converges [47, Theorem 4.1].
- (ii) $x_n - x_{n-1} = \frac{\mu}{2}(y_n - z_n)$ tends to 0 with rate $\min_{n \leq N} \|y_n - z_n\| = \mathcal{O}(\frac{1}{\sqrt{N}})$ [47, Theorem 4.2].
- (iii) (y_n) and (z_n) have the same cluster points, which are critical points of \mathcal{J} [47, Theorem 4.2].
- (iv) If the sequence (x_n, y_n, z_n) is bounded, and if \mathcal{J}_γ^{DR} is KL, then the sequences (y_n) and (z_n) converge to the same critical point of \mathcal{J} [29, Theorem 2].

Remark 16. (i) We can assume f weakly convex without loss of generality. Indeed, as f is assumed L -smooth, f is at least L -weakly convex and we have necessarily $\alpha \leq L$.

- (ii) Contrary to what we had with Forward-Backward, for DR, the decreasing function is not \mathcal{J} itself but the Lyapunov function \mathcal{J}^{DR} . We can verify ([29, Theorem 4]) that if f or g are coercive, \mathcal{J}^{DR} is coercive and the iterates remain bounded.

We refer to [48] for more specific results on the convergence of the ADMM in the nonconvex setting.

6.3.2 Nonconvex primal version of the Chambolle-Pock algorithm

The Chambolle-Pock algorithm (99) can be written in a fully primal version using Moreau's identity (Corollary 3). We then get rid of the convex conjugate and the algorithm is well-defined for **nonconvex** f and g functions, and writes

$$\begin{cases} z_{n+1} & \in \text{prox}_{\frac{1}{\sigma}f}\left(\frac{1}{\sigma}y_n + K\bar{x}_n\right) \\ y_{n+1} & = y_n + \sigma(K\bar{x}_k - z_{n+1}) \\ x_{n+1} & \in \text{prox}_{\tau g}(x_n - \tau K^*y_{n+1}) \\ \bar{x}_{n+1} & = x_{n+1} + \beta(x_{n+1} - x_n). \end{cases} \quad (154)$$

The authors of [34] study the convergence of this algorithm for weakly convex f and strongly convex g , also assuming that the strong convexity of g dominates the weak convexity of f in order to ensure that the overall objective $F = f(K \cdot) + g$ is convex. In the fully nonconvex general case, the authors of [46] study the convergence of the algorithm (154) with $\beta = 0$, for nonconvex f and g functions. Similar to the DRS case in Theorem 16, they give a sufficient decrease condition on a particular Lyapunov function and prove convergence of the iterates only if these iterates are assumed to remain bounded. However, the coercivity of the functions f and g does not imply the coercivity of their proposed Lyapunov function and the boundedness of the iterates. The boundedness of the iterates has next been proven in [25, Theorem 12] provided f is differentiable with Lipschitz gradient.

6.3.3 Nonlinear Chambolle-Pock algorithm

We finally mention the problem

$$\min_x f(K(x)) + g(x),$$

for convex functions f and g associated to a nonlinear operator K . As shown in [14], it is possible to study the nonconvex/concave problem

$$\min_x \max_p \langle K(x), p \rangle - f^*(p) + g(x)$$

and show the local convergence of a linearized Chambolle-Pock algorithm inspired from (100):

$$\begin{cases} x_{n+1} &= \text{prox}_{\gamma g}(x_n - \gamma(\nabla_{x_n} K)^* y_{n+1}) \\ \bar{x}_{n+1} &= 2x_{n+1} - x_n \\ y_{n+1} &= \text{prox}_{\sigma f^*}(y_n + \sigma K \bar{x}_{n+1}). \end{cases} \quad (155)$$

7 Various imaging problems, various optimization algorithms

For any problem, several algorithms may be available. Choosing one algorithm among all possible existing ones may be a matter of personal taste and we do not provide any recommendation. In this section, we rather give practical imaging problems and detail which algorithms can be used to solve these problems. In particular cases, this will require a preliminary reformulation such as dualization or dimension extension.

In all the following examples, we want to minimize the sum of functions defined in \mathbb{R}^n and we assume that a minimizer always exists. We will in particular focus on inverse problems in which we have a noisy signal $y = Ax^* + \epsilon$ corresponding to the observation of a clean signal x^* measured through a linear operator A and perturbed by an additive Gaussian noise ϵ .

Such a formulation includes problems such as deblurring (with a smoothing operator A), super-resolution (with the combination of a smoothing operator with a subsampling one) or inpainting (with a masking operator A).

Lasso The LASSO in statistics is an estimator that is computed by solving a problem of the form:

$$\min_x \frac{1}{2} \|Ax - y\|_2^2 + \lambda \|x\|_1. \quad (156)$$

The L_1 -norm regularization term promotes sparse solutions. Such a class of problems can be efficiently treated with the Forward-Backward algorithm, with a proximal step on the L_1 -norm term. When the operator A is positive semi-definite, the problem is even $\|A\|$ -strongly convex and accelerated schemes such as FISTA provide optimal convergence rates.

Orthogonal wavelet regularization When dealing with images, it is possible to regularize the problem using an orthogonal wavelet decomposition $T^*T = TT^* = \text{Id}$:

$$\min_x \frac{1}{2} \|Ax - y\|_2^2 + \lambda \|Tx\|_1$$

Thanks to the orthogonality of T , a closed form expression of the proximal operator of $\|Tx\|_1$ is available. Hence, we can consider all the algorithms dedicated to non smooth optimization, using the proximity operator of $g(x) = \|Tx\|_1$ and an explicit or an implicit descent on $f(x) = \frac{1}{2} \|Ax - y\|_2^2$.

Total Variation Denoising A standard method to regularize ill-posed inverse problems is to promote piece-wise constant images through the minimization of the total variation (TV):

$$\min_x \frac{1}{2} \|x - y\|_2^2 + \lambda \|\nabla x\|_1, \quad (157)$$

where $\nabla = [\nabla_x, \nabla_y]$ is the $2D$ spatial gradient operator. The optimization of (157) requires the computation of the proximal operator of $g(x) = \|\nabla x\|_1$ for which no closed-form expression exists. This issue can be treated in different ways:

- Using Douglas-Rachford or PPXA algorithms (see sections 4.3.4 and 4.3.5) thanks to the dimension extension:

$$\min_{x,z} \underbrace{\frac{1}{2} \|x - y\|_2^2 + \lambda \|z\|_1}_{f(x,z)} + \underbrace{\iota_D(x, z)}_{g(x,z)} \quad \text{with } D = \{(x, z) \text{ s.t. } \nabla z = x\}.$$

The projection on D can be computed using conjugate gradient (with inner loops) or in closed-form if the gradient is periodic, as the proximal operator can be computed in an explicit way in the Fourier domain.

- Using Primal Dual algorithms (Chambolle-Pock and Condat) thanks to the saddle-point formulation:

$$\min_x \max_p \frac{1}{2} \|x - y\|_2^2 + \langle \nabla x, p \rangle - \iota_B(p), \quad (158)$$

where ι_B is the convex conjugate of $\|\cdot\|_1$, with

$$B = \{p \text{ such that } \|p\|_\infty \leq \lambda\}.$$

With respect to problem (87), this corresponds to $f(\cdot) = \|\cdot\|_1$, $K = \nabla$ and $g(x) = \frac{1}{2} \|x - y\|^2$.

- Using the Forward-Backward algorithm (49) on the dual problem associated to (157):

$$\min_p \underbrace{\frac{1}{2} \|y + \nabla^* p\|_2^2}_{f(p)} + \underbrace{\iota_{\mathcal{B}}(p)}_{g(p)} \quad (159)$$

where $\mathcal{B} = \{p \text{ such that } \|p\|_\infty \leq \lambda\}$. Indeed, as shown in [10], if p^* is a minimizer of (159) then $x^* = y + \nabla^* p^*$ is a solution of (157).

Total variation regularization We now consider that an operator is included in the data fidelity term in the inverse problem regularized by TV

$$\min_x \frac{1}{2} \|Ax - y\|_2^2 + \lambda \|\nabla x\|_1.$$

Without any assumptions on the operator A , the proximal operator of the data fidelity term $\frac{1}{2} \|Ax - y\|_2^2$ has no closed-form expression. To avoid approximate computation of this proximal operator with inner loops, one can consider primal-dual strategies:

- The Primal-dual algorithm (104) by Condat can be directly applied, with an explicit descent on f and a dualization of the TV term. Looking at the problem (103), it corresponds to $f(x) = \frac{1}{2} \|Ax - y\|_2^2$, $g = 0$, $h_1(\cdot) = \|\cdot\|_1$ and $L_1 = \nabla$.
- The Chambolle-Pock algorithm can also be considered with the following reformulation including two dualizations:

$$\min_x \max_{p,q} \langle p, Ax \rangle + \langle q, \nabla x \rangle - \frac{1}{2} \|y + p\|_2^2 - \iota_{\mathcal{B}}(q)$$

where $\mathcal{B} = \{q \text{ such that } \|q\|_\infty \leq \lambda\}$, $\iota_{\mathcal{B}}$ is the convex conjugate of $\|\cdot\|_1$ and $\frac{1}{2} \|\cdot + y\|_2^2$ is the convex conjugate of $\frac{1}{2} \|\cdot - y\|_2^2$. The saddle-point problem (91) is recovered with $y = [p, q]$, $f^*(y) = \frac{1}{2} \|y + p\|_2^2 + \iota_{\mathcal{B}}(q)$, $K = [A, \nabla]$ and $g = 0$.

TV Denoising with non smooth data fidelity Problems involving non Gaussian noise (such as salt and pepper, speckle or Laplacian) may require non smooth data attachment terms such as:

$$\min_x \frac{1}{2} \|x - y\|_1 + \lambda \|\nabla x\|_1$$

Primal-dual algorithms can directly be applied to this problem, whereas Douglas-Rachford and PPXA algorithms require a dimension extension:

$$\min_{x,z} \underbrace{\|x - y\|_1 + \lambda \|z\|_1}_f + \underbrace{\iota_D(x, z)}_g \text{ with } D = \{(x, z) \text{ such that } \nabla z = x\},$$

and inner loops to project onto the set D .

Image fusion We consider here the Poisson Image Editing problem proposed in [37] that consists in inserting in a smooth way a source image s into a target image y within an area delimited by a mask Ω . This problem can be solved by minimizing the following function:

$$\min_x \frac{1}{2} \|\nabla s - \nabla x\|_{\Omega}^2 + \iota_D(x) \quad \text{where } D = \{z \text{ such that } z|_{\bar{\Omega}} = y|_{\bar{\Omega}}\}. \quad (160)$$

This optimization problem may be solved using the Forward-Backward algorithm, which here corresponds to a project gradient algorithm: an explicit gradient descent on $\frac{1}{2} \|\nabla s - \nabla x\|_{\Omega}^2$ is followed by a projection on the set D .

Regularization with an image denoiser Given an image denoiser operator D_{σ} built to remove Gaussian noise with standard deviation σ from an image, one can consider regularization functions such as

$$g_{\sigma}(x) = \langle x, x - D_{\sigma}(x) \rangle \quad \text{or} \quad g_{\sigma}(x) = \frac{1}{2} \|x - D_{\sigma}(x)\|^2, \quad (161)$$

as respectively proposed in the Regularization by Denoising (RED) [43] or gradient step Plug-and-Play [27] frameworks. These regularization are then used to solve a variety of inverse problems by minimizing

$$\min_x f(x) + \lambda g_{\sigma}(x), \quad (162)$$

where f is a data-fidelity term.

The denoiser D_{σ} is typically a neural network which has been trained to denoise. With such denoiser, the RED regularization function g_{σ} is typically smooth (i.e. with Lipschitz gradient) but nonconvex. Depending on the regularity of the data-fidelity term, different algorithms can be used to solve this nonconvex objective. If f is smooth, one can use a standard (nonconvex) gradient descent; if f is non-smooth but has closed-form proximal operator, one can use the forward-backward algorithm with proximal step on f and gradient step on g_{σ} . Finally, it can happen that f is non-smooth and has no closed form proximity operator, for example when the observation noise has a Poisson distribution. In such situation, it can be useful to change the optimization algorithm to adapt to the particular geometry of f , using Bregman optimization algorithms [9, 26].

Acknowledgements

This work was funded in part by the Agence nationale de la recherche (ANR), Grant ANR-23- CE40-0017 and by the France 2030 program, with the reference ANR-23-PEIA-0004.

A Convergence of Chambolle-Pock algorithm

We here reproduce the convergence proof of the Chambolle-Pock algorithm introduced in [12]. We first recall that this algorithm targets a saddle-point of the following minmax problem:

$$\min_{x \in E} \max_{y \in F} \langle K^* y, x \rangle - f^*(y) + g(x), \quad (163)$$

and that we rely on the function:

$$h(x, y) = \langle Kx, y \rangle + g(x) - f^*(y). \quad (164)$$

related to the *partial primal-dual gap*:

$$\mathcal{G}_{B_1 \times B_2}(x, y) = \max_{y' \in B_2} \langle y', Kx \rangle - f^*(y') + g(x) - \min_{x' \in B_1} \langle y, Kx' \rangle + g(x') - f^*(y) \geq 0. \quad (165)$$

Theorem 17 (Chambolle-Pock Algorithm). *Let f and g be two convex, proper, lower semi-continuous functions, with f defined from F to $[-\infty, +\infty]$, g defined from E to $[-\infty, +\infty]$ and K a linear operator from F to E . We suppose that the problem (91) admits a solution (\hat{x}, \hat{y}) . Let us denote $L = \|K\| = \sqrt{\|K^*K\|}$, and take σ and τ such that $\tau\sigma L^2 < 1$. We choose $(x_0, y_0) \in E \times F$ and set $\bar{x}_0 = x_0$. We define the sequences $(x_n)_{n \in \mathbb{N}}$, $(y_n)_{n \in \mathbb{N}}$, and $(\bar{x}_n)_{n \in \mathbb{N}}$ by*

$$\begin{cases} y_{n+1} &= \text{prox}_{\sigma f^*}(y_n + \sigma K \bar{x}_n) \\ x_{n+1} &= \text{prox}_{\tau g}(x_n - \tau K^* y_{n+1}) \\ \bar{x}_{n+1} &= 2x_{n+1} - x_n \end{cases} \quad (166)$$

Then

1. For all $n \in \mathbb{N}$,

$$\frac{\|y_n - \hat{y}\|^2}{2\sigma} + \frac{\|x_n - \hat{x}\|^2}{2\tau} \leq (1 - \tau\sigma L^2)^{-1} \left(\frac{\|y_0 - \hat{y}\|^2}{2\sigma} + \frac{\|x_0 - \hat{x}\|^2}{2\tau} \right) \quad (167)$$

2. If we set $x^N = (\sum_{n=1}^N x_n)/N$ and $y^N = (\sum_{n=1}^N y_n)/N$, then for each bounded set $(B_1 \times B_2) \subset E \times F$, the partial primal-dual gap satisfies

$$\mathcal{G}_{B_1 \times B_2}(x^N, y^N) \leq \frac{D(B_1, B_2)}{N}$$

where

$$D(B_1, B_2) = \sup_{(x, y) \in B_1 \times B_2} \frac{\|x - x_0\|^2}{2\tau} + \frac{\|y - y_0\|^2}{2\sigma}.$$

3. The sequence $(x_n, y_n)_{n \in \mathbb{N}}$ converges to a solution (x^*, y^*) of (91).

Proof. In order to prove this theorem we will rewrite the iterations (166) on the following form:

$$\begin{cases} y_{n+1} = \text{prox}_{\sigma f^*}(y_n + \sigma K\bar{x}) \\ x_{n+1} = \text{prox}_{\tau g}(x_n - \tau K^*\bar{y}) \end{cases} \quad (168)$$

The fundamental properties of the proximity operators ensure that

$$\begin{aligned} K\bar{x} + \frac{y_n - y_{n+1}}{\sigma} &\in \partial f^*(y_{n+1}) \\ -K^*\bar{y} + \frac{x_n - x_{n+1}}{\tau} &\in \partial g(x_{n+1}) \end{aligned}$$

and thus, for all $(x, y) \in E \times F$,

$$\begin{aligned} f^*(y) &\geq f^*(y_{n+1}) + \left\langle \frac{y_n - y_{n+1}}{\sigma}, y - y_{n+1} \right\rangle + \langle K\bar{x}, y - y_{n+1} \rangle \\ g(x) &\geq g(x_{n+1}) + \left\langle \frac{x_n - x_{n+1}}{\tau}, x - x_{n+1} \right\rangle - \langle K(x - x_{n+1}), \bar{y} \rangle \end{aligned}$$

By adding these two inequalities we obtain:

$$\begin{aligned} &h(x_{n+1}, y) - h(x, y_{n+1}) \\ &+ \frac{\|y - y_{n+1}\|^2}{2\sigma} + \frac{\|x - x_{n+1}\|^2}{2\tau} + \frac{\|y_n - y_{n+1}\|^2}{2\sigma} + \frac{\|x_n - x_{n+1}\|^2}{2\tau} \\ &\quad + \langle K(x_{n+1} - \bar{x}), y_{n+1} - y \rangle - \langle K(x_{n+1} - x), y_{n+1} - \bar{y} \rangle \\ &\leq \frac{\|y - y_n\|^2}{2\sigma} + \frac{\|x - x_n\|^2}{2\tau} \end{aligned} \quad (169)$$

Chambolle and Pock propose to choose $\bar{y} = y_{n+1}$ and $\bar{x} = 2x_n - x_{n-1}$. Thus the second last line of the previous inequality can be written

$$\begin{aligned} &\langle K(x_{n+1} - \bar{x}), y_{n+1} - y \rangle - \langle K(x_{n+1} - x), y_{n+1} - \bar{y} \rangle \\ &= \langle K((x_{n+1} - x_n) - (x_n - x_{n-1})), y_{n+1} - y \rangle \\ &= \langle K(x_{n+1} - x_n), y_{n+1} - y \rangle - \langle K(x_n - x_{n-1}), y_n - y \rangle \\ &\quad - \langle K(x_n - x_{n-1}), y_{n+1} - y_n \rangle \\ &\geq \langle K(x_{n+1} - x_n), y_{n+1} - y \rangle - \langle K(x_n - x_{n-1}), y_n - y \rangle \\ &\quad - L\|x_n - x_{n-1}\|\|y_{n+1} - y_n\|. \end{aligned} \quad (170)$$

By using the fact that for all $\alpha > 0$ we have $2ab \leq \alpha a^2 + \frac{b^2}{\alpha}$, we obtain

$$L\|x_n - x_{n-1}\|\|y_{n+1} - y_n\| \leq \frac{L\alpha\tau}{2\tau}\|x_n - x_{n-1}\|^2 + \frac{L\sigma}{2\alpha\sigma}\|y_{n+1} - y_n\|^2 \quad (171)$$

By combining (169) with the previous inequality taken with $\alpha = \sqrt{\frac{\sigma}{\tau}}$, we deduce

that for all $(x, y) \in E \times F$,

$$\begin{aligned}
& h(x_{n+1}, y) - h(x, y_{n+1}) \\
& + \frac{\|y - y_{n+1}\|^2}{2\sigma} + \frac{\|x - x_{n+1}\|^2}{2\tau} + (1 - \sqrt{\sigma\tau}L) \frac{\|y_n - y_{n+1}\|^2}{2\sigma} \\
& + \frac{\|x_n - x_{n+1}\|^2}{2\tau} - \sqrt{\sigma\tau}L \frac{\|x_{n-1} - x_n\|^2}{2\tau} \\
& + \langle K(x_{n+1} - x_n), y_{n+1} - y \rangle - \langle K(x_n - x_{n-1}), y_n - y \rangle \\
& \leq \frac{\|y - y_n\|^2}{2\sigma} + \frac{\|x - x_n\|^2}{2\tau}.
\end{aligned} \tag{172}$$

Summing the previous inequalities from $n = 0$ to $N - 1$, and taking $x_{-1} = x_0$, we obtain:

$$\begin{aligned}
& \sum_{n=1}^N h(x_n, y) - h(x, y_n) \\
& + \frac{\|y - y_N\|^2}{2\sigma} + \frac{\|x - x_N\|^2}{2\tau} + (1 - \sqrt{\sigma\tau}L) \sum_{n=1}^N \frac{\|y_n - y_{n-1}\|^2}{2\sigma} \\
& + (1 - \sqrt{\sigma\tau}L) \sum_{n=1}^{N-1} \frac{\|x_n - x_{n-1}\|^2}{2\tau} + \frac{\|x_N - x_{N-1}\|^2}{2\tau} \\
& \leq \frac{\|y - y_0\|^2}{2\sigma} + \frac{\|x - x_0\|^2}{2\tau} + \langle K(x_N - x_{N-1}), y_N - y \rangle.
\end{aligned}$$

We use again the upper bound $2ab \leq \alpha a^2 + \frac{b^2}{\alpha}$, now with $\alpha = 1/(\sigma\tau)$:

$$\langle K(x_N - x_{N-1}), y_N - y \rangle \leq \|x_N - x_{N-1}\|^2/(2\tau) + (\sigma\tau L^2)\|y - y_N\|^2/(2\sigma)$$

and we obtain the following relation:

$$\begin{aligned}
& \sum_{n=1}^N h(x_n, y) - h(x, y_n) \\
& + (1 - \sigma\tau L^2) \frac{\|y - y_N\|^2}{2\sigma} + \frac{\|x - x_N\|^2}{2\tau} + (1 - \sqrt{\sigma\tau}L) \sum_{n=1}^N \frac{\|y_n - y_{n-1}\|^2}{2\sigma} \\
& + (1 - \sqrt{\sigma\tau}L) \sum_{n=1}^N \frac{\|x_n - x_{n-1}\|^2}{2\tau} \leq \frac{\|y - y_0\|^2}{2\sigma} + \frac{\|x - x_0\|^2}{2\tau}
\end{aligned} \tag{173}$$

We apply this inequality to a saddle-point $(x, y) = (\hat{x}, \hat{y})$ of (163). The first line of (173) is the sum of *partial primal-dual gaps* with (x, y) solutions to (163). As we have noted previously, this implies that all the terms of the sum from this first line are positive. We deduce the first result of the theorem from the fact that $\tau\sigma L^2 < 1$. We now consider an arbitrary pair $(x, y) \in B_1 \times B_2$. As

$$\sum_{n=1}^N h(x_n, y) - h(x, y_n) \leq \frac{\|y - y_0\|^2}{2\sigma} + \frac{\|x - x_0\|^2}{2\tau}. \tag{174}$$

As f^* and g are convex, we deduce that

$$h(x^N, y) - h(x, y^N) \leq \frac{1}{N} \left(\frac{\|y - y_0\|^2}{2\sigma} + \frac{\|x - x_0\|^2}{2\tau} \right)$$

By taking the supremum on $(x, y) \in B_1 \times B_2$ of the two sides of the inequality, we obtain the second point of the Lemma.

Using the first point of the theorem we know that the sequences $(x_n)_{n \in \mathbb{N}}$ and $(y_n)_{n \in \mathbb{N}}$ are bounded. As the dimension of E is finite, from each of these sequences we can extract subsequences that converge to some limits x^* et y^* . By taking the limit when N goes to $+\infty$ in the previous inequality we obtain

$$\langle Kx^*, y \rangle - f^*(y) + g(x^*) - (\langle Kx, y^* \rangle - f^*(y^*) + g(x)) \leq 0$$

for all $(x, y) \in B_1 \times B_2$. By taking the supremum over the pairs $(x, y) \in B_1 \times B_2$, we deduce that the *partial primal-dual gap* in the point (x^*, y^*) is negative and thus zero, and thus (x^*, y^*) is a solution to (163).

As the sequences $(x_n)_{n \in \mathbb{N}}$ and $(y_n)_{n \in \mathbb{N}}$ are bounded we can extract subsequences $(x_{n_k})_{k \in \mathbb{N}}$ and $(y_{n_k})_{k \in \mathbb{N}}$ that converge to some points x^∞ et y^∞ . As the series having the general term $\|x_n - x_{n-1}\|^2$ and $\|y_n - y_{n-1}\|^2$ are convergent we deduce that the sequences $(\|x_n - x_{n-1}\|^2)_{n \in \mathbb{N}}$ and $(\|y_n - y_{n-1}\|^2)_{n \in \mathbb{N}}$ go to zero when n goes to $+\infty$. We deduce that the sequences $(x_{n_k+1})_{k \in \mathbb{N}}$ et $(y_{n_k+1})_{k \in \mathbb{N}}$ also converge to x^∞ et y^∞ and thus x^∞ et y^∞ are fixed points of (166). Fixed points of (166) satisfy relations (92). Thus they are saddle-points, that is to say solutions of (163).

By adding the inequalities (172) for $n = n_k$ to N we obtain

$$\begin{aligned} & \frac{\|y^* - y_N\|^2}{2\sigma} + \frac{\|x^* - x_N\|^2}{2\tau} + (1 - \sqrt{\sigma\tau}L) \sum_{n=n_k+1}^N \frac{\|y_n - y_{n-1}\|^2}{2\sigma} \\ & - \frac{\|x_{n_k} - x_{n_k-1}\|^2}{2\tau} + (1 - \sqrt{\sigma\tau}L) \sum_{n=n_k}^{N-1} \frac{\|x_n - x_{n-1}\|^2}{2\tau} + \frac{\|x_N - x_{N-1}\|^2}{2\tau} \\ & + \langle K(x_N - x_{N-1}), y_N - y^* \rangle - \langle K(x_{n_k} - x_{n_k-1}), y_{n_k} - y^* \rangle \\ & \leq \frac{\|y^* - y_{n_k}\|^2}{2\sigma} + \frac{\|x^* - x_{n_k}\|^2}{2\tau}. \end{aligned} \quad (175)$$

By using the fact that $\lim_{n \rightarrow +\infty} \|x_n - x_{n-1}\| = \lim_{n \rightarrow +\infty} \|y_n - y_{n-1}\| = 0$, we deduce that the sequence $(x_N, y_N)_{N \in \mathbb{N}}$ goes to (x^*, y^*) , which gives the third point of the theorem. \square

References

- [1] H. Attouch, J. Bolte, P. Redont, and A. Soubeyran. Proximal alternating minimization and projection methods for nonconvex problems: An approach based on the kurdyka-łojasiewicz inequality. *Mathematics of operations research*, 35(2):438–457, 2010.
- [2] H. Attouch, J. Bolte, and B. Svaiter. Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward-backward splitting, and regularized Gauss-Seidel methods. *Mathematical Programming*, 137(1-2, Ser. A):91–129, 2013.
- [3] J.-F. Aujol, C. Dossal, and A. Rondepierre. Fista is an automatic geometrically optimized algorithm for strongly convex functions. *Mathematical Programming*, pages 1–43, 2023.
- [4] H. H. Bauschke and P. Combettes. *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer, 2011.
- [5] H. H. Bauschke, S. M. Moffat, and X. Wang. Firmly nonexpansive mappings and maximally monotone operators: correspondence and duality. *Set-Valued and Variational Analysis*, 20(1):131–153, 2012.
- [6] A. Beck. *First-order methods in optimization*. SIAM, 2017.
- [7] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.
- [8] J. Bolte, A. Daniilidis, and A. Lewis. The łojasiewicz inequality for nonsmooth subanalytic functions with applications to subgradient dynamical systems. *SIAM Journal on Optimization*, 17(4):1205–1223, 2007.
- [9] J. Bolte, S. Sabach, M. Teboulle, and Y. Vaisbourd. First order methods beyond convexity and lipschitz gradient continuity with applications to quadratic inverse problems. *SIAM Journal on Optimization*, 28(3):2131–2151, 2018.
- [10] A. Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging and Vision*, 20(1–2):89–97, 2004.
- [11] A. Chambolle and C. Dossal. On the convergence of the iterates of the “fast iterative shrinkage/thresholding algorithm”. *Journal of Optimization theory and Applications*, 166:968–982, 2015.
- [12] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision* 40, 2011.
- [13] A. Chambolle and T. Pock. On the ergodic convergence rates of a first-order primal–dual algorithm. *Mathematical Programming*, 159(1-2):253–287, 2016.

- [14] C. Clason and T. Valkonen. Primal-dual extragradient methods for nonlinear nonsmooth pde-constrained optimization. *SIAM Journal on Optimization*, 27(3):1314–1339, 2017.
- [15] P. Combettes and J.-C. Pesquet. Proximal splitting methods in signal processing. *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, 2011.
- [16] L. Condat. A primal-dual splitting method for convex optimization involving lipschitzian, proximable and linear composite terms. *Journal of Optimization Theory and Applications*, 158(2):460–479, 2013.
- [17] L. Condat, D. Kitahara, A. Contreras, and A. Hirabayashi. Proximal splitting algorithms for convex optimization: A tour of recent advances, with new twists. *SIAM Review*, 65(2):375–435, 2023.
- [18] M. Coste. An introduction to semialgebraic geometry, 2000.
- [19] J. Eckstein and D. P. Bertsekas. On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators. *Mathematical Programming*, 55:293–318, 1992.
- [20] D. Gabay. Chapter ix applications of the method of multipliers to variational inequalities. In *Studies in mathematics and its applications*, volume 15, pages 299–331. Elsevier, 1983.
- [21] D. Gabay and B. Mercier. A dual algorithm for the solution of nonlinear variational problems via finite element approximation. *Computers & Mathematics with Applications*, 2(1):17 – 40, 1976.
- [22] G. Garrigos and R. M. Gower. Handbook of convergence theorems for (stochastic) gradient methods. *arXiv preprint arXiv:2301.11235*, 2023.
- [23] R. Glowinski and A. Marroco. Sur l’approximation, par éléments finis d’ordre un, et la résolution, par pénalisation-dualité d’une classe de problèmes de dirichlet non linéaires. *ESAIM: Mathematical Modelling and Numerical Analysis*, 9(R2):41–76, 1975.
- [24] B. He and X. Yuan. Convergence analysis of primal-dual algorithms for a saddle-point problem: from contraction perspective. *SIAM Journal on Imaging Sciences*, 5(1):119–149, 2012.
- [25] S. Hurault. *Convergent plug-and-play methods for image inverse problems with explicit and nonconvex deep regularization*. PhD thesis, Université de Bordeaux, 2023.
- [26] S. Hurault, U. Kamilov, A. Leclaire, and N. Papadakis. Convergent bregman plug-and-play image restoration for poisson inverse problems. *Advances in Neural Information Processing Systems*, 36, 2024.

- [27] S. Hurault, A. Leclaire, and N. Papadakis. Gradient step denoiser for convergent plug-and-play. In *International Conference on Learning Representations*, 2022.
- [28] K. Kurdyka. On gradients of functions definable in o-minimal structures. *Annales de l'institut Fourier*, 48(3):769–783, 1998.
- [29] G. Li and T. K. Pong. Douglas–rachford splitting for nonconvex optimization with application to nonconvex feasibility problems. *Math. Progr.*, 159:371–401, 2016.
- [30] S. Lojasiewicz. Une propriété topologique des sous-ensembles analytiques réels. *Les équations aux dérivées partielles*, 117:87–89, 1963.
- [31] S. Lojasiewicz. Sur les trajectoires du gradient d’une fonction analytique. *Seminari di geometria*, 1983:115–117, 1982.
- [32] H. Lu and J. Yang. On a unified and simplified proof for the ergodic convergence rates of ppm, pdhg and admm. *arXiv preprint arXiv:2305.02165*, 2023.
- [33] J.-J. Moreau. Proximité et dualité dans un espace hilbertien. *Bulletin de la Société Mathématique de France*, 93:273–299, 1965.
- [34] T. Möllenhoff, E. Strelakovski, M. Moeller, and D. Cremers. The primal-dual hybrid gradient method for semiconvex splittings. *SIAM Journal on Imaging Sciences*, 8(2):827–857, 2015.
- [35] Y. Nesterov. A method for solving the convex programming problem with convergence rate $O(1/k^2)$. *Doklady Akademii Nauk SSSR*, 269(3):543–547, 1983.
- [36] N. Papadakis, G. Peyré, and E. Oudet. Optimal transport with proximal splitting. *SIAM Journal on Imaging Sciences*, 7(1):212–238, 2014.
- [37] P. Pérez, M. Gangnet, and A. Blake. Poisson image editing. In *ACM SIG-GRAPH 2003 Papers*, pages 313–318, 2003.
- [38] B. T. Polyak. Some methods of speeding up the convergence of iteration methods. *Ussr computational mathematics and mathematical physics*, 4(5):1–17, 1964.
- [39] R. Rockafellar. On the maximality of sums of nonlinear monotone operators. *Transactions of the American Mathematical Society.*, 149:75–88, 1970.
- [40] R. Rockafellar. *Convex Analysis*. Convex Analysis. Princeton University Press, 1997.
- [41] R. T. Rockafellar. Monotone operators and the proximal point algorithm. *SIAM journal on control and optimization*, 14(5):877–898, 1976.
- [42] R. T. Rockafellar and R. J.-B. Wets. *Variational analysis*, volume 317. Springer Science & Business Media, 2009.

- [43] Y. Romano, M. Elad, and P. Milanfar. The little engine that could: Regularization by denoising (red). *SIAM Journal on Imaging Sciences*, 10(4):1804–1844, 2017.
- [44] A. Rondepierre. Méthodes numériques pour l’optimisation non linéaires déterministe 4eme année. *INSA Toulouse*, 2017.
- [45] W. Su, S. Boyd, and E. J. Candes. A differential equation for modeling nesterov’s accelerated gradient method: Theory and insights. *Journal of Machine Learning Research*, 17(153):1–43, 2016.
- [46] T. Sun, R. Barrio, L. Cheng, and H. Jiang. Precompact convergence of the nonconvex primal–dual hybrid gradient algorithm. *Journal of Comp. Appl. Math.*, 330:15–27, 2018.
- [47] A. Themelis and P. Patrinos. Douglas–rachford splitting and admm for nonconvex optimization: Tight convergence results. *SIAM Journal on Optimization*, 30(1):149–181, 2020.
- [48] Y. Wang, W. Yin, and J. Zeng. Global convergence of admm in nonconvex nonsmooth optimization. *Journal of Scientific Computing*, 78:29–63, 2019.
- [49] P. Weiss. éléments d’analyse et d’optimisation convexe. *INSA Toulouse*, 2015.