Convergence of a Piggyback-style method for the differentiation of solutions of standard saddle-point problems

Lea Bogensperger, Antonin Chambolle, Thomas Pock

Convergence of a Piggyback-style method for the differentiation of solutions of standard saddle-point problems

Lea Bogensperger*, Antonin Chambolle†, and Thomas Pock*

Abstract. We analyse a "piggyback"-style method for computing the derivative of a loss which depends on the solution of a convex-concave saddle point problems, with respect to the bilinear term. We attempt to derive guarantees for the algorithm under minimal regularity assumption on the functions. Our final convergence results include possibly nonsmooth objectives. We illustrate the versatility of the proposed piggyback algorithm by learning optimized shearlet transforms, which are a class of popular sparsifying transforms in the field of imaging.

Key words. First-order methods, saddle-point problems, differentiation, adjoint methods, piggyback algorithm, learning

AMS subject classifications. 90C31, 49M29, 90C06, 90C47, 90C25

1. Introduction.

In [11], the authors considered the problem of "learning" consistent discretizations of the total variation (TV), for improving the solution of image recovery tasks such as denoising or inpainting. In order to achieve this goal, it was needed to compute the gradient of an error (loss) function depending on the minimizer of a convex, and non-smooth, optimization problem. This problem is of course just a particular instance of a more general class of optimization problems frequently arising in machine learning, optimal control or bilevel optimization [18].

In [11], the gradient was computed by means of an iterative algorithm which is known in the automatic differentiation literature [26] as "piggyback" [25]. This method, which is both memory and computationally efficient, aims at finding simultaneously the solution of a linear or non-linear equation (corresponding either to a differential equation such as the optimality condition of an optimization problem) and the (solution dependent) adjoint state which allows to compute the gradient of a loss depending on the solution, by running in parallel a fixed-point iteration and an appropriate combination of its derivative with the derivative of the loss term. Historically, the piggyback algorithm is inspired by the study of error propagation in the reverse mode of automatic differentiation, for example in [39, Eq. (9)] in the context of multidisciplinary design procedures and in [13] (see pp. 6-7). See also [47] for a comparative study.

The issue of differentiating with respect to parameters solutions of optimization algorithms or fixed-point iterations is essential in imaging and machine learning and has been studied by many authors in this context, in the recent years. For imaging applications, the starting point seems to be the work by Samuel and Tappen [45] in the context of learning the pa-

†CEREMADE, CNRS & Université Paris-Dauphine PSL, 75016 Paris, France (chambolle@ceremade.dauphine.fr).

*Institute of Computer Graphics and Vision, Graz University of Technology, 8010 Graz, Austria (lea.bogensperger@icg.tugraz.at, pock@icg.tugraz.at).

rameters for (maximum a-posterior) MAP estimation in continuous valued Markov random field (MRF) models for image restoration. They adopted implicit differentiation to derive a formula for computing the gradient of the loss function with respect to the parameters of a (local) minimizer of the non-convex MRF model. Note that their formula [45, Eq. (6)] corresponds to eq. (2.2) in the present work. In the subsequent work [42], Peyré and Fadili derived formula (2.2) (see [42, Eq. (5)]) in a particular setting, the context of analysis based dictionary learning. As in our work, one interesting feature is that these authors are able to address problems with relatively low smoothness and growth conditions.

In a similar flavor, the paper [31], focused on image reconstruction methods, derives an adjoint problem from the discrete equation, which is then solved by semi-smooth Newton algorithms. For such applications, a quite different school chose to work rather in the continuous setting, starting from [16], where a bi-level parameter identification scheme was proposed for a total-variation reconstruction problem with varying weight [20]. These authors proposed, as in many subsequent works (see the review [7] and the references therein), a detailed study of the (smoothed) total variation minimization and its sensitivity analysis in the continuous setting, with a derivation of an adjoint problem and its discretization. A similar point of view is adopted in [28, 29], see also the review [27].

In the context of more general nonsmooth optimization, such as for the Least absolute shrinkage and thresholding operator (LASSO) problem, and applications to machine learning, the literature is of course also very important. (See for instance [17] where the authors adapt their approach to a whole family of standard non-smooth iterative solvers.)

In practice, one usually distinguishes *Forward* and *Reverse* approaches. The latter, which come from classical automatic differentiation (see for instance [26]), consist in "unrolling" the iterations of an optimization algorithm and differentiate a function of the output by classical backpropagation. In imaging, early paper based on this approach are [48, 24, 19], or the more recent works [46, 12, 21]. In order to deal with non-smooth models, the work [40] unrolls and differentiates non-linear proximal algorithms that ensure differentiability of the iterates. In very recent work [36] the convergence rate of gradients is investigated by unrolling and differentiating accelerated proximal algorithms.

On the other hand, a forward approach will compute, in parallel to the optimization algorithm, the Jacobian of the solution with respect to some parameters. One issue with this latter approach is that such a variable is often huge, and one needs techniques to reduce the size of the computation. For instance, the recent contribution [6], for a LASSO problem, leverages the sparsity of the solutions in order to reduce the size of the Jacobians. Before this, many authors have been addressing general techniques for forward differentiation of fixed-point iterations. In the context of TV reconstruction, one notable reference is [17] where the regularity of the iterative scheme is discussed and the technique is developed for most common non-smooth optimization algorithms, while in the former reference [49], a projection technique to reduce the size of the Jacobian is proposed. A detailed comparison of Forward and Reverse derivation in learning is found in [23]. In these approaches, it is not easy to rely on adjoint states to reduce the size of the problems.

An adjoint discrete approach is found, for instance, in [41, 35], which is based on the computation of an "inverse-Hessian-vector" product (quite similar to the seminal approach in [45], and corresponding to [11, eqs. (A3)-(A4)] in our analysis, that is, to the explicit

evaluation of an adjoint state). In particular, the approach in [41], based on inexact non-smooth algorithms, shares some similarities with the nonsmooth piggyback method. (See also [43] for a variant.)

In this paper, we analyse further the piggyback method used in [11] and where, as proposed in [25], the variables and the adjoint variables are computed iteratively in a parallel forward way. One advantage, clearly, is that the size of the adjoint variable is the same as the main variable. Our analysis shows that while we still need the algorithm to be contractive like in most of the literature, the method can work with little smoothness. Our typical problem is a min-max convex-concave saddle-point of the form

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

for convex functions f, g with convex conjugates f^*, g^*, which we tackle by first order primal-dual algorithms [8, 9, 10], and we restrict our attention to derivating with respect to the operator K (the same method will work for derivating with respect to parameters in g or f^*, but then we believe that more regularity is needed). Assuming only that f^* and g are strongly convex (that is, f and g^* are $C^{1,1}$), we prove that the method makes sense: a fixed point of the algorithm allows to define a derivative. We also show convergence of the piggyback under slightly stronger assumptions (for f, g^* $C^{2,\alpha}$). In particular, this allows to consider problems of the primal form $\min_x f(Kx) + g(x)$ with, possibly, g non-smooth, or f with linear growth, slightly extending the framework of [42].

The paper is organized as follows: in the next section we introduce the problem and the algorithm used in [11], and state the main results (of consistency, Thm 2.1, and convergence, Thms 2.2-2.3). The section which follows is a remark of the consequences of the analysis for the primal form of the problem. Section 4 gathers various preliminary results on convex functions and their regularization, and the main proofs are in Section 6. We illustrate this approach in Section 7 where it is applied to design optimal shearlets for denoising (see also [11]).

2. Derivatives of saddle-points.

We consider a standard problem in optimization

$$(\mathcal{P}) \qquad \min_{x\in\mathcal{X}} f(Kx) + g(x)$$

where $\mathcal{X} = \mathbb{R}^n$, $K \in \mathbb{R}^{m\times n}$, $m, n \geq 0$, and here f and g are proper, convex lower semicontinuous functions, in addition we assume that f, g are such that a solution of $(\mathcal{P})$ exists, as well as of the *dual problem* defined as

$$(\mathcal{D}) \qquad \sup_{y\in\mathcal{Y}} -f^*(y) - g^*(-K^*y)$$

with $\mathcal{Y} = \mathbb{R}^m$, and where K^* is the adjoint operator of K (hence defined by the transpose $n \times m$ matrix K^T). In that case, a solution x_K to $(\mathcal{P})$ and a solution y_K to $(\mathcal{D})$ are also such that (x_K, y_K) is a saddle point of the min-max optimization problem:

$$(\mathcal{S}) \qquad \min_{x\in\mathcal{X}} \max_{y\in\mathcal{Y}} \langle Kx, y\rangle + g(x) - f^*(y)$$

which can be solved by a standard modified Arrow-Hurwicz type iteration as analysed in [8, 10]. It is a solution of the system:

$$\begin{cases} Kx_K - \partial f^*(y_K) \ni 0 \\ K^* y_K + \partial g(x_K) \ni 0 \end{cases} \tag{2.1}$$

In this paper, we will additionally assume that the solution (x_K, y_K), given an operator K, is unique. This is ensured if g and f^* are strictly convex, and we make this stronger assumption. As a result, g^* and f are $C^{1,1}$ functions, respectively defined on the full domains $\mathcal{X}$ and $\mathcal{Y}$. This uniqueness is crucial, as we want to define a "loss function" depending on K through the solution (x_K, y_K). (An interesting extension would be to analyse a loss depending only on x_K, and drop the uniqueness assumption on y, but this remains difficult.)

Given a smooth (generally convex, for instance quadratic) function $\ell : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}_+$, we define for any $K \in \mathbb{R}^{m\times n}$ the "loss" $\mathcal{L}(K) := \ell(x_K, y_K)$. Our goal is too understand how to estimate the gradient of $\mathcal{L}$ with respect to K.

In [11, Appendix] was proposed, in case g, f^* are smooth, $C^{2,\alpha}$ and strongly convex, an iterative algorithm to compute $\nabla\mathcal{L}(K)$. It is shown that

$$\nabla\mathcal{L}(K) = Y_K \otimes x_K + y_K \otimes X_K. \tag{2.2}$$

where the adjoint variables $(X_K, Y_K) \in \mathcal{X} \times \mathcal{Y}$ solve the the saddle-point problem:

$$\min_{X\in\mathcal{X}} \sup_{Y\in\mathcal{Y}} \langle KX, Y\rangle + \frac{1}{2}\langle D^2 g(x_K)X, X\rangle - \frac{1}{2}\langle D^2 f^*(y_K)Y, Y\rangle + \left\langle \nabla\ell(x_K, y_K), \begin{pmatrix} X \\ Y \end{pmatrix} \right\rangle. \tag{2.3}$$

The adjoint states were found by the following piggyback algorithm [25].

Piggyback algorithm:. Choose $(x^0, y^0), (X^0, Y^0) \in \mathcal{X} \times \mathcal{Y}$. For each $k \geq 0$,

1. compute $\tilde{x} = x^k - \tau K^* y^k$ and $\tilde{X} = X^k - \tau(K^* Y^k + \nabla_x \ell(x^k, y^k))$;
2. compute using automatic differentiation $x^{k+1} = \operatorname{prox}_{\tau g}(\tilde{x})$ and $X^{k+1} = \nabla \operatorname{prox}_{\tau g}(\tilde{x}) \cdot \tilde{X}$;
3. compute $\bar{x}^{k+1} := x^{k+1} + \theta(x^{k+1} - x^k)$, $\bar{X}^{k+1} := X^{k+1} + \theta(X^{k+1} - X^k)$, and $\tilde{y} = y^k + \sigma K\bar{x}^{k+1}$, $\tilde{Y} = Y^k + \sigma(K\bar{X}^{k+1} + \nabla_y \ell(x^k, y^k))$;
4. compute using a.d. again $y^{k+1} = \operatorname{prox}_{\sigma f^*}(\tilde{y})$, $Y^{k+1} = \nabla \operatorname{prox}_{\sigma f^*}(\tilde{y}) \cdot \tilde{Y}$;
5. return to 1.

It was shown in [11], with additional regularity assumptions on f^* and g, that the iterates converge linearly to (x_K, y_K, X_K, Y_K) as $k \to \infty$ for appropriate choices of σ, τ, and $\theta \in]0, 1]$.

With this method, the limiting (X_K, Y_K) are a fixed point of:

$$\begin{cases} X = \nabla \operatorname{prox}_{\tau g}(x_K - \tau K^* y_K) \cdot \big[X - \tau(K^* Y + \nabla_x \ell(x_K, y_K))\big] \\ Y = \nabla \operatorname{prox}_{\sigma f^*}(y_K + \sigma K x_K) \cdot \big[Y + \sigma(K\, X + \nabla_y \ell(x_K, y_K))\big] \end{cases} \tag{2.4}$$

In practice, an advantage of this approach is that the algorithm can be run easily on far less smooth problems, since for any convex function φ, $\operatorname{prox}_\varphi$ is a one-Lipschitz (more precisely, firmly non-expansive) operator and in particular its classical gradient exists almost

everywhere. (Of course, it is far from clear that the solutions $x_K - \tau K^* y_K$ or the iterates $\tilde{x}$ in point (1) of the Algorithm avoid the exceptional set where $\nabla \operatorname{prox}_{\tau g}$ fails to exists, for a.e. K, and this issue is extremely hard to address.)

Under the assumption (weaker than in [11]) that f^* and g are strongly convex, we study here whether, assuming $\operatorname{prox}_{\tau g}$ and $\operatorname{prox}_{\sigma f^*}$ are differentiable at, respectively, $x_K - \tau K^* y_K$ and $y_K + \sigma K x_K$, a fixed point (X_K, Y_K) of (2.4) allows to compute the gradient of $\mathcal{L}$ through (2.2). Then, assuming slightly more regularity on f and g^*, we show the convergence of the piggyback method in this context. We stress that f^*, g are not assumed to be regular, more than what is implied by the regularity of f and g^*, in particular they could be non-differentiable. We now state our main three results. The proof of the first is given in Section 5, while the two other are proved in Section 6.

Theorem 2.1. *Assume that g and f^* are strongly convex (with respective moduli $\gamma, \delta > 0$). Assume that (X_K, Y_K) is a fixed point of (2.4) and in particular that $\operatorname{prox}_{\tau g}$ and $\operatorname{prox}_{\sigma f^*}$ are differentiable at, respectively, $x_K - \tau K^* y_K$ and $y_K + \sigma K x_K$, and symmetric[1]. Then, $\mathcal{L}$ is differentiable at K with derivative given by (2.2).*

Remark 1. *The statement is local, and should remain valid, at least for τ, σ small enough, when f^*, g are strongly convex in a neighborhood of y, x, respectively, or equivalently if f, g^* are $C^{1,1}$ in a neighborhood of the respective closed sets $\{z : \nabla f(z) = \nabla f(Kx)\}$ and $\{z : \nabla g^*(z) = \nabla g^*(-K^* y)\}$.*

Theorem 2.2. *Assume that g and f^* are strongly convex (with respective moduli $\gamma, \delta > 0$), and that in addition, g^* and f are locally $C^{2,\alpha}$. Then for τ, σ chosen as in [8, Alg. 3] and θ chosen slightly larger, the iterates (x^k, y^k), (X^k, Y^k) of the piggyback algorithm converge linearly to (x_K, y_K) and (X_K, Y_K) which satisfy (2.4), and $\nabla \mathcal{L}(K)$ is given by (2.2).*

The choice of the parameters in [8] is as follows: one picks $\mu \leq 2\sqrt{\gamma\delta}/\|K\|$, $\tau = \mu/(2\gamma)$, $\sigma = \mu/(2\delta)$ and $1/(1+\mu) < \theta \leq 1$. In the proof, we will show the result for $\theta = 2/(2+\mu)$, yet it could be adapted, following the steps in [8, pp. 130-131], to the range $1/(1+\mu) < \theta \leq 1$. The limiting case $\theta = 1/(1+\mu)$ does not allow to absorb, in the iterations for X^k, the errors due to the inexactness of the value of (x, y). The proof of Theorem 2.2 is given in Section 6.2.

Remark 2. *The proof of Theorem 2.2 shows that a similar result still holds if (x^k, y^k) is replaced, in the algorithm, with any sequence linearly converging to the saddle-point (x_K, y_K).*

Alternatively to the piggyback algorithm described above, one can also estimate $\nabla \mathcal{L}(K)$ as follows: one first runs an algorithm (such as [8, Alg. 3], see also [10, Alg. 5]) which provides an approximation (x, y) of (x_K, y_K). Then, one performs, for (X^k, Y^k) the same iterations as in the piggyback above, replacing however (x^k, y^k) with the fixed values (x, y). The analysis in Section 6.1 shows the following result.

Theorem 2.3. *Assume that g and f^* are strongly convex (with respective moduli $\gamma, \delta > 0$), and that in addition, g^* and f are locally $C^{2,\alpha}$. Then for τ, σ, θ chosen as in [8, Alg. 3], the iterates (X^k, Y^k) obtained with this variant of the algorithm satisfy*

$$\|y \otimes X^k + Y^k \otimes x - \nabla \mathcal{L}(K)\| \leq C((\|x - x_K\| + \|y - y_K\|)^\alpha + \omega^{k/2}).$$

[1] We will see later on that these Lipschitz operators have symmetric derivative almost everywhere.

for some constant C (depending on the all the data) and some rate $\omega \leq \theta$.

Precisely, the rate which can be achieved here is $\omega = (1+\theta)/(2+\mu) < 1$ for the choices already mentioned above: $\mu \leq 2\sqrt{\gamma\delta}/\|K\|$, $\tau = \mu/(2\gamma)$, $\sigma = \mu/(2\delta)$ and $\theta \in [1/(1+\mu), 1]$, see [8].

Remark 3 (Parameter dependent functions f or g). *An easy formal analysis shows that for instance, in case $g(x) = g(x, r)$ depends on a parameter $r \in \mathbb{R}^{n_r}$, then a loss defined through $\mathcal{L}(r) := \ell(x, y)$ should satisfy $\nabla\mathcal{L}(r) = \sum_i (X_K)_i \nabla_r \partial_i g(x_K, r)$, for the same adjoint states (X_K, Y_K). However, it seems that the analysis requires much stronger regularity of the function g (at least $C^{1,1}$ and strongly convex, locally uniformly with respect to the parameter r).*

3. (Accelerated) forward-backward splitting. In the setting which we consider in Theorems 2.1–2.3, since by assumption $f(K\cdot)$ has Lipschitz gradient (with constant at most $\|K\|^2/\delta$), it is also possible to find $(x_K, y_K) = (x_K, \nabla f(Kx_K))$ by an accelerated forward-backward method (such as FISTA [5], or in the strongly convex case rather [38, (2.2.19)], see also [9], Theorem B.1 and Remark B.2): given x^0, $x^{-1} = x^0$, one lets for each $k \geq 1$:

$$\begin{cases} \bar{x}^k = x^k + \beta_k(x^k - x^{k-1}), \\ x^{k+1} = \operatorname{prox}_{\tau g}(\bar{x}^k - \tau K^* \nabla f(K\bar{x}^k)). \end{cases} \tag{3.1}$$

Then, for an appropriate choice of β^k, and $\tau = \delta/\|K\|^2$, letting also $y_k = \nabla f(Kx^k)$, one has linear convergence of (x^k, y^k) to (x_K, y_K), so that in particular this scheme can be used to define the sequence in Theorem 2.2. The choice $\beta_k = (1-\sqrt{q})/(1+\sqrt{q})$, for $q = \tau\gamma/(1+\tau\gamma)$, yields a linear rate with almost optimal contraction factor $(1-\sqrt{q})$, *cf* [38, 9] (a varying step is also possible and slightly improves the convergence).

A natural question is whether it is also possible, at this point, to replace the piggyback primal-dual algorithm by a descent algorithm. Observe that formally taking the sup in (2.3) yields the minimization problem:

$$\begin{aligned} \min_{X \in \mathcal{X}} \frac{1}{2} \left\langle D^2 f(Kx_K)(KX + \nabla_y \ell(x_K, y_K)), KX + \nabla_y \ell(x_K, y_K)) \right\rangle \\ + \frac{1}{2} \left\langle D^2 g(x_K) X, X \right\rangle + \left\langle \nabla_x \ell(x_K, y_K), X \right\rangle, \end{aligned} \tag{3.2}$$

so that one should expect to be able to find X_K (and $Y_K = D^2 f(Kx_K)(KX + \nabla_y \ell(x_K, y_K))$) by iterating:

$$\begin{cases} \bar{X}^k = X^k + \beta_k(X^k - X^{k-1}), \\ X^{k+1} = \nabla \operatorname{prox}_{\tau g}(\bar{x}^k - \tau K^* \nabla f(K\bar{x}^k)) \cdot \big[\bar{X}^k \\ \qquad\qquad -\tau K^* D^2 f(Kx^k)(K\bar{X}^k + \nabla_y \ell(x_k, y_k)) - \tau \nabla_x \ell(x_k, y_k)\big]. \end{cases} \tag{3.3}$$

A few remarks are in order:

1. If one chooses $\beta_k = 0$, then this corresponds to an inexact forward-backward descent [14], with iterates which will converge linearly to X_K, yet with a much worse contraction rate, that is, of order $1 - q$.

2. On the other hand, if, as in Theorem (2.3), one freezes the points $x_k, \bar{x}_k$ in (3.3) to some fixed approximation x, then using as before $\beta_k = (1-\sqrt{q})/(1-\sqrt{q})$ will yield a result similar to Theorem 2.3 (one needs to estimate the distance between the minimizer of a perturbed problem, defined by replacing x_K with x in (3.2), and the solution of (3.2), see for instance Section 6.1 for possible ways to do so.
3. An acceleration in (3.3) is possible but has to be carefully analysed, following the analysis for inexact accelerated forward-backward schemes such as in [2, 3], and one has to expect a rate in between the optimal rate and the one obtained with $\beta_k \equiv 0$.

4. Convex functions, prox operator. We gather here some more or less standard results on convex function, conjugates, Moreau-Yosida regularization and proximity operators.

4.1. Moreau's proximity operator. Let g convex, proper, lower semicontinous in $\mathbb{R}^d$, with values in $]-\infty, +\infty]$, we let $\operatorname{dom} g := \{g < +\infty\}$. We recall that the subgradient is defined as $\partial g(x) = \{p \in \mathbb{R}^d : g(y) \geq g(x) + p \cdot (y-x) \forall y \in \mathbb{R}^d\}$, it is nonempty for any x in the interior of $\operatorname{dom} g$. We define for $\tau > 0$:

$$\operatorname{prox}_{\tau g}(x) = \arg\min_{y \in \mathbb{R}^d} g(y) + \frac{\|y-x\|^2}{2\tau} = (I + \tau \partial g)^{-1}(x).$$

It is standard that it coincides with the resolvent of the maximal-monotone operator $\tau \partial g$, in particular it is one-Lipschitz, and firmly non-expansive, see for instance [4] for details.

Defining, for $\tau > 0$, the Yosida approximation:

$$g_\tau(x) = \min_{y \in \mathbb{R}^d} g(y) + \frac{\|y-x\|^2}{2\tau}$$

it is also well known (and easy to show) that g_τ is convex and C^1 with full domain and with $(1/\tau)$-Lipschitz gradient, in addition,

$$\nabla g_\tau(x) = \frac{x - \operatorname{prox}_{\tau g}(x)}{\tau} \in \partial g(\operatorname{prox}_{\tau g}(x)). \tag{4.1}$$

In particular, one sees that ∇g_τ is differentiable at the same points as $\operatorname{prox}_{\tau g}$.

Note that with our assumptions on g one has, for $p \in \mathbb{R}^d$,

$$(g_\tau)^*(p) = g^*(p) + \frac{\tau}{2}\|p\|^2$$

and the celebrated identity (of Moreau):

$$x = \operatorname{prox}_{\tau g}(x) + \tau \operatorname{prox}_{\frac{1}{\tau} g^*}\left(\tfrac{x}{\tau}\right). \tag{4.2}$$

We deduce from (4.1), (4.2) that:

$$\nabla g_\tau(x) = \operatorname{prox}_{\frac{1}{\tau} g^*}\left(\tfrac{x}{\tau}\right), \quad \nabla (g^*)_{\frac{1}{\tau}}\left(\tfrac{x}{\tau}\right) = \operatorname{prox}_{\tau g}(x). \tag{4.3}$$

4.2. Regularity of the Moreau-Yosida regularization. For our convergence proofs, we will need to use that the regularized function g_τ preserves some of the regularity properties of g, and in particular one has:

Lemma 4.1. *Let $g : \mathbb{R}^d \to \mathbb{R}$ be convex and locally $C^{2,\alpha}$. Let $\tau > 0$. Then g_τ is locally $C^{2,\alpha}$ (and as a consequence $\operatorname{prox}_{\tau g}$, $\operatorname{prox}_{\frac{1}{\tau}g^*}$ are locally $C^{1,\alpha}$).*

Proof. Let $R > 0$. A first remark is that $\operatorname{prox}_{\tau g}(\overline{B}(0,R))$ is a compact set in $\mathbb{R}^d$, so that by assumption, one has, for all $x, y \in B(0,R)$:

$$\|D^2 g(\operatorname{prox}_{\tau g}(x)) - D^2 g(\operatorname{prox}_{\tau g}(y))\| \le \omega(\|\operatorname{prox}_{\tau g}(x) - \operatorname{prox}_{\tau g}(y)\|) \le \omega(\|x-y\|)$$

for some modulus of continuity $\omega(t)$ (with our assumption, of the form ct^α); in addition,

$$C := \max_{x \in \overline{B}(0,R)} \|D^2 g(\operatorname{prox}_{\tau g}(x)\| < +\infty.$$

For $x \in B(0,R)$, $\nabla g_\tau(x) = \nabla g(\operatorname{prox}_{\tau g}(x))$, *cf.* (4.1). Let $y = \operatorname{prox}_{\tau g}(x)$ so that $x = y + \tau \nabla g(y) = y + \tau \nabla g_\tau(x)$. Considering $x_s = x + s\xi$, $s > 0$ and $y_s = y + s\eta_s = \operatorname{prox}_{\tau g}(x_s)$ one has that $\eta_s \le |\xi|$ so that $\nabla g_\tau(x_s) = \nabla g(y_s) = \nabla g(y) + sD^2 g(y) \cdot \eta_s + o(s)$. In addition, $y_s + \tau \nabla g(y_s) = x_s$ and in particular $\eta_s + \tau D^2 g(y) \cdot \eta_s + o(1) = \xi$, that is, $\eta_s = (I + \tau D^2 g(y))^{-1}(\xi + o(1))$. Hence $\lim_{s\to 0} \eta_s = (I + \tau D^2 g(y))^{-1}\xi$, and it follows that:

$$\lim_{s\to 0} \frac{\nabla g_\tau(x + s\xi) - \nabla g_\tau(x)}{s} = D^2 g(y) \cdot \eta = D^2 g(y)(I + \tau D^2 g(y))^{-1}\xi.$$

We deduce: $D^2 g_\tau(x) = D^2 g(y)(I + \tau D^2 g(y))^{-1}$. Let now $x, x' \in B(0,R)$ and $y = \operatorname{prox}_{\tau g}(x)$, $y' = \operatorname{prox}_{\tau g}(x')$. One has

$$\begin{aligned}
&\|D^2 g_\tau(x) - D^2 g_\tau(x')\| \\
&\le \|D^2 g(y) - D^2 g(y')\|\|(I + \tau D^2 g(y))^{-1}\| + \|D^2 g(y')\|\|(I + \tau D^2 g(y))^{-1} - (I + \tau D^2 g(y'))^{-1}\| \\
&\le \omega(\|y - y'\|) + C\|(I + \tau D^2 g(y))^{-1} - (I + \tau D^2 g(y'))^{-1}\| \le (1 + C)\omega(\|x - x'\|)
\end{aligned}$$

where we have used that for A, B positive semi-definite matrices, one has:

$$\|(I + A)^{-1} - (I + B)^{-1}\| = \|(I + A)^{-1}(I + B - (I + A))(I + B)^{-1}\| \le \|B - A\|. \qquad \blacksquare$$

Remark 4. *Observe that the conclusion still holds, in case g is not defined everywhere, provided that for $x \in \partial \operatorname{dom} g$, either g is $C^{2,\alpha}$ up to x, or $\lim_{x' \to x} \|\nabla g(x')\| = +\infty$. The proof shows, also, that if g is merely of class C^2 (locally), then also g_τ is.*

5. Proof of Theorem 2.1. We start with a collection of more or less standard results on the differentiability of convex functions. First, recall that if $g : \mathbb{R}^d \to \mathbb{R}$ is strictly convex then in particular ∂g is injective, $\nabla g(\mathbb{R}^d) = \operatorname{dom} \partial g^*$ has nonempty interior (otherwise g would be "flat" in the orthogonal direction), and g^* is C^1 in the interior of $\operatorname{dom} g^*$, with $\nabla g^*(p) = x$ for all $p \in \partial g(x)$ and all $x \in \mathbb{R}^d$. If g is in addition (γ-)strongly convex ($g(x) - \gamma|x|^2/2$ is convex for some $\gamma > 0$), then ∂g is (strongly monotone and) surjective and ∇g^* is $(1/\gamma)$-Lipschitz with full domain.

According to Alexandrov's theorem [22, §6.4], a convex function $g : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is twice differentiable $\mathcal{L}^n$-almost everywhere (in the interior of its domain), in the sense that it admits a second order Taylor expansion near almost every point. More refined proofs (see in particular [37, §1.2], and the following papers [15, 30]) show that the subgradient ∂g is differentiable almost everywhere, with a symmetric gradient. We say that $D^2 g(x)$ *exists* if x is a point of differentiability of ∂g (in which case, $\partial g(x)$ is single-valued so that also $\nabla g(x)$ exists) and is symmetric; the version of Alexandrov's theorem in [30] shows that it is the case almost everywhere in the domain of g.

Lemma 5.1. *Let $g : \mathbb{R}^d \to \mathbb{R}$ be strongly convex. Then*

1. *$D^2 g(x) = \nabla(\nabla g)(x)$ exists and is nonsingular ($\det D^2 g(x) \neq 0$) a.e. in $\operatorname{dom} g$;*
2. *If $D^2 g$ exists and is nonsingular at x, then $D^2 g^*(\nabla g(x))$ exists and is $D^2 g(x)^{-1}$.*
3. *If $D^2 g^*(p)$ exists and is nonsingular, $x = \nabla g^*(p)$, then $p = \nabla g(x)$, and $D^2 g(x)$ exists (and is nonsingular).*

Proof. The first statement is, as said, a version of Alexandrov's theorem The gradient itself, ∇g, is defined also almost everywhere, yet in the statement one can also use the subgradient and the definition of $\nabla(\partial g)$ provided in [37]. The fact that $D^2 g(x)$ is nonsingular a.e. follows from the strong convexity, as if g is γ-convex then clearly $D^2 g \geq \gamma I$ (using for instance that $g(\cdot) - \gamma\|\cdot\|^2/2$ is convex).

One has that $D^2 g(x)$ exists if and only if there is a symmetric matrix (obviously denoted $D^2 g(x)$) such that for almost all y near x, $\nabla g(y) = \nabla g(x) + D^2 g(x) \cdot (y - x) + o(\|y - x\|)$, or following [37, Def. 2.1], $\partial g(y) \subset B(\nabla g(x) + D^2 g(x) \cdot (y - x), o(\|y - x\|))$. Hence for $\eta > 0$ small,

$$\begin{aligned}\nabla g(x) + \eta \subset \partial g(\nabla g^*(\nabla g(x) + \eta)) \\ = \nabla g(x) + D^2 g(x) \cdot (\nabla g^*(\nabla g(x) + \eta) - x) + o(\|\nabla g^*(\nabla g(x) + \eta) - x\|).\end{aligned}$$

Since ∇g^* is Lipschitz, $\|\nabla g^*(\nabla g(x) + \eta) - x\| = O(\|\eta\|)$, so that:

$$\eta = D^2 g(x) \cdot (\nabla g^*(\nabla g(x) + \eta) - x) + o(\|\eta\|)$$

and it follows, since $D^2 g(x)$ is nonsingular, that

$$\nabla g^*(\nabla g(x) + \eta) = x + (D^2 g(x))^{-1}\eta + o(\|\eta\|) = \nabla g^*(\nabla g(x)) + (D^2 g(x))^{-1}\eta + o(\|\eta\|)$$

which shows the thesis.

The last statement is proved similarly. Let $x = \nabla g^*(p)$ so that $p \in \partial g(x)$. The version of the local inversion theorem in [30, Thm. 4.1] applied to the continuous function ∇g^* shows that since $D^2 g^*(p)$ is non-singular, there is a neighborhood B_x of x and a neighborhood B_p of p such that for each $y \in B_x$, there exists $q \in B_p$ with $\nabla g^*(q) = y$, so that $q \in \partial g(y)$; in addition $C^{-1}\|q - p\| \leq \|y - x\| \leq C\|q - p\|$ for some constant C, depending only on the eigenvalues of $D^2 g^*(p)$.

In particular, $\partial g(y) \to p$ as $y \to x$, and one deduces that $\partial g(x)$ is single-valued, in other words, $p = \nabla g(x)$. We then write, for $q \in \partial g(y)$:

$$y = \nabla g^*(q) = \nabla g^*(\nabla g(x)) + D^2 g^*(\nabla g(x)) \cdot (q - \nabla g(x)) + o(\|q - \nabla g(x)\|)$$

and thanks to the inversion theorem above, observe that $o(\|q-\nabla g(x)\|) = o(\|y-x\|)$. We deduce that $q = \nabla g(x) + D^2 g^*(\nabla g(x))^{-1}\cdot(y-x) + o(\|y-x\|)$, so that $D^2 g(x) = D^2 g^*(\nabla g(x))^{-1}$ exists. ∎

Lemma 5.2. *Let g be convex with Lipschitz gradient, $\tau > 0$. Then: $\nabla \operatorname{prox}_{\tau g}(x)$ exists and is symmetric, if and only if $D^2 g(\operatorname{prox}_{\tau} g(x))$ exists. One has in addition:*

$$\nabla \operatorname{prox}_{\tau g}(x) = \tfrac{1}{\tau}\left[\tfrac{1}{\tau} I + D^2 g(\operatorname{prox}_{\tau g}(x))\right]^{-1}.$$

The proof can be deduced from [15, Appendix]. We sketch it for convenience.

Proof. $D^2 g(\operatorname{prox}_{\tau g}(x))$ exists if and only if $D^2 g(\operatorname{prox}_{\tau g}(x)) + \frac{1}{\tau} I = D^2(g + \frac{\|\cdot\|^2}{2\tau})(\operatorname{prox}_{\tau g}(x))$ exists. Then Lemma 5.1 implies that $D^2(g^*)_{\frac{1}{\tau}}(\nabla g(\operatorname{prox}_{\tau g}(x)) + \frac{1}{\tau}\operatorname{prox}_{\tau g}(x)) = D^2(g^*)_{\frac{1}{\tau}}(\frac{x}{\tau})$ (*cf.* (4.1)) exists, and is $(D^2 g(\operatorname{prox}_{\tau g}(x)) + \frac{1}{\tau})^{-1}$. The converse also is true, using that ∇g is Lipschitz, so that $(g^*)_{1/\tau}$ is strongly convex, and Lemma 5.1 again. The conclusion follows from (4.3). ∎

Lemma 5.3. *Let g be strongly convex, and let $p \in \partial g(x)$. Then $D^2 g^*(p)$ exists if and only if $\nabla \operatorname{prox}_{\tau g}(x+\tau p)$ exists (and is symmetric). Precisely, one has*

$$\nabla \operatorname{prox}_{\tau g}(x+\tau p) = I - \left[I + \tfrac{1}{\tau} D^2 g^*(p)\right]^{-1}.$$

Proof. Observe that $x + \tau p \in x + \tau \partial g(x)$ hence $x = \operatorname{prox}_{\tau g}(x+\tau p)$. By Moreau's identity, $\nabla \operatorname{prox}_{\tau g}(x+\tau p)$ exists if and only if $\nabla \operatorname{prox}_{\frac{1}{\tau} g^*}(\frac{x}{\tau} + p)$ does. By the previous lemma this is true if and only if $D^2 g^*(\operatorname{prox}_{\frac{1}{\tau} g^*}(\frac{x}{\tau} + p))$ exists. Using (4.2), $\operatorname{prox}_{\frac{1}{\tau} g^*}(\frac{x}{\tau} + p) = \frac{1}{\tau}(x + \tau p - \operatorname{prox}_{\tau g}(x+\tau p)) = p$. The first part of the thesis follows. One has also, using again (4.2) and the previous Lemma,

$$\begin{aligned}\nabla \operatorname{prox}_{\tau g}(x+\tau p) &= I - \nabla \operatorname{prox}_{\frac{1}{\tau} g^*}(x+\tau p) \\ &= I - \tau\left[\tau I + D^2 g^*\left(\operatorname{prox}_{\frac{1}{\tau} g^*}\left(\tfrac{x}{\tau} + p\right)\right)\right]^{-1} = I - \left[I + \tfrac{1}{\tau} D^2 g^*(p)\right]^{-1}.\end{aligned}$$ ∎

Proof of Theorem 2.1. The first fixed point equation (2.4) reads

$$X = \nabla \operatorname{prox}_{\tau g}(x_K - \tau K^* y_K)\cdot\left[X - \tau(K^* Y + \nabla_x \ell(x_K, y_K))\right].$$

We have $-K^* y_K \in \partial g(x_K)$ by optimality of (x_K, y_K). Hence if we assume that g is strongly convex, and that the equation is well defined, that is, $\nabla \operatorname{prox}_{\tau g}(x_K - \tau K^* y_K)$ exists and in addition is symmetric, then thanks to Lemma 5.3 this equation is equivalent to:

$$X = X - \tau(K^* Y + \nabla_x \ell(x_K, y_K)) - \left[I + \tfrac{1}{\tau} D^2 g^*(-K^* y_K)\right]^{-1}\left(X - \tau(K^* Y + \nabla_x \ell(x_K, y_K))\right),$$

that is,

$$X = -D^2 g^*(-K^* y_K)\left(K^* Y + \nabla_x \ell(x_K, y_K)\right). \tag{5.1}$$

Equivalently, one finds that if the second equation in (2.4) makes sense and is true, then f has a second derivative at Kx and it holds:

$$Y = D^2 f(Kx_K)(KX + \nabla_y \ell(x_K, y_K)). \tag{5.2}$$

Now let $L \in \mathbb{R}^{m\times n}$ and assume (x_s, y_s) is the solution of the saddle-point problem $(\mathcal{S})$ for K replaced with $K + sL$, for $s > 0$ small. One has therefore $(K + sL)^* y_s + \partial g(x_s) \ni 0$, $-(K + sL)x_s + \partial f^*(y_x) \ni 0$. Denote $p_s := -(K + sL)^* y_s \in \partial g(x_s)$, $p = -K^* y_K \in \partial g(x_K)$, $q_s := (K + sL)x_s \in \partial f^*(y_s)$, $q = Kx_K \in \partial f^*(y_K)$, and denote also $\xi_s = (x_s - x_K)/s$, $\eta_s = (y_s - y_K)/s$. One has

$$K^*\eta_s + L^* y_s + \frac{p_s - p}{s} = 0, \quad -(K\xi_s + Lx_s) + \frac{q_s - q}{s} = 0 \tag{5.3}$$

Hence (multiplying the first equation by ξ_s, the second by η_s, using the strong convexity of g and f^* and summing we get:

$$\gamma\|\xi_s\|^2 + \delta\|\eta_s\|^2 \le -\xi_s \cdot (K^*\eta_s + L^* y_s) + \eta_s \cdot (K\xi_s + Lx_s) = (Lx_s)\cdot\eta_s - (L^* y_s)\cdot\xi_s$$

and it follows that ξ_s, η_s are uniformly bounded as $s \to 0$. As a consequence, along some subsequence $(s_i)_{i\ge 0}$, $s_i \to 0$, one has $\xi_{s_i} \to \xi$, $\eta_{s_i} \to s$ and

$$\lim_{i\to\infty} \frac{p_{s_i} - p}{s_i} = -(K^*\eta + L^* y_K), \quad \lim_{i\to\infty} \frac{q_{s_i} - q}{s_i} = K\xi + Lx_K.$$

In addition, we remark that since ∇g^* is differentiable at p (Lemma 5.3),

$$x_s = \nabla g^*(p_s) = \nabla g^*(p) + sD^2 g^*(p)\frac{p_s - p}{s} + o(s) = x + sD^2 g^*(p)\frac{p_s - p}{s} + o(s)$$

so that in the limit $i \to \infty$, one finds

$$\xi = -D^2 g^*(-K^* y_K)(K^*\eta + L^* y_K), \quad \eta = D^2 f(Kx_K)(K\xi + Lx_K). \tag{5.4}$$

We have

$$\lim_{i\to\infty} \frac{\mathcal{L}(K + s_i L) - \mathcal{L}(K)}{s_i} = \xi \cdot \nabla_x \ell(x_K, y_K) + \eta \cdot \nabla_y \ell(x_K, y_K) =: \Delta. \tag{5.5}$$

We compute, using (5.4) and (5.1)

$$\begin{aligned}\xi \cdot \nabla_x \ell(x_K, y_K) &= -D^2 g^*(-K^* y_k)(K^*\eta + L^* y_K)\cdot(K^* Y + \nabla_x \ell(x_K, y_K)) - \xi\cdot(K^* Y)\\ &= (K^*\eta + L^* y_K)\cdot X - \xi\cdot(K^* Y)\end{aligned}$$

and, using (5.2),

$$\begin{aligned}\eta \cdot \nabla_Y \ell(x_K, y_K) &= D^2 f(Kx_K)(K\xi + Lx_K)\cdot(KX + \nabla_y \ell(x_K, y_K)) - \eta\cdot(KX)\\ &= (K\xi + Lx_K)\cdot Y - \eta\cdot(KX).\end{aligned}$$

Summing, we deduce

$$\Delta = (K^*\eta + L^* y_K)\cdot X - \xi\cdot(K^* Y) + (K\xi + Lx_K)\cdot Y - \eta\cdot(KX) = y_K\cdot(LX) + Y\cdot(Lx_K).$$

In particular, the limit in (5.5) is independent on the sequence (s_i) and we deduce that $\mathcal{L}$ is differentiable at K, with $\nabla\mathcal{L}(K) = y_K \otimes X + Y \otimes x_K$. This proves Theorem 2.1. ∎

6. Error analysis and convergence. In this section we prove first Theorem 2.3, and then Theorem 2.2, which relies on a similar but slightly more complicated analysis. With respect to the previous result, we have to assume in addition that f and g^* are locally $C^{2,\alpha}$ for some parameter $\alpha > 0$. (We point out that a variant of Theorem 2.3 would still remain valid with a less precise modulus of continuity for the Hessians of f and g^*, as is clear from the proof.)

6.1. Proof of Theorem 2.3. We assume (x, y) and (x', y') are approximations of (x_K, y_K), with $\max\{\|x - x_K\|, \|x' - x_K\|, \|y - y_K\|, \|y' - y_K\|\} \leq \varepsilon$ for some $\varepsilon > 0$. The iterates defining (X^k, Y^k) can be written, using (4.1) and (4.2):

$$\begin{aligned} X^{k+1} &= \tfrac{1}{\tau} D^2(g^*)_{\frac{1}{\tau}}(\tfrac{x}{\tau} - K^* y) \cdot \Big(X^k - \tau(K^* Y^k + \nabla_x \ell(x, y))\Big) \\ Y^{k+1} &= \tfrac{1}{\sigma} D^2 f_{\frac{1}{\sigma}}(\tfrac{y'}{\sigma} + K\bar{x}') \cdot \Big(Y^k + \sigma(K\bar{X}^{k+1} + \nabla_y \ell(x', y'))\Big)\,. \end{aligned}$$

with $\bar{X}^{k+1} = X^{k+1} + \theta(X^{k+1} - X^k)$. Let us introduce (X, Y) the fixed point of the problem:

$$\begin{aligned} X &= \tfrac{1}{\tau} D^2(g^*)_{\frac{1}{\tau}}(\tfrac{x}{\tau} - K^* y) \cdot (X - \tau(K^* Y + \nabla_x \ell(x, y))) \\ Y &= \tfrac{1}{\sigma} D^2 f_{\frac{1}{\sigma}}(\tfrac{y'}{\sigma} + Kx') \cdot \big(Y + \sigma(KX + \nabla_y \ell(x', y'))\big)\,. \end{aligned}$$

Then, the iterates are solving a standard primal-dual algorithm optimizing a strongly convex / strongly concave saddle-point problem with solution this fixed point (X, Y). We know from [8, 10] that for a good choice of the parameters, such as $\mu = 2\sqrt{\gamma\delta}/\|K\|$, $\theta \in [1/(1+\mu), 1]$ $\tau = \mu/(2\gamma)$ and $\sigma = \mu/(2\delta)$ we obtain, letting $\omega = (1+\theta)/(2+\mu) \leq \theta$, that

$$\gamma\|X^k - X\|^2 + (1-\omega)\delta\|Y^k - Y\|^2 \leq C\omega^k. \tag{6.1}$$

The next step is to estimate $\|X - X_K\|$ and $\|Y - Y_K\|$. Recall that (X_K, Y_K) satisfy

$$\begin{aligned} X_K &= \tfrac{1}{\tau} D^2(g^*)_{\frac{1}{\tau}}(\tfrac{x_K}{\tau} - K^* y_K) \cdot (X_K - \tau(K^* Y_K + \nabla_x \ell(x_K, y_K))) \\ Y_K &= \tfrac{1}{\sigma} D^2 f_{\frac{1}{\sigma}}(\tfrac{y_K}{\sigma} + Kx_K) \cdot (Y_K + \sigma(KX_K + \nabla_y \ell(x_K, y_K)))\,. \end{aligned}$$

Substracting the equations for X_K from the equation for X^{k+1}, we get:

$$\begin{aligned} X - X_K = {} & \\ & \tfrac{1}{\tau} D^2(g^*)_{\frac{1}{\tau}}(\tfrac{x}{\tau} - K^* y) \cdot (X - X_K - \tau K^*(Y - Y_K) - \tau(\nabla_x \ell(x, y)) - \nabla_x \ell(x_K, y_K))) \\ & + \tfrac{1}{\tau}\Big[D^2(g^*)_{\frac{1}{\tau}}(\tfrac{x}{\tau} - K^* y) - D^2(g^*)_{\frac{1}{\tau}}(\tfrac{x_K}{\tau} - K^* y_K)\Big] \cdot (X_K - \tau(K^* Y_K + \nabla_x \ell(x_K, y_K)))\,. \end{aligned} \tag{6.2}$$

One has by assumption that $\|\nabla_x \ell(x, y) - \nabla_x \ell(x_K, y_K)\| \leq C\varepsilon$, for some constant C depending on ℓ (near (x_K, y_K), as we assumed ℓ is C^1), and thanks to Lemma 4.1,

$$\Big\|D^2(g^*)_{\frac{1}{\tau}}(\tfrac{x}{\tau} - K^* y) - D^2(g^*)_{\frac{1}{\tau}}(\tfrac{x_K}{\tau} - K^* y_K)\Big\| \leq C\epsilon^\alpha.$$

Hence, (6.2) can be rewritten as

$$X - X_K = \tfrac{1}{\tau} D^2(g^*)_{\frac{1}{\tau}}(\tfrac{x}{\tau} - K^* y) \cdot (X - X_K - \tau(K^*(Y - Y_K) + u_X)) + v_X,$$

where the error terms satisfy $\|u_X\| \leq C\epsilon$ and $\|v_X\| \leq C\epsilon^\alpha$. Similarly,

$$Y - Y_K = \tfrac{1}{\sigma} D^2 f_{\frac{1}{\sigma}}(\tfrac{y'}{\sigma} - K^* x') \cdot (Y - Y_K + \sigma(K(X - X_K) + u_Y)) + v_Y,$$

with obvious notation and the same control on the error terms. Letting then $\tilde{X} = X - v_X$ and $\tilde{Y} = Y - v_Y$, this is the same as:

$$\tilde{X} - X_K = \tfrac{1}{\tau} D^2(g^*)_{\frac{1}{\tau}}(\tfrac{x}{\tau} - K^* y) \cdot \left(\tilde{X} - X_K - \tau(K^*(\tilde{Y} - Y_K) + e_X)\right), \tag{6.3}$$

with $e_X = u_X + K^* v_Y - v_X/\tau$, and

$$\tilde{Y} - Y_K = \tfrac{1}{\sigma} D^2 f_{\frac{1}{\sigma}}(\tfrac{y'}{\sigma} - K^* x') \cdot \left(Y - Y_K + \sigma(K(\tilde{X} - X_K) + e_Y)\right) \tag{6.4}$$

with $e_Y = u_Y + K v_X + v_Y/\sigma$.

We then recall that if A is a semidefinite positive matrix in $\mathbb{R}^d$ and $\eta = A\xi$, then $\xi \cdot \eta \geq \|\eta\|^2/\|A\|$, and we get:

$$\begin{aligned}(1 + \tau\gamma)\|\tilde{X} - X_K\|^2 &\leq \left(\tilde{X} - X_K - \tau(K^*(\tilde{Y} - Y_K) + e_X)\right) \cdot (\tilde{X} - X_K)\\ (1 + \sigma\delta)\|\tilde{Y} - Y_K\|^2 &\leq \left(\tilde{Y} - Y_K + \sigma(K(\tilde{X} - X_K) + e_Y)\right) \cdot (\tilde{Y} - Y_K).\end{aligned}$$

Summing and rearranging, we deduce

$$\gamma\|\tilde{X} - X_K\|^2 + \delta\|\tilde{Y} - Y_K\|^2 \leq -e_X \cdot (\tilde{X} - X_K)) + e_Y \cdot (\tilde{Y} - Y_K) \leq \frac{\|e_X\|^2}{\gamma} + \frac{\|e_Y\|^2}{\delta} \leq C\varepsilon^{2\alpha}.$$

By definition of $(\tilde{X}, \tilde{Y})$, we see that a similar error control holds for (X, Y). Theorem 2.3 is obtained by combining this estimate together with (6.1).

6.2. Convergence of the Piggyback algorithm. The proof of the piggyback algorithm is almost the same, only slightly more complicated as it corresponds to solving directly the saddle-point problem defining (X_K, Y_K), but with an inexact primal-dual method, such as studied in [44]. Again, an obvious observation is that, thanks to [8, Thm. 3], choosing $\mu = 2\sqrt{\gamma\delta}/\|K\|$, $\theta \in [1/(1+\mu), 1]$, $\tau = \mu/(2\gamma)$ and $\sigma = \mu/(2\delta)$ we have, for $\omega = (1+\theta)/(2+\mu)$,

$$\gamma\|x^k - x_K\|^2 + (1-\omega)\delta\|y^k - y_K\|^2 \leq C\omega^k \left(\gamma\|x^0 - x_K\|^2 + (1-\omega)\delta\|y^0 - y_K\|^2\right),$$

so that we have the linear convergence $\|x^k - x_K\| + \|y^k - y_K\| \leq C\omega^{k/2}$.

Substituting as before the iterations for X^{k+1} and and the fixed-point equation for X_K, we obtain now:

$$\begin{aligned}&X^{k+1} - X_K =\\ &\tfrac{1}{\tau} D^2(g^*)_{\frac{1}{\tau}}(\tfrac{x^k}{\tau} - K^* y^k) \cdot \left(X^k - X_K - \tau K^*(Y^k - Y_K) - \tau(\nabla_x \ell(x^k, y^k)) - \nabla_x \ell(x_K, y_K))\right)\\ &+ \tfrac{1}{\tau}\left[D^2(g^*)_{\frac{1}{\tau}}(\tfrac{x^k}{\tau} - K^* y^k) - D^2(g^*)_{\frac{1}{\tau}}(\tfrac{x_K}{\tau} - K^* y_K)\right] \cdot (X_K - \tau(K^* Y_K + \nabla_x \ell(x_K, y_K))).\end{aligned} \tag{6.5}$$

One has $\|\nabla_x \ell(x^k, y^k) - \nabla_x \ell(x_K, y_K)\| \leq C\omega^{k/2}$ for some constant C depending on ℓ (near (x_K, y_K)), and thanks to Lemma 4.1,

$$\left\| D^2(g^*)_{\frac{1}{\tau}}(\tfrac{x^k}{\tau} - K^* y^k) - D^2(g^*)_{\frac{1}{\tau}}(\tfrac{x_K}{\tau} - K^* y_K) \right\| \leq C\omega^{k\alpha/2}.$$

Hence, (6.5) can be rewritten as

$$X^{k+1} - X_K = \tfrac{1}{\tau} D^2(g^*)_{\frac{1}{\tau}}(\tfrac{x^k}{\tau} - K^* y^k) \cdot \left(X^k - X_K - \tau K^*(Y^k - Y_K) + u_X^k \right) + v_X^k,$$

where the error terms satisfy global bounds $\|u_X^k\| \leq C\omega^{k/2}$ and $\|v_X^k\| \leq C\omega^{k\alpha/2}$. Similarly,

$$Y^{k+1} - Y_K = \tfrac{1}{\sigma} D^2 f_{\frac{1}{\sigma}}(\tfrac{y^k}{\sigma} - K^* \bar{x}^k) \cdot \left(Y^k - Y_K + \sigma K(\bar{X}^{k+1} - X_K) + u_Y^k \right) + v_Y^k,$$

with obvious notation and the same control on the error terms.

One finds as previously, after taking the scalar product of (6.2) with $X^{k+1} - X_k - v_k$,

$$(1 + \tau\gamma)\|X^{k+1} - X_K - v_X^k\|^2 \leq \left(X^k - X_K - \tau(K^*(Y^k - Y_K) + u_X^k) \right) \cdot \left(X^{k+1} - X_K - v_X^k \right).$$

We let, for all $k \geq 1$, $e_X^k := u_X^k - v_X^{k-1}/\tau + K^* v_Y^{k-1}$ and denote $\tilde{X}^k = X^k - v_X^{k-1}$, $\tilde{Y}^k = Y^k - v_Y^{k-1}$, and we fall back in the situation of the smoother case which was considered in [11, Appendix]:

$$\begin{aligned}(1 + \tau\gamma)\|\tilde{X}^{k+1} - X_K\|^2 &\leq \left(\tilde{X}^k - X_K - \tau K^*(\tilde{Y}^k - Y_K) + \tau e_X^k \right) \cdot \left(\tilde{X}^{k+1} - X_K \right) \\ &= \tfrac{1}{2}\|\tilde{X}^k - X_K\|^2 + \tfrac{1}{2}\|\tilde{X}^{k+1} - X_K\|^2 - \tfrac{1}{2}\|\tilde{X}^{k+1} - \tilde{X}^k\|^2 \\ &\quad - \tau(\tilde{Y}^k - Y_K) \cdot K(\tilde{X}^{k+1} - X_K) + \tau e_X^k \cdot (\tilde{X}^{k+1} - X_K).\end{aligned}$$

We deduce:

$$\begin{aligned}(6.6)\quad &\tfrac{1+\mu}{2\tau}\|\tilde{X}^{k+1} - X_K\|^2 + \tfrac{1}{2\tau}\|\tilde{X}^{k+1} - \tilde{X}^k\|^2 \\ &\qquad \leq \tfrac{1}{2\tau}\|\tilde{X}^k - X_K\|^2 - (\tilde{Y}^k - Y_K) \cdot K(\tilde{X}^{k+1} - X_K) + e_X^k \cdot (\tilde{X}^{k+1} - X_K).\end{aligned}$$

In the same way, denoting $e_Y^k := u_Y^k + v_Y^{k-1}/\sigma + K^*(v_X^k + \theta(v_X^k - v_X^{k-1}))$, we have:

$$\begin{aligned}(6.7)\quad &\tfrac{1+\mu}{2\sigma}\|\tilde{Y}^{k+1} - Y_K\|^2 + \tfrac{1}{2\sigma}\|\tilde{Y}^{k+1} - \tilde{Y}^k\|^2 \\ &\qquad \leq \tfrac{1}{2\sigma}\|\tilde{Y}^k - Y_K\|^2 + (\tilde{Y}^{k+1} - Y_K) \cdot K(\bar{\tilde{X}}^{k+1} - X_K) + e_Y^k \cdot (\tilde{Y}^{k+1} - Y_K),\end{aligned}$$

with obviously $\bar{\tilde{X}}^{k+1} = \tilde{X}^{k+1} + \theta(X^{k+1} - X^k)$.

Now, we follow [44], where the techniques of [8, 10] are adapted to an inexact setting, with a control of the errors. The algorithm in [8] is presented a bit differently, actually, the over-relaxation step is performed before the two updates. In this form, the analysis of the linearly converging version is much easier. We therefore combine the inequalities (6.6) and (6.7) at respectively the steps k and $k-1$. We start by letting for all $k \geq 1$, $\Delta_k :=$

$\|\tilde{X}^k - X_K\|^2/(2\tau) + \|\tilde{Y}^{k-1} - Y_K\|^2/(2\sigma)$, then we sum the estimates (at rank k for X and $k-1$ for Y) to obtain:

$$(1+\mu)\Delta_{k+1} + \tfrac{1}{2\tau}\|\tilde{X}^{k+1} - \tilde{X}^k\|^2 + \tfrac{1}{2\sigma}\|\tilde{Y}^k - \tilde{Y}^{k-1}\|^2 \\ \leq \Delta_k - (\tilde{Y}^k - Y_K)\cdot K(\tilde{X}^{k+1} - \bar{\tilde{X}}^k). + e_X^k \cdot (\tilde{X}^{k+1} - X_K) + e_Y^{k-1}\cdot(\tilde{Y}^k - Y_K),$$

and thus:

$$(1+\mu)\Delta_{k+1} + (\tilde{Y}^k - Y_K)\cdot K(\tilde{X}^{k+1} - \tilde{X}^k) + \tfrac{1}{2\tau}\|\tilde{X}^{k+1} - \tilde{X}^k\|^2 \\ \leq \Delta_k + \theta(\tilde{Y}^{k-1} - Y_K)\cdot K(\tilde{X}^k - \tilde{X}^{k-1}) + \tfrac{\theta}{2\tau}\|\tilde{X}^k - \tilde{X}^{k-1}\|^2 \\ + e_X^k\cdot(\tilde{X}^{k+1} - X_K) + e_Y^{k-1}\cdot(\tilde{Y}^k - Y_K).$$

To control the error terms, we observe that:

$$e_X^k\cdot(\tilde{X}^{k+1} - X_K) + e_Y^{k-1}\cdot(\tilde{Y}^k - Y_K) \\ \leq \frac{\mu}{4\tau}\|\tilde{X}^{k+1} - X_K\|^2 + \frac{\tau}{\mu}\|e_X^k\|^2 + \frac{\mu}{4\sigma}\|\tilde{Y}^k - Y_K\|^2 + \frac{\sigma}{\mu}\|e_Y^{k-1}\|^2 \leq \frac{\mu}{2}\Delta_{k+1} + C\omega^{k\alpha}, \tag{6.8}$$

obtaining eventually:

$$(1+\tfrac{\mu}{2})\Delta_{k+1} + (\tilde{Y}^k - Y_K)\cdot K(\tilde{X}^{k+1} - \tilde{X}^k) + \tfrac{1}{2\tau}\|\tilde{X}^{k+1} - \tilde{X}^k\|^2 \\ \leq \Delta_k + \theta(\tilde{Y}^{k-1} - Y_K)\cdot K(\tilde{X}^k - \tilde{X}^{k-1}) + \tfrac{\theta}{2\tau}\|\tilde{X}^k - \tilde{X}^{k-1}\|^2 + C\omega^{k\alpha}$$

To simplify, we choose $\theta = 1/(1+\mu/2) = 2/(2+\mu)$. We remark that the left-hand side term in the previous expression is always non-negative (using $\sqrt{\tau\sigma}\|K\| \leq 1$): hence the inequality remains valid if it is multiplied by a factor less than one. Introducing $\rho = \max\{\theta, \omega^{\alpha/2}\} < 1$, one therefore has:

$$\rho^{-1}\left(\Delta_{k+1} + \theta(\tilde{Y}^k - Y_K)\cdot K(\tilde{X}^{k+1} - \tilde{X}^k) + \tfrac{\theta}{2\tau}\|\tilde{X}^{k+1} - \tilde{X}^k\|^2\right) \\ \leq \Delta_k + \theta(\tilde{Y}^{k-1} - Y_K)\cdot K(\tilde{X}^k - \tilde{X}^{k-1}) + \tfrac{\theta}{2\tau}\|\tilde{X}^k - \tilde{X}^{k-1}\|^2 + C\omega^{k\alpha}$$

Summing again from $k=1$ to $n-1$ after multiplication with ρ^{-k}, we get:

$$\rho^{-n}\left(\Delta_n + \theta(\tilde{Y}^{n-1} - Y_K)\cdot K(\tilde{X}^n - \tilde{X}^n) + \tfrac{\theta}{2\tau}\|\tilde{X}^n - \tilde{X}^{n-1}\|^2\right) \\ \leq \Delta_1 + \theta(\tilde{Y}^0 - Y_K)\cdot K(\tilde{X}^1 - \tilde{X}^0) + \tfrac{\theta}{2\tau}\|\tilde{X}^1 - \tilde{X}^0\|^2 + C\sum_{k=1}^{n-1}\rho^{-k}\omega^{k\alpha}$$

Our choice of ρ guarantees that the last sum is finite, bounded by $C/(1-\omega^{\alpha/2})$. We deduce that

$$\Delta_n + \theta(\tilde{Y}^{n-1} - Y_K)\cdot K(\tilde{X}^n - \tilde{X}^n) + \tfrac{\theta}{2\tau}\|\tilde{X}^n - \tilde{X}^{n-1}\|^2 \leq C\rho^n$$

for some constant $C > 0$, and in particular, using again that $\sqrt{\tau\sigma}\|K\| \leq 1$ we deduce that

$$\tfrac{1}{2\tau}\|\tilde{X}^n - X_K\|^2 + \tfrac{1-\theta}{2\sigma}\|\tilde{Y}^{n-1} - Y_K\|^2 \leq C\rho^n$$

showing that $(\tilde{X}^n, \tilde{Y}^n)$, and hence also (X^n, Y^n), converges linearly to (X_K, Y_K).

7. Application to Shearlets.

We close this paper by applying the proposed piggyback algorithm to the problem of learning an optimized shearlet transform, which is a wavelet-like transform but somewhat optimized for the task of recovering piecewise smooth images with smooth boundaries.

The application of shearlets in image processing was motivated by the shortcomings of wavelets, which despite being a very powerful tool in signal processing are not well suited for images due to their anisotropic nature. Rotations can capture the anisotropy of images, but they are hard to digitize on a discrete grid, whereas shearing operations used in shearlets can be faithfully discretized [32]. Using a piggyback algorithm, the parameters of a shearlet system can be learned for solving a convex minimization problem.

7.1. Digital Shearlet Transform.

To setup a shearlet system the frequency domain is divided into a cone-like partition, which avoids an extensive elongation of shearlets at higher shearing levels. We use a non-separable shearlet generator to obtain a wedge-like support in the frequency domain, which was first proposed by Lim [34] and discussed in detail by Kutyniok et al. [33]. To construct a digital shearlet system, scaling $j > 0$, translations $m \in \mathbb{Z}^2$, and shearing $|k| \leq \lceil 2^{j/2} \rceil$ have to be set, where $j/2 \in \mathbb{Z}$ is assumed, else $\lfloor j/2 \rfloor$ is taken. A 1D low-pass filter h_1 and a 2D directional filter P are the basic building blocks. The 1D filters $h_{J-j/2}$ and g_{J-j} are derived from h_1 in a wavelet multiresolution analysis which are tensorized to yield $W_j = g_{J-j} \otimes h_{J-j/2}$, and p_j are the Fourier coefficients of P. The digital shearlet $\psi^d_{j,k} \in \mathbb{C}^{M \times N}$ for a scale j and shearing k is then computed by

$$\psi^d_{j,k} = \left[\left(S_k \big((p_j * W_j)_{\uparrow 2^{j/2}} *_1 h_{j/2} \big) \right) *_1 \bar{h}_{j/2} \right]_{\downarrow 2^{j/2}},$$

where up-sampling and down-sampling operations ensure that the shearing operator $S_{k/2^{j/2}}$ is well defined on the discrete grid, since shearlets are generated by $\psi_{j,k}(\cdot) = \psi_{j,0}(S_{k/2^{j/2}} \cdot)$. The flipped filter $\bar{h}(n) = h(-n)$ indicates the reversal of the convolution. Finally, the digital shearlet transform applied to an image $u \in \mathbb{R}^{M \times N}$ is given by

$$DST_{j,k}(u) = \overline{\psi^d_{j,k}} * u.$$

7.2. Saddle-Point Formulation.

We consider an imaging problem as in ($\mathcal{P}$) with a shearlet operator $S : \mathcal{X} \to \mathcal{Y}$, $u, z \in \mathcal{X} \simeq \mathbb{R}^{M \times N}$, and $p \in \mathcal{Y} \simeq \mathbb{R}^{n \times M \times N}$. The shearlet regularized minimization problem is defined as

$$\min_u g(u, z) + \sum_{i=1}^{n} \sum_{j=1}^{MN} \sqrt{(\lambda_i S_i u)_j^2 + \varepsilon^2}, \tag{7.1}$$

which can be interpreted as a smooth approximation of a sparsity inducing ℓ_1 penalization of the shearlet coefficients. Observe that the formulation recovers the standard ℓ_1 penalization for $\varepsilon \to 0$. The function $g(u, z)$ can be any convex function promoting data fidelity for a given input z. We use the standard choice $g(u, z) = \frac{\mu}{2} \|u - z\|^2$ for denoising Gaussian noise.

Transforming (7.1) into a saddle-point problem yields

$$\min_u \max_p g(u, z) + \sum_{i=1}^{n} \Big(\langle \lambda_i S_i u, p_i \rangle - f^*(p_i) \Big), \tag{7.2}$$

with $f^*(p_i) = -\varepsilon \sum_{j=1}^{MN} \sqrt{1 - p_{ij}^2} + \delta_{|\cdot|\leq 1}(p_{ij})$, which can be solved by a standard primal-dual algorithm [8] as long as the proximal maps with respect to the nonlinear functions are easy to compute. While the proximal map of g is easy by construction, the proximal map of f^*, which is the dual of the smoothed ℓ_1 norm, is more involved. It is solved with a projected Newton method which consists of finding the correct root of the pointwise quartic equation

$$p_{ij}^4 - 2\widetilde{p}_{ij} p_{ij}^3 + (\widetilde{p}_{ij}^2 - 1 + \sigma^2\varepsilon^2) p_{ij}^2 + 2\widetilde{p}_{ij} p_{ij} - \widetilde{p}_{ij}^2 = 0,$$

and re-projecting onto to the constraint $\delta_{|\cdot|\leq 1}$. Although a closed form solution for the quartic equations is clearly available, we use Newton's algorithm which (when properly initialized) recovers the correct root within a few (5-10) iterations.

7.3. Learning the Shearlet Parameters. The regularization weights λ_i, the scaling function h_1, and the 2D fan filter P can be optimized with the piggyback algorithm. These parameters $\theta = \{\lambda_i, h_1, P\}$ are learned with a set of input images $\{z_1, ..., z_L\}$ and corresponding targets $\{t_1, ..., t_L\}$ by minimizing

$$\min_{\theta=\{\lambda_i,h_1,P\}} \mathcal{L}(\theta) + \mathcal{R}(\theta) \coloneqq \frac{1}{MNL} \sum_{l=1}^{L} \ell(u_l^*(\theta), t_l) + \mathcal{R}(\theta). \tag{7.3}$$

A quadratic loss function is chosen and the regularization on the learned parameters $\mathcal{R}(\theta)$ ensures that $\sum_n h_1(n) = 1$, $\lambda_i \in \mathbb{R}^+$, and $\sum_i |P_i| = 1$. The solution to the saddle-point problem defined in (7.2) is $u_l^*(\theta)$, which amounts to the lower level solution in the bilevel optimization problem. For the piggyback primal-dual algorithm, we compute the saddle-point u^K, p^K and its adjoint states U^K, P^K using Algorithm 1, which are the solutions to the biquadratic saddle-point problem as stated in (2.3).

Algorithm 1: Piggyback primal-dual algorithm for solving (7.2) and its adjoint.

- Initialization: $u^0, U^0 \in \mathcal{X}$, $p^0, P^0 \in \mathcal{Y}$.
- Step sizes: Choose the step sizes τ, σ such that $\sigma\tau L^2 \leq 1$.
- Iterations: For each $k = 0, \ldots, K-1$ let

$$\begin{cases} \widetilde{u}^{k+1} &= u^k - \tau S^T \lambda p^k, \quad \widetilde{U}^{k+1} = U^k - \tau(S^T \lambda p^k + \nabla_u \ell(u^k, t)) \\ u^{k+1} &= \operatorname{prox}_{\tau g}(\widetilde{u}^{k+1}), \quad U^{k+1} = \nabla \operatorname{prox}_{\tau g}(\widetilde{u}^{k+1}) \cdot \widetilde{U}^{k+1} \\ \bar{u}^{k+1} &= 2u^{k+1} - u^k, \quad \bar{U}^{k+1} = 2U^{k+1} - U^k \\ \widetilde{p}^{k+1} &= p^k + \sigma\lambda S \bar{u}^{k+1}, \quad \widetilde{P}^{k+1} = P^k + \sigma\lambda S \bar{U}^{k+1} \\ p^{k+1} &= \operatorname{prox}_{\sigma f^*}(\widetilde{p}^{k+1}), \quad p^{k+1} = \nabla \operatorname{prox}_{\sigma f^*}(\widetilde{p}^{k+1}) \cdot \widetilde{P}^{k+1}. \end{cases} \tag{7.4}$$

- Output: Approximate saddle-point (u^K, p^K) and corresponding adjoint state (U^K, P^K).

After K iterations of the piggyback primal-dual algorithm (Algorithm 1), the derivatives of $\ell(u^*, t)$ with respect to the parameters θ are given by

$$\langle \nabla_\theta \ell(u^*, t), \theta \rangle = \langle u^K, S^T P^K \rangle + \langle P^K, S u^K \rangle,$$

which are computed with automatic differentiation provided by PyTorch. The parameters are then updated using an accelerated proximal gradient descent scheme described in Algorithm 2, where the projection for each parameter depends on the specified constraint.

Algorithm 2: Accelerated proximal gradient method for solving (7.3)

- Initialization: $\theta^0 = \{\lambda_i^0, h_1^0, P^0\}$.
- Step sizes: Choose $\eta^s > 0$, $\beta^s \in [0, 1)$.
- Iterations: For $s = 0, \dots, S-1$ let

$$\begin{cases} \bar{\theta}^s = \theta^s + \beta^s(\theta^s - \theta^{s-1}) \\ \theta^{s+1} = \operatorname{proj}(\bar{\theta}^s - \eta^s \nabla \mathcal{L}(\bar{\theta}^s)). \end{cases} \tag{7.5}$$

- Output: Learned parameters $\theta^S = \{\lambda_i^S, h_1^S, P^S\}$.

7.4. Results. In this section we show results for learned parameters $\theta = \{\lambda_i, h_1, P\}$ of a shearlet system used as a regularizer in natural image denoising, a convex imaging application described in Section 7.2. We generate training and test datasets each comprised of 32 images of size 256×256, which are randomly sampled from the BSDS500 dataset [1]. The images are corrupted with i.i.d. zero-mean Gaussian noise with a standard deviation of $\sigma = 0.05$.

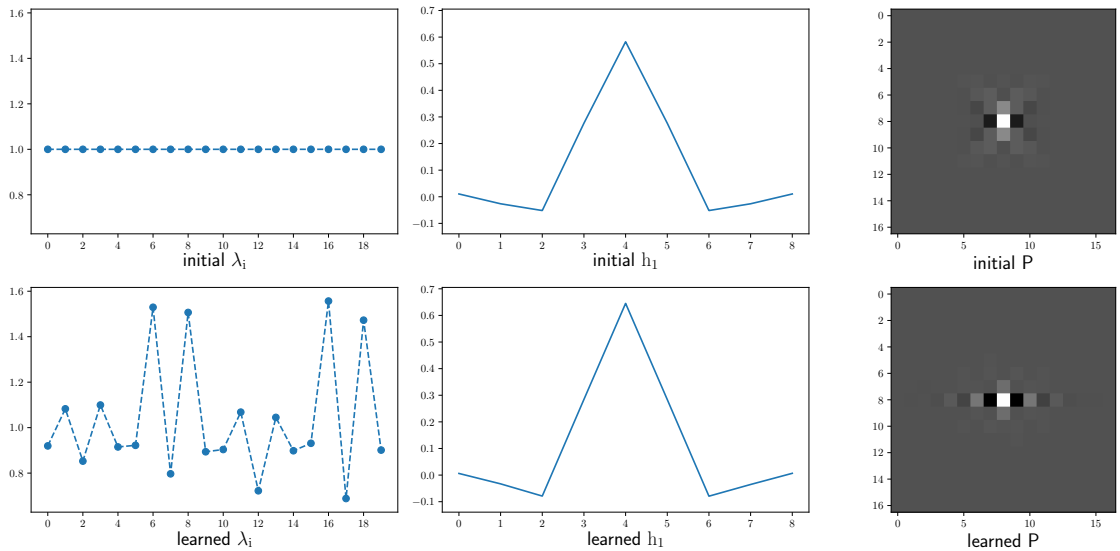
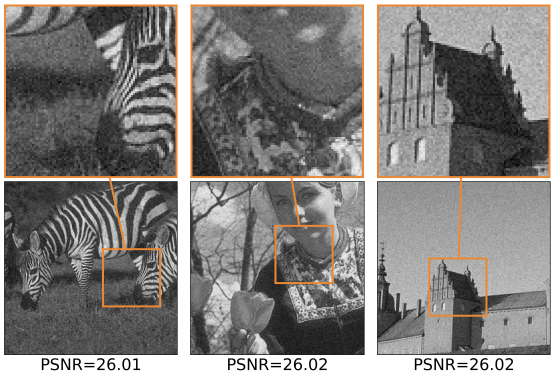


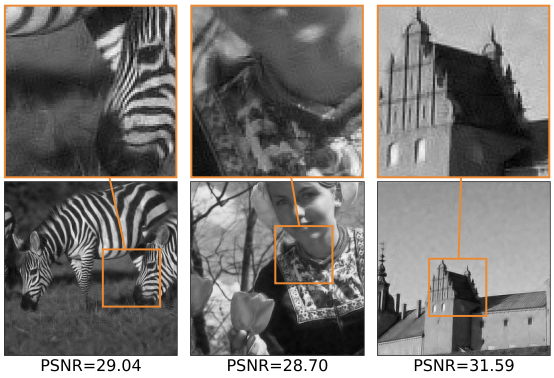
Figure 7.1. *Initial (top) and learned (bottom) parameters of a shearlet system with 2 scales. The learnable parameters consist of the regularization parameter λ_i that allows to individually balance different scales and shear levels and the filters h_1 and P that govern the construction of shearlets $\psi_{j,k}$.*

For the piggyback algorithm (Algorithm 1) $K = 50$ iterations are computed and a warm-starting initialization scheme is used for both u and p and their adjoint variables to get more accurate results. The primal and dual step sizes were set to their theoretically optimal values, based on the settings of μ, ε. For the learning setting $S = 1000$ gradient update steps of Algorithm 2 are performed to ensure a sufficient number of iterations for the loss function to stabilize. The inertial parameter is set to $\beta^s = 0.7$ and the step size is set to $\eta^s = 10^{-2}$.

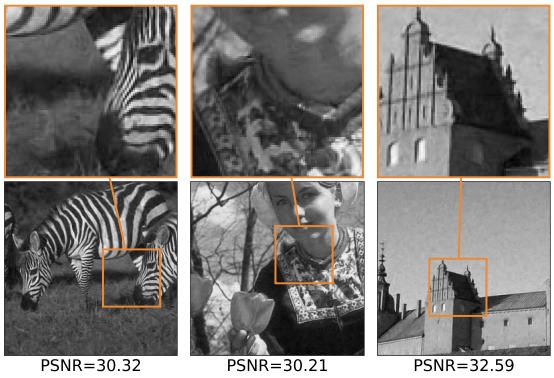
(a) ground truth sample test images t_l.



(b) noisy sample images z_l.



(c) sample denoised images u_l with initial shearlet system parameters and global $\lambda = 1.33$.



(d) sample denoised images u_l with learned shearlet system parameters shown in Figure 7.1.

Figure 7.2. *Sample denoised images from the test set with accompanying PSNR values comparing initial and learned shearlet parameters. The corrupted test images were denoised using shearlet regularization based on initial shearlet parameters with a global $\lambda = 1.33$ (mean PSNR 30.09 dB) and learned shearlet parameters with the piggyback primal-dual algorithm (mean PSNR 31.2 dB).*

The initial and learned parameters for a shearlet system with 2 scales used to solve the denoising problem in Equation 7.2 with $\varepsilon = 10^{-4}$ are shown in Figure 7.1. The regularization parameter λ_i allows to individually balance the shearlets $\psi_{j,k}$ ordered by scale $j = \{0, 1\}$ and shearing $k = \{-2, ..., 2\}$, i.e. $\{\psi_{0,-2}, ..., \psi_{0,2}, \psi_{1,-2}, ..., \psi_{1,2}\}$ for the first frequency cone and analogously for the second. The learned λ_i are similar among both frequency cones, which is manifested in the repeating pattern and shearlets at the higher scale are given more weight to emphasize high frequency details in images. The optimized low-pass filter h_1 exhibits only minor numerical changes where the overall filter structure remains the same. The learned 2D directional filter P shows noticeable deviations from its initialization, which significantly impacts the resulting shearlets. Further enhancing the directional selectivity benefits the task of shearlet regularized image denoising while preserving the wedge shaped frequency support of the generated shearlets.

Samples of denoised test images are shown in Figure 7.2, where input images were denoised

using shearlet regularization with $\varepsilon = 10^{-4}$ for two different parameter settings. First, initial parameters for h_1 and P and a global, hand-tuned $\lambda = 1.33$ were used while the second setting is based on the learned parameters shown in Figure 7.1. Corresponding quantitative peak signal-to-noise ratio (PSNR) values are shown for each sample image which emphasize the qualitative visual improvement. Using shearlet regularization with initial parameters and a global λ delivers results with a mean PSNR of 30.09 dB compared to 26.02 dB in the noisy input images, provided that a suitable regularization parameter λ is chosen. However, optimizing shearlet parameters and individually weighting the shearlets with λ_i increases the mean PSNR to 31.2 dB which is supported in the enhanced visual quality of the denoised images. This can be further observed in the enlarged sections in Figure 7.2, where higher frequent structures in (c) associated with remaining noise or minor artifacts from shearlet regularization are removed in (d) when using the learned shearlet parameters.

Furthermore, different settings for ε governing the smoothness of the regularizing function were compared. As ε is decreased, the regularizing function approximates an ℓ_1 penalization, while still fulfilling the assumption that g^* and f in $(\mathcal{S})$ are $C^{2,\alpha}$ functions. 500 iterations of a primal-dual algorithm are performed to denoise the test dataset by solving Equation 7.2 using the learned shearlet parameters for the corresponding cases of ε. Quantitative results for mean squared error (MSE) and mean PSNR for both the initial and optimized shearlet transforms are summarized in Table 7.1, showing improved results with decreasing ε. For the sake of completeness, the case $\varepsilon = 0$ penalizing the shearlet coefficients with the ℓ_1 norm is included, indicating that a penalizing function with $\varepsilon = 10^{-4}$ is already a very good approximation in terms of quantitative error scores. Moreover, it shows robustness of the piggyback algorithm which works even in the case of less regular functions.

Table 7.1

Comparison between the performance of the initial shearlet transform (hand tuned) and the optimized shearlet transform for various setting of the smoothing parameter ε. Note that the learned transform clearly outperforms the initial shearlet transform and that smaller settings of ε lead to better results.

	$\varepsilon = 10^{-1}$		$\varepsilon = 10^{-2}$		$\varepsilon = 10^{-3}$		$\varepsilon = 10^{-4}$		$\varepsilon = 0$	
	MSE	PSNR	MSE	PSNR	MSE	PSNR	MSE	PSNR	MSE	PSNR
Initial	0.002143	26.7	0.001344	28.77	0.001076	29.93	0.001051	30.09	0.00105	30.1
Optimized	0.001173	29.46	0.000985	30.27	0.000846	30.99	0.000814	31.2	0.000813	31.2

REFERENCES

[1] Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(5):898–916, May 2011.

[2] Jean François Aujol and Charles Dossal. Stability of over-relaxations for the forward-backward algorithm, application to FISTA. *SIAM J. Optim.*, 25(4):2408–2433, 2015.

[3] Jean-François Aujol, Charles Dossal, Gersende Fort, and Éric Moulines. Rates of Convergence of Perturbed FISTA-based algorithms. working paper or preprint, July 2019.

[4] Heinz H. Bauschke and Patrick L. Combettes. *Convex analysis and monotone operator theory in Hilbert spaces*. CMS Books in Mathematics/Ouvrages de Mathématiques de la SMC. Springer, New York, 2011. With a foreword by Hédy Attouch.

[5] Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Sci.*, 2(1):183–202, 2009.

[6] Quentin Bertrand, Quentin Klopfenstein, Mathieu Blondel, Samuel Vaiter, Alexandre Gramfort, and Joseph Salmon. Implicit differentiation of lasso-type models for hyperparameter optimization. In *International Conference on Machine Learning*, pages 810–821. PMLR, 2020.

[7] Luca Calatroni, Chung Cao, Juan Carlos De los Reyes, Carola-Bibiane Schönlieb, and Tuomo Valkonen. Bilevel approaches for learning of variational imaging models. In *Variational methods*, volume 18 of *Radon Ser. Comput. Appl. Math.*, pages 252–290. De Gruyter, Berlin, 2017.

[8] Antonin Chambolle and Thomas Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *J. Math. Imaging Vision*, 40(1):120–145, 2011.

[9] Antonin Chambolle and Thomas Pock. An introduction to continuous optimization for imaging. *Acta Numer.*, 25:161–319, 2016.

[10] Antonin Chambolle and Thomas Pock. On the ergodic convergence rates of a first-order primal-dual algorithm. *Math. Program.*, 159(1-2, Ser. A):253–287, 2016.

[11] Antonin Chambolle and Thomas Pock. Learning consistent discretizations of the total variation. *SIAM J. Imaging Sci.*, 14(2):778–813, 2021.

[12] Y Chen, R Ranftl, and T Pock. Insights into analysis operator learning: A view from higher-order filter-based mrf model. *IEEE Trans. Image Process*, 23(3):1060–1072, 2014.

[13] Bruce Christianson. Reverse accumulation and implicit functions. *Optim. Methods Softw.*, 9(4):307–322, 1998.

[14] P. L. Combettes and V. R. Wajs. Signal recovery by proximal forward-backward splitting. *Multiscale Model. Simul.*, 4(4):1168–1200, 2005.

[15] Michael G. Crandall, Hitoshi Ishii, and Pierre-Louis Lions. User's guide to viscosity solutions of second order partial differential equations. *Bull. Amer. Math. Soc. (N.S.)*, 27(1):1–67, 1992.

[16] Juan Carlos De los Reyes and Carola-Bibiane Schönlieb. Image denoising: learning the noise model via nonsmooth PDE-constrained optimization. *Inverse Probl. Imaging*, 7(4):1183–1214, 2013.

[17] Charles-Alban Deledalle, Samuel Vaiter, Jalal Fadili, and Gabriel Peyré. Stein Unbiased GrAdient estimator of the Risk (SUGAR) for multiple parameter selection. *SIAM J. Imaging Sci.*, 7(4):2448–2487, 2014.

[18] Stephan Dempe. Bilevel optimization: Theory, algorithms, applications and a bibliography. In *Bilevel Optimization*, pages 581–672. Springer, 2020.

[19] Justin Domke. Generic methods for optimization-based modeling. In *Artificial Intelligence and Statistics*, pages 318–326. PMLR, 2012.

[20] Yiqiu Dong, Michael Hintermüller, and M. Monserrat Rincon-Camacho. Automated regularization parameter selection in multi-scale total variation models for image restoration. *J. Math. Imaging Vision*, 40(1):82–104, 2011.

[21] Alexander Effland, Erich Kobler, Karl Kunisch, and Thomas Pock. Variational networks: An optimal control approach to early stopping variational methods for image restoration. *Journal of mathematical imaging and vision*, pages 1–21, 2020.

[22] Lawrence C. Evans and Ronald F. Gariepy. *Measure theory and fine properties of functions*. Textbooks in Mathematics. CRC Press, Boca Raton, FL, revised from 1992 edition, 2015.

[23] Luca Franceschi, Michele Donini, Paolo Frasconi, and Massimiliano Pontil. Forward and reverse gradient-based hyperparameter optimization. In *International Conference on Machine Learning*, pages 1165–1173. PMLR, 2017.

[24] Karol Gregor and Yann LeCun. Learning fast approximations of sparse coding. In *Proceedings of the 27th international conference on international conference on machine learning*, pages 399–406, 2010.

[25] Andreas Griewank and Christèle Faure. Piggyback differentiation and optimization. In *Large-scale PDE-constrained optimization (Santa Fe, NM, 2001)*, volume 30 of *Lect. Notes Comput. Sci. Eng.*, pages 148–164. Springer, Berlin, 2003.

[26] Andreas Griewank and Andrea Walther. *Evaluating derivatives: principles and techniques of algorithmic differentiation*. SIAM, 2008.

[27] Michael Hintermüller and Kostas Papafitsoros. Generating structured nonsmooth priors and associated primal-dual methods. In *Processing, analyzing and learning of images, shapes, and forms. Part 2*, volume 20 of *Handb. Numer. Anal.*, pages 437–502. Elsevier/North-Holland, Amsterdam, 2019.

[28] Michael Hintermüller and Carlos N. Rautenberg. Optimal selection of the regularization function in a weighted total variation model. Part I: Modelling and theory. *J. Math. Imaging Vision*, 59(3):498–514,

2017.

[29] Michael Hintermüller, Carlos N. Rautenberg, Tao Wu, and Andreas Langer. Optimal selection of the regularization function in a weighted total variation model. Part II: Algorithm, its analysis and numerical tests. *J. Math. Imaging Vision*, 59(3):515–533, 2017.

[30] Ralph Howard. Alexandrov's theorem on the second derivatives of convex functions *via* Rademacher's theorem on the first derivatives of Lipschitz functions. Lecture Notes, Dept. of Math., Univ. South Carolina, 1998.

[31] Karl Kunisch and Thomas Pock. A bilevel optimization approach for parameter learning in variational models. *SIAM J. Imaging Sci.*, 6(2):938–983, 2013.

[32] Gitta Kutyniok and Demetrio Labate. *Shearlets: Multiscale analysis for multivariate data*. Springer Science & Business Media, 2012.

[33] Gitta Kutyniok, Wang-Q Lim, and Rafael Reisenhofer. Shearlab 3d: Faithful digital shearlet transforms based on compactly supported shearlets. *ACM Trans. Math. Softw.*, 42(1), January 2016.

[34] Wang-Q Lim. Nonseparable shearlet transform. *IEEE Transactions on Image Processing*, 22(5):2056–2065, 2013.

[35] Jonathan Lorraine, Paul Vicol, and David Duvenaud. Optimizing millions of hyperparameters by implicit differentiation. In *International Conference on Artificial Intelligence and Statistics*, pages 1540–1552. PMLR, 2020.

[36] Sheheryar Mehmood and Peter Ochs. Automatic differentiation of some first-order methods in parametric optimization. In *International Conference on Artificial Intelligence and Statistics*, pages 1584–1594. PMLR, 2020.

[37] Fulbert Mignot. Contrôle dans les inéquations variationelles elliptiques. *J. Functional Analysis*, 22(2):130–185, 1976.

[38] Yurii Nesterov. *Introductory lectures on convex optimization*, volume 87 of *Applied Optimization*. Kluwer Academic Publishers, Boston, MA, 2004. A basic course.

[39] P. A. Newman, G. J.-W. Hou, H. E. Jones, A. C. Taylor III, and V. M. Korivi. Observations on computational methodologies for use in large-scale, gradient-based, multidisciplinary design incorporating advanced CFD code. In *4th Symposium on Multidisciplinary Analysis and Optimization*, page 4753, 1992.

[40] Peter Ochs, René Ranftl, Thomas Brox, and Thomas Pock. Techniques for gradient-based bilevel optimization with non-smooth lower level problems. *Journal of Mathematical Imaging and Vision*, 56(2):175–194, 2016.

[41] Fabian Pedregosa. Hyperparameter optimization with approximate gradient. In *International conference on machine learning*, pages 737–746. PMLR, 2016.

[42] Gabriel Peyré and Jalal M. Fadili. Learning Analysis Sparsity Priors. In *Sampta'11*, page 4 pp., Singapour, Singapore, 2011.

[43] Aravind Rajeswaran, Chelsea Finn, Sham Kakade, and Sergey Levine. Meta-learning with implicit gradients. *Advances in neural information processing systems*, 2019.

[44] Julian Rasch and Antonin Chambolle. Inexact first-order primal-dual algorithms. *Comput. Optim. Appl.*, 76(2):381–430, 2020.

[45] Kegan GG Samuel and Marshall F Tappen. Learning optimized map estimates in continuously-valued mrf models. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 477–484. IEEE, 2009.

[46] Jeremias Sulam, Vardan Papyan, Yaniv Romano, and Michael Elad. Multilayer convolutional sparse modeling: Pursuit and dictionary learning. *IEEE Transactions on Signal Processing*, 66(15):4090–4104, 2018.

[47] Ala Taftaf, Valérie Pascual, and Laurent Hascoët. Adjoints of fixed-point iterations. In *11th World Congress on Computational Mechanics (WCCM XI)*, 2014.

[48] Marshall F. Tappen. Utilizing variational optimization to learn markov random fields. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2007.

[49] Cédric Vonesch, Sathish Ramani, and Michael Unser. Recursive risk estimation for non-linear image deconvolution with a wavelet-domain sparsity constraint. In *2008 15th IEEE International Conference on Image Processing*, pages 665–668. IEEE, 2008.