A Generic Proximal Algorithm for Convex Optimization — Application to Total Variation Minimization

Laurent Condat

***Abstract*—We propose new optimization algorithms to minimize a sum of convex functions, which may be smooth or not and composed or not with linear operators. This generic formulation encompasses various forms of regularized inverse problems in imaging. The proposed algorithms proceed by splitting: the gradient or proximal operators of the functions are called individually, without inner loop or linear system to solve at each iteration. The algorithms are easy to implement and have proven convergence to an exact solution. The classical Douglas–Rachford and forward–backward splitting methods, as well as the recent and efficient algorithm of Chambolle–Pock, are recovered as particular cases. The application to inverse imaging problems regularized by the total variation is detailed.**

***Index Terms*—Convex nonsmooth optimization, proximal splitting algorithm, regularized inverse problem, total variation**

I. Introduction and Problem Formulation

Numerous problems in signal processing and imaging [1]–[4], statistical learning and data mining [5], [6], or computer vision [7], [8], can be formulated as optimization problems, which consist in minimizing a sum of convex functions, not necessarily differentiable, possibly composed with linear operators. Each function is typically either a data-dependent loss function, a.k.a. data fidelity term, or a regularization term enforcing some properties on the solution. Using nonsmooth penalties, e.g. based on the ℓ_1 norm, has proved beneficial in many fields, to constrain the solution to be regular and parsimonious in some sense [5], [9]. A typical example of a nonsmooth function composed with a linear operator is the total variation seminorm, which is an efficient regularizer for many ill-posed inverse problems in imaging [10].

To formulate the optimization problem of interest (2), let us recall some classical definitions of convex optimization; we refer the reader to a textbook of convex analysis, such as [11], for more in-depth study. A function g defined on a real Hilbert space $\mathcal{X}$ with values in $\mathbb{R} \cup \{+\infty\}$ is said *proper* if its domain $\mathrm{dom}(g) = \{x \in \mathcal{X} \ ; \ g(x) < +\infty\}$ is nonempty. g is said *convex* if $g(ax + (1-a)x') \leq ag(x) + (1-a)g(x')$ for every $x, x' \in \mathcal{X}$ and $a \in [0,1]$. g is said *lower semicontinuous* at $x \in \mathcal{X}$ if, for every $v \in \left]-\infty, g(x)\right[$, we can find a neighborhood Ω of x such that $f(\Omega) \subset \left]v, +\infty\right]$. The set of convex, proper, lower semicontinuous functions from $\mathcal{X}$ to $\mathbb{R} \cup \{+\infty\}$ is denoted by $\Gamma_0(\mathcal{X})$. Let us define the Moreau *proximity operator* of $g \in \Gamma_0(\mathcal{X})$ by

$$\mathrm{prox}_g : \mathcal{X} \rightarrow \mathcal{X}, x \mapsto \underset{x' \in \mathcal{X}}{\arg\min} \ \tfrac{1}{2}\|x - x'\|^2 + g(x'). \qquad (1)$$

There exist simple explicit expressions for the proximity operators of a large class of functions [12], [13]. An iterative algorithm which aims at minimizing a sum of functions by successive evaluations of their gradients or proximity operators is called a *proximal algorithm.*

In this paper, we aim at solving the following generic optimization problem, formulated in real Hilbert spaces $\mathcal{X}$ and $\{\mathcal{U}_m\}_{m=1}^M$, for some $M \in \mathbb{N}$:

$$\boxed{\text{Find} \ \ \hat{x} \in \underset{x \in \mathcal{X}}{\arg\min} \ f(x) + g(x) + \sum_{m=1}^{M} h_m(L_m x),} \qquad (2)$$

where:

- $f, g \in \Gamma_0(\mathcal{X})$, $h_m \in \Gamma_0(\mathcal{U}_m)$.
- The operators $L_m : \mathcal{X} \rightarrow \mathcal{U}_m$ are linear and bounded.
- f is differentiable on $\mathcal{X}$ and its gradient ∇f is β-Lipschitz continuous, for some real constant $\beta > 0$, i.e.,

$$\|\nabla f(x) - \nabla f(x')\| \leq \beta \|x - x'\|, \quad \forall x, x' \in \mathcal{X}. \qquad (3)$$

- The set of minimizers is supposed nonempty.
- The following qualification constraint is satisfied: $(0, \ldots, 0) \in \mathrm{sri}\big\{(L_m x - u_m)_{1 \leq m \leq M} \mid x \in \mathrm{dom}(g)$ and $u_m \in \mathrm{dom}(h_m), \ \forall m = 1, \ldots, M\big\}$, where sri denotes the strong relative interior.

Note that we allow some of the functions in (2) to be zero and a linear operator L_m can be the identity Id. Also, we recall that, given a nonempty closed convex set $\Omega \subset \mathcal{X}$, we can define the indicator function $\imath_\Omega \in \Gamma_0(\mathcal{X}) : x \in \mathcal{X} \mapsto \{0 \text{ if } x \in \Omega, \ +\infty \text{ else}\}$. Such functions are convenient to enforce hard constraints on the solution: the problem of minimizing $f \in \Gamma_0(\mathcal{X})$ over Ω can be written as the minimization of $f + \imath_\Omega$ over the whole space $\mathcal{X}$. Since the proximity operator of an indicator function is simply the projection onto the set, proximal algorithms can be viewed as generalizations of algorithms to find an element in the intersection of convex sets by successive projections (POCS) [11], [13].

The main difficulty in solving (2) stems form the fact that the spaces $\mathcal{X}$ and $\mathcal{U}_m$ are typically of high dimension, hence the denomination of *large-scale optimization*. For instance, if the sought-after solution $\hat{x}$ is an image, the dimension N of $\mathcal{X}$ is the number of pixels in the image. For $N = 10^6$ and

L. Condat is with GIPSA-lab, University of Grenoble–Alpes, in Grenoble, France. Contact: see http://www.gipsa-lab.fr/∼laurent.condat/

```
Proposed Algorithm 1
Choose the parameters $\tau > 0$, $\sigma > 0$, $\rho > 0$ and the
initial estimates $x^{(0)} \in \mathcal{X}$, $u_1^{(0)} \in \mathcal{U}_1, \dots, u_M^{(0)} \in \mathcal{U}_M$.
Then iterate, for $i = 0, 1, \dots$
    $\tilde{x}^{(i+1)} := \mathrm{prox}_{\tau g}\Big(x^{(i)} - \tau \nabla f(x^{(i)}) - \tau \sum_{m=1}^{M} L_m^* u_m^{(i)}\Big)$,
    $x^{(i+1)} := \rho\, \tilde{x}^{(i+1)} + (1-\rho) x^{(i)}$,
    For $m = 1, \dots, M$,
        $\tilde{u}_m^{(i+1)} := \mathrm{prox}_{\sigma h_m^*}\big(u_m^{(i)} + \sigma L_m (2\tilde{x}^{(i+1)} - x^{(i)})\big)$,
        $u_m^{(i+1)} := \rho\, \tilde{u}_m^{(i+1)} + (1-\rho) u_m^{(i)}$.
```

```
Proposed Algorithm 2
Choose the parameters $\tau > 0$, $\sigma > 0$, $\rho > 0$ and the
initial estimates $x^{(0)} \in \mathcal{X}$, $u_1^{(0)} \in \mathcal{U}_1, \dots, u_M^{(0)} \in \mathcal{U}_M$.
Then iterate, for $i = 0, 1, \dots$
    For $m = 1, \dots, M$,
        $\tilde{u}_m^{(i+1)} := \mathrm{prox}_{\sigma h_m^*}\big(u_m^{(i)} + \sigma L_m x^{(i)}\big)$,
        $u_m^{(i+1)} := \rho\, \tilde{u}_m^{(i+1)} + (1-\rho) u_m^{(i)}$,
    $\tilde{x}^{(i+1)} := \mathrm{prox}_{\tau g}\Big(x^{(i)} - \tau \nabla f(x^{(i)}) - \tau \sum_{m=1}^{M} L_m^* (2\tilde{u}_m^{(i+1)} - u_m^{(i)})\Big)$,
    $x^{(i+1)} := \rho\, \tilde{x}^{(i+1)} + (1-\rho) x^{(i)}$.
```

Fig. 1. The two proposed algorithms to solve the convex optimization problem (2).

beyond, it is not possible to manipulate, at every iteration of an algorithm, matrices of size $N \times N$, like the Hessian of a function. So, proximal algorithms, which only exploit first-order information of the functions, are often the only viable way to solve (2). In this paper, we place ourselves in the case where it is not possible to apply, at every iteration of an algorithm, an inverse operator like $(\mathrm{Id} + L_m^* L_m)^{-1}$, which amounts to solving a linear system, or operators like $\mathrm{prox}_{h_m \circ L_m}$ or prox_{f+g}. Also, we exclude nested strategies, which consist in solving iteratively an optimization problem at every iteration, as this raises theoretical and practical convergence issues [14]. Thus, we propose two new proximal algorithms to solve the problem (2) by *full splitting*; that is, at every iteration, the only operations involved are evaluations of ∇f, prox_g, prox_{h_m}, L_m, or the adjoint operators L_m^*. Thus, it is required that these evaluations are "simple"; that is, that they can be performed in time like $O(N)$ or $O(N \log(N))$, where N is the dimension of the ambiant space $\mathcal{X}$ or $\mathcal{U}_m$.

The paper aims at making known the optimization algorithms proposed by the author in [15], to the signal and image processing community. Thus, the two algorithms are presented in Sect. II in a slightly simplified form, along with the corresponding convergence results, but the mathematical proofs are omitted. The relationship of the algorithm with other methods of the literature is discussed. In Sect. III, we detail as a proof of concept the application to inverse imaging problems regularized by the total variation.

II. Proposed Algorithms and Convergence Analysis

The two proposed algorithms are in Fig. 1. Note that we make use of $h_m^* \in \Gamma_0(\mathcal{U}_m)$, the Fenchel–Rockafellar conjugate of h_m, which satisfies the useful Moreau identity: for every $u \in \mathcal{U}_m$ and real $\sigma > 0$, $\mathrm{prox}_{\sigma h_m^*}(u) = u - \sigma\, \mathrm{prox}_{h_m/\sigma}(u/\sigma)$. Some classical expressions of the gradient and proximity operator are provided in Fig. 2. The two algorithms behave very similarly in practice; which one is the most appropriate for a particular problem depends on the way they are implemented, since the different place of the extrapolation step $2\tilde{\circ}^{(i+1)} - \circ^{(i)}$ can lead to different memory storage requirements.

The convergence results, whose proofs can be found in [15], are the following:

$g(x)$	$\mathrm{prox}_{\tau g}(x)$	$\nabla g(x)$
0	x	0
$\imath_\Omega(x)$	$P_\Omega(x)$	
$\imath_{(\mathbb{R}+)^N}(x)$	$\big(\max\{x_n, 0\}\big)_{n=1}^N$	
$\lVert x\rVert_1 = \sum_{n=1}^N x_n$	$\big(\mathrm{sgn}(x_n) \max\{\lvert x_n\rvert - \tau, 0\}\big)_{n=1}^N$	
$\imath_{\{x'\ ;\ Ax'=y\}}(x)$	$x + A^\dagger(y - Ax)$	
$\frac{1}{2}\lVert Ax - y\rVert^2$	$(\mathrm{Id} + \tau A^* A)^{-1}(x + \tau A^* y)$	$A^*(Ax - y)$
$\langle Ax, y\rangle = \langle x, A^* y\rangle$	$x - \tau A^* y$	$A^* y$
$\frac{1}{2}\langle Ax, x\rangle$	$(\mathrm{Id} + \tau A)^{-1} x$	Ax

Fig. 2. Expressions of the gradient and proximity operator of a convex function $g : \mathcal{X} \to \mathbb{R} \cup \{+\infty\}$, in some classical cases. In the tables, x is an arbitrary element of some real Hilbert space $\mathcal{X}$, τ a positive real, y an arbitrary element of some real Hilbert space $\mathcal{Y}$, $A : \mathcal{X} \to \mathcal{Y}$ a linear operator, $A^\dagger$ its Moore–Penrose pseudoinverse, Ω a subset of $\mathcal{X}$.

Theorem 1. *Suppose that the parameters in Algorithm 1 or Algorithm 2 satisfy:*

(i) $\tau\Big(\frac{\beta}{2} + \sigma\Big\lVert \sum_{m=1}^{M} L_m^* L_m \Big\rVert\Big) < 1$, *where the Lipschitz constant β is defined in* (3) *and $\lVert\cdot\rVert$ is the operator norm.*

(ii) $\rho \in \,]0, 1]$.

Then both sequences $(\tilde{x}^{(i)})_{i\in\mathbb{N}}$ and $(x^{(i)})_{i\in\mathbb{N}}$ generated by Algorithm 1 or Algorithm 2 converge (weakly if $\mathcal{X}$ has infinite dimension) to an element $\hat{x} \in \mathcal{X}$ solution of the problem (2).

Theorem 2. *Suppose that $f = 0$, that the spaces $\mathcal{X}$ and $\mathcal{U}_m$ have finite dimension, and that the parameters in Algorithm 1 or Algorithm 2 satisfy:*

(i) $\tau\sigma\Big\lVert \sum_{m=1}^{M} L_m^* L_m \Big\rVert \leq 1$.

(ii) $\rho \in \,]0, 2[$.

Then both sequences $(\tilde{x}^{(i)})_{i\in\mathbb{N}}$ and $(x^{(i)})_{i\in\mathbb{N}}$ generated by Algorithm 1 or Algorithm 2 converge to an element $\hat{x} \in \mathcal{X}$ solution of the problem (2).

We refer to the mathematical article [15] for a more detailed study of the convergence conditions, including error terms and variable relaxation parameters. Moreover, the proposed approach is primal-dual, in the sense that $(u_1^{(i)}, \dots, u_M^{(i)})$ converges to an element $(\hat{u}_1, \dots, \hat{u}_M) \in \mathcal{U}_1 \times \cdots \times \mathcal{U}_M$

solution to the dual problem of (2). Qualification constraints other than the ones given in Sect. 1 can be found [16]. Also, the problem (2) and the algorithms can be generalized to include infimal convolutions, or to solve monotone inclusions instead of optimization problems [17]. Finally, the metric can be changed, with potential acceleration [18], [19].

A. Particular Cases and Related Work

The most classical splitting methods to minimize the sum of two functions are the *forward-backward* and *Douglas–Rachford* algorithms, see [13] and references therein. In our notations, they allow to minimize $f(x)+g(x)$ and $g(x)+h(x)$, respectively. Until recently, there was no convenient way of solving a problem like (2) with nontrivial linear operators L_m. A step forward regarding this issue was made in [20]: the *Chambolle–Pock* algorithm allows to minimize $g(x)+h(Lx)$. Several other algorithms have been proposed recently for particular instances of the problem (2) [4], [16], [21]–[25], but only the algorithm in [16] is a rival to ours and allows to solve the problem (2) in whole generality.

In the case $f=0$, the proposed algorithms revert to the ones of Chambolle–Pock, with additional relaxation. Indeed, according to Theorem 2, we allow a value ρ close to 2, instead of $\rho=1$ in [20], which can significantly speed up the convergence. Moreover, the convergence is guaranteed with the choice $\sigma\tau\big\|\sum_m L_m^* L_m\big\|=1$, which we recommend in practice, whereas the condition $\sigma\tau\big\|\sum_{i=1}^m L_i^* L_i\big\|<1$ was given in [20]. This new condition is important, since it allows to recover the Douglas–Rachford algorithm as a particular case of ours, when $f=0$, $M=1$, $L_1=\mathrm{Id}$, by setting $\sigma=1/\tau$ to let τ appear as the only parameter.

If $M=0$ and one simply wants to minimize $f(x)+g(x)$, the proposed algorithms revert to the forward-backward algorithm.

We remark that there are often several ways to assign the functions of a given problem to the terms f, g, $h_m\circ L_m$ in (2). In particular, a function like $\frac{1}{2}\|A\cdot-y\|^2$ in (4) can be assigned either to f or to a term $h_m\circ L_m$ with $L_m=A$. These two formulations yield different algorithms. Although it is hard to make general statements, assigning a function to f whenever one can is probably better for the convergence speed, because of the serial way the variables are updated: the step of gradient descent with respect to f updates and improves the variable x, and this updated version is used to update the dual variables u_m. By contrast, the variables u_m are updated independently and in parallel, with respect to the antagonist functions h_m, before being essentially averaged to form the new estimate of x. So, except if the algorithm is run on a parallel architecture, the higher is M, the slower is the convergence. For the same reason, one should make use of the function g instead of a term $h_m\circ\mathrm{Id}$, especially because every iterate $\tilde{x}^{(i)}$, as well as $x^{(i)}$ if $\rho\leq 1$, belongs to $\mathrm{dom}(g)$; for instance, if $g=\imath_\Omega$, then $\tilde{x}^{(i)}\in\Omega$, for every $i\geq 1$.

III. Application to Inverse Problems Regularized by the Total Variation

A. Formulation of the Method

In this section, we consider inverse problems in imaging. So, we first place ourselves in the space $\mathcal{X}=\mathbb{R}^{N_\mathrm{h}\times N_\mathrm{v}}$ of grayscale images of size N_h columns times N_v rows, endowed with the usual Euclidean inner product. We want to restore or reconstruct an image $\hat{x}$ by solving

$$\text{Find}\quad \hat{x}\in\operatorname*{arg\,min}_{x\in\Omega}\ \tfrac{1}{2}\|Ax-y\|^2+\lambda.\mathrm{TV}(x),\tag{4}$$

where

- y, which lives in a real Hilbert space $\mathcal{Y}$, represents the available data.
- $A:\mathcal{X}\rightarrow\mathcal{Y}$ is the linear operator modeling the acquisition process.
- Ω is a closed and convex subset of $\mathcal{X}$.
- $\lambda>0$ is a tradeoff parameter to tune, depending on the properties of A and the noise level.

The discrete total variation, denoted by TV in (4), is defined as follows. We define the discrete gradient operator $D:\mathcal{X}\rightarrow\mathcal{X}^2$, which maps an image x to a pair of images $(u_\mathrm{h},u_\mathrm{v})$ with, for every $k_\mathrm{h}=1,\dots,N_\mathrm{h}$, $k_\mathrm{v}=1,\dots,N_\mathrm{v}$,
$u_\mathrm{h}[k_\mathrm{h},k_\mathrm{v}]=\big\{x[k_\mathrm{h},k_\mathrm{v}]-x[k_\mathrm{h}-1,k_\mathrm{v}]\text{ if }k_\mathrm{h}\geq 2,\ 0\text{ else}\big\}$,
$u_\mathrm{v}[k_\mathrm{h},k_\mathrm{v}]=\big\{x[k_\mathrm{h},k_\mathrm{v}]-x[k_\mathrm{h},k_\mathrm{v}-1]\text{ if }k_\mathrm{v}\geq 2,\ 0\text{ else}\big\}$.
Note that $\|D^*D\|\leq 8$ [20]. Then we have $\mathrm{TV}(x)=\|Dx\|_{1,2}$, where $\|(u_\mathrm{h},u_\mathrm{v})\|_{1,2}=\sum_{k_\mathrm{h}=1}^{N_\mathrm{h}}\sum_{k_\mathrm{v}=1}^{N_\mathrm{v}}\sqrt{u_\mathrm{h}[k_\mathrm{h},k_\mathrm{v}]^2+u_\mathrm{v}[k_\mathrm{h},k_\mathrm{v}]^2}$. Let us set $h=\lambda\|\cdot\|_{1,2}$. For every $\sigma>0$, we have

$$\mathrm{prox}_{\sigma h^*}:(u_\mathrm{h},u_\mathrm{v})\mapsto(u_\mathrm{h},u_\mathrm{v})/\max\big\{\sqrt{u_\mathrm{h}^2+u_\mathrm{v}^2})/\lambda,1\big\},\tag{5}$$

which does not depend on σ, and for which the operations are to be understood as pixelwise.

Hence, the problem (4) can be put under the form (2), with:

- $f(x)=\frac{1}{2}\|Ax-y\|^2$, whose gradient, given in Fig. 2, has Lipschitz constant $\beta=\|A\|^2$.
- $g(x)=\imath_\Omega(x)$.
- $M=1$, so that we omit the index m for simplicity, $\mathcal{U}=\mathcal{X}^2$, $\lambda.\mathrm{TV}=h\circ L$ with $h=\lambda\|\cdot\|_{1,2}$, $L=D$.

Now we turn our attention to the case of color images. A color image $\mathbf{x}$ has three red (R), green (G), blue (B) channels x^R, x^G, x^B, which can be manipulated like grayscale images. Equivalently, the pixel value of $\mathbf{x}$ at location $\mathbf{k}=(k_\mathrm{h},k_\mathrm{v})$ is the vector $\mathbf{x}[\mathbf{k}]=\big[x^R[\mathbf{k}]\ x^G[\mathbf{k}]\ x^B[\mathbf{k}]\big]^\mathrm{T}$. It is well known that the R, G, B channels of natural images are strongly correlated. So, it is often better to work within a luminance-chrominance representation where, in first approximation, the information is decorrelated. So, we define the orthonormal change of basis which maps a vector in the R, G, B basis to a vector of luminance, green-red and yellow-blue opponent chrominance:

$$\begin{bmatrix} x^L\\ x^{G/R}\\ x^{Y/B}\end{bmatrix}=\begin{bmatrix}\frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}}\\ \frac{-1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0\\ \frac{1}{\sqrt{6}} & \frac{1}{\sqrt{6}} & \frac{-2}{\sqrt{6}}\end{bmatrix}\begin{bmatrix} x^R\\ x^G\\ x^B\end{bmatrix}.\tag{6}$$

Note that $\mathbf{W}^{-1}=\mathbf{W}^\mathrm{T}$, where $\mathbf{W}$ is the 3×3 matrix in (6). Then we introduce the *color total variation* as a regularizer of color images: $\mathrm{TV}(\mathbf{x})=\sum_{\mathbf{k}\in N_\mathrm{h}\times N_\mathrm{v}}\sqrt{\|\mathbf{u}_\mathrm{h}[\mathbf{k}]\|^2+\|\mathbf{u}_\mathrm{v}[\mathbf{k}]\|^2}$, where $(\mathbf{u}_\mathrm{h},\mathbf{u}_\mathrm{v})=\mathbf{D}\mathbf{x}$ is the pair of color images such that $(u_\mathrm{h}^L,u_\mathrm{v}^L)=\mu Dx^L$, $(u_\mathrm{h}^{G/R},u_\mathrm{v}^{G/R})=Dx^{G/R}$, $(u_\mathrm{h}^{Y/B},u_\mathrm{v}^{Y/B})=Dx^{Y/B}$. The real parameter $\mu>0$ in the

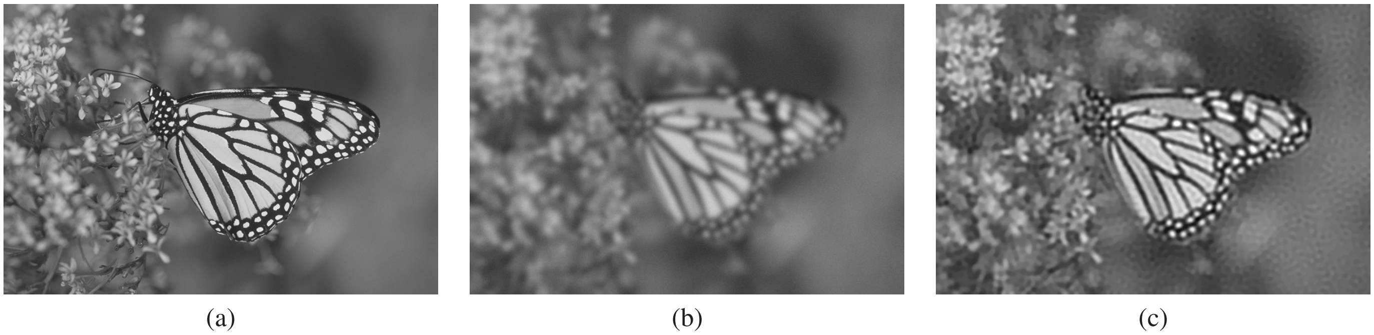

Fig. 3. Deconvolution example: the image (b) is a blurred and noisy version of the unknown image (a) to estimate. The restored image (c) is the solution to the problem (4), obtained with the proposed Algorithm 1, after 300 iterations. The blurring filter is Gaussian (std. dev. 5), the noise is white and Gaussian (std. dev. 3), $\lambda = 0.02$ in (4) and, in the algorithm, $\sigma =$ 1e-4, $\tau = 0.99/(0.5 + 8\sigma)$, $\rho = 1$. The images u_h, u_v are initialized with zeros and $x^{(0)} = y$.

previous definition plays a crucial role, as it controls the balance between the regularization of the luminance and of the chrominance in the reconstructed image. For $\mu = 1$, the color total variation reverts to the usual vectorial total variation in the R, G, B basis [26]. But for many problems, a value of μ closer to zero gives better results, since it reflects the prior knowledge that the hue in natural images is smoother than the luminance. We have $\|\mathbf{D}\| = \max\{\mu, 1\}\|D\|$.

B. Experiments

We first consider the classical problem of deconvolution of a grayscale image, as illustrated in Fig. 3. We solve the problem (4) with A a lowpass convolution operator, with symmetric boundary conditions, so that $\|A\| = 1$. We set $\Omega = [0, 255]^{N_\mathrm{h} \times N_\mathrm{v}}$, so that $\mathrm{prox}_{\tau g} = P_\Omega$ is clipping: the pixel values in the image larger than 255 or lower than 0 are set to 255 and 0, respectively. This choice is known to be better than $\Omega = \mathcal{X}$ to limit the appearance of oscillation artifacts. Note that many optimization algorithms in the literature allow to perform deconvolution, but in most cases, artificial periodic boundary conditions are assumed, in order to use the FFT to invert convolutions in Fourier domain. Since only the operators A and A^* are called with the proposed algorithms, every type of boundary conditions can be used. The method is flexible and can be adapted without difficulty to spatially-varying blur [27], by changing the operator A, or to the presence of Poisson-Gaussian noise, by replacing the least-squares in f by the appropriate negative log-likelihood [28].

To compare the convergence speed with a well known algorithm, we solve the same problem (4) without the constraint $x \in \Omega$, i.e. $g = 0$, with the proposed Algorithm 1 and with the alternating direction method of multipliers (ADMM) [29], also known as split Bregman [30]–[32]. In our case, ADMM consists in iterating [32]:

$$\left\lfloor\begin{array}{l} z^{(i+1)} := \mathrm{prox}_{\lambda\|\cdot\|_{1,2}/\alpha}(Dx^{(i)} - p^{(i)}/\alpha),\\ x^{(i+1)} := (\alpha D^* D + A^* A)^{-1}(A^* y + \alpha D^* z^{(i+1)} + D^* p^{(i)}),\\ p^{(i+1)} := p^{(i)} + \alpha(z^{(i+1)} - Dx^{(i+1)}). \end{array}\right.$$

At every iteration, the linear system is solved approximately with one Richardson iteration. Note that the guarantee of convergence is lost in that case. We consider the same conditions as in Fig. 3 and $\alpha =$ 1e-3. The number i of iterations to reach a RMSE $\|\hat{x} - x^{(i)}\|/\sqrt{N_\mathrm{h} N_\mathrm{v}}$ of 2 gray levels is 3481 and 3608 with the proposed Algorithm 1 and ADMM, respectively.

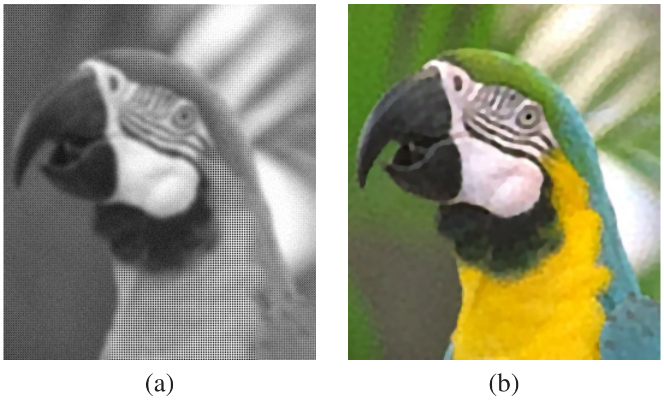

Fig. 4. Joint demosaicing-deconvolution of the image v depicted in (a). In (b), the reconstructed image solution to (4) with $\lambda = 1.5$, $\mu = 0.2$, obtained with 300 iterations of Algorithm 1. The blurring filter is Gaussian (std. dev. 2), the noise is white and Gaussian (std. dev. 5), $\sigma = 0.03$, $\tau = 0.99/(0.5 + 8\sigma)$, $\rho = 1$. $\mathbf{u}_\mathrm{h}^{(0)}$, $\mathbf{u}_\mathrm{v}^{(0)}$ are set to zero and $x^{(0),R} = x^{(0),G} = x^{(0),B} = y$.

The second experiment consists in reconstructing a color image by joint deblurring-demosaicing-denoising, see [33], [34] for a presentation of the problem. We solve (4) with $A = MB$, where B is the same blurring operator as in the first experiment, applied on each R, G, B channel, and M is the Bayer mosaicing operator [33], [34]. We have $\|A\| = 1$. We set $\Omega = [0, 255]^{N_\mathrm{h} \times N_\mathrm{v} \times 3}$.

Matlab code implementing Algorithm 1 and generating the images in Figs. 3,4 is available on the webpage of the author.

IV. Conclusion

We proposed two algorithms to exactly solve a large class of convex optimization problems. The simplicity, universality and ease of implementation of the algorithms make them well suited to prototyping new methods, for instance to test various types of regularization for some inverse problem. However, the algorithms do not exploit any further structure of the problem at hand that may be present. So, our future work will focus on the theoretical study of the convergence rates, on the development of possible accelerations, and on the practical comparison, in terms of computation time and memory usage, with other algorithms, for several typical large-scale problems.

References

[1] S. Becker, E. Candès, and M. Grant, "Templates for convex cone problems with applications to sparse signal recovery," *Mathematical Programming Computation*, vol. 3, no. 3, pp. 165–218, Sept. 2011.

[2] S. Anthoine, J. F. Aujol, C. Mélot, and Y. Boursier, "Some proximal methods for Poisson intensity CBCT and PET," *Inverse Problems and Imaging*, vol. 6, no. 4, Nov. 2012.

[3] E. Y. Sidky, J. H. Jørgensen, and X. Pan, "Convex optimization problem prototyping for image reconstruction in computed tomography with the Chambolle–Pock algorithm," *Physics in Medicine and Biology*, vol. 57, no. 10, pp. 3065–3091, 2012.

[4] P. Chen, J. Huang, and X. Zhang, "A primal-dual fixed point algorithm for convex separable minimization with applications to image restoration," *Inverse Problems*, vol. 29, no. 2, 2013.

[5] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski, "Optimization with sparsity-inducing penalties," *Foundations and Trends in Machine Learning*, vol. 4, no. 1, pp. 1–106, 2012.

[6] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *Journal of ACM*, vol. 58, no. 1, pp. 1–37, 2009.

[7] D. Cremers, T. Pock, K. Kolev, and A. Chambolle, "Convex relaxation techniques for segmentation, stereo and multiview reconstruction," in *Markov Random Fields for Vision and Image Processing*. MIT Press, 2011.

[8] J. Lellmann, D. Breitenreicher, and C. Schnörr, "Fast and exact primal-dual iterations for variational problems in computer vision," in *Proc. of ECCV: Part II*, Heraklion, Crete, Greece, 2010, pp. 494–505.

[9] J.-L. Starck, F. Murtagh, and J. Fadili, *Sparse Image and Signal Processing: Wavelets, Curvelets, Morphological Diversity*. Cambridge University Press, 2010.

[10] A. Chambolle, V. Caselles, D. Cremers, M. Novaga, and T. Pock, "An introduction to total variation for image analysis," in *Theoretical Foundations and Numerical Methods for Sparse Recovery*, vol. 9. De Gruyter, Radon Series Comp. Appl. Math., 2010, pp. 263–340.

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

[12] C. Chaux, P. L. Combettes, J.-C. Pesquet, and V. R. Wajs, "A variational formulation for frame based inverse problems," *Inverse Problems*, vol. 23, pp. 1495–1518, June 2007.

[13] P. L. Combettes and J.-C. Pesquet, "Proximal splitting methods in signal processing," in *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, H. H. Bauschke, R. Burachik, P. L. Combettes, V. Elser, D. R. Luke, and H. Wolkowicz, Eds. New York: Springer-Verlag, 2010.

[14] P. Machart, S. Anthoine, and L. Baldassarre, "Optimal computational trade-off of inexact proximal methods," 2012, preprint arXiv:1210.5034.

[15] L. Condat, "A primal-dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms," *J. Optimization Theory and Applications*, vol. 158, no. 2, pp. 460–479, 2013.

[16] P. L. Combettes and J.-C. Pesquet, "Primal–dual splitting algorithm for solving inclusions with mixtures of composite, Lipschitzian, and parallel-sum type monotone operators," *Set-Valued and Variational Analysis*, vol. 20, no. 2, pp. 307–330, 2012.

[17] B. C. Vũ, "A splitting algorithm for dual monotone inclusions involving cocoercive operators," *Advances in Computational Mathematics*, vol. 38, no. 3, pp. 667–681, Apr. 2013.

[18] T. Pock and A. Chambolle, "Diagonal preconditioning for first order primal–dual algorithms in convex optimization," in *Proc. of ICCV*, Nov. 2011.

[19] P. L. Combettes and B. C. Vũ, "Variable metric forward–backward splitting with applications to monotone inclusions in duality," *Optimization*, 2013, to be published.

[20] A. Chambolle and T. Pock, "A first-order primal–dual algorithm for convex problems with applications to imaging," *Journal of Mathematical Imaging and Vision*, vol. 40, no. 1, pp. 120–145, 2011.

[21] L. M. Briceño-Arias and P. L. Combettes, "A monotone+skew splitting model for composite monotone inclusions in duality," *SIAM J. Optim.*, vol. 21, no. 4, pp. 1230–1250, 2011.

[22] H. Raguet, J. Fadili, and G. Peyré, "Generalized forward–backward splitting," *SIAM Journal on Imaging Sciences*, vol. 6, no. 3, pp. 1199–1226, 2013.

[23] I. Loris and C. Verhoeven, "On a generalization of the iterative soft-thresholding algorithm for the case of non-separable penalty," *Inverse Problems*, vol. 27, no. 12, 2011.

[24] J.-C. Pesquet and N. Pustelnik, "A parallel inertial proximal optimization method," *Pacific Journal of Optimization*, vol. 8, no. 2, pp. 273–305, Apr. 2012.

[25] L. M. Briceño-Arias, "Forward–Douglas–Rachford splitting and forward–partial inverse method for solving monotone inclusions," *Optimization*, 2013, to be published.

[26] X. Bresson and T. F. Chan, "Fast dual minimization of the vectorial total variation norm and applications to color image processing," *Inverse problems and Imaging*, vol. 2, no. 4, pp. 455–484., Nov. 2008.

[27] F. Soulez, É. Thiébaut, and L. Denis, "Restoration of hyperspectral astronomical data with spectrally varying blur," *EAS Publications Series*, vol. 59, pp. 403–416, 2013.

[28] A. Jezierska, E. Chouzenoux, J.-C. Pesquet, and H. Talbot, "A convex approach for image restoration with exact Poisson-Gaussian likelihood," 2013, preprint hal-00922151, submitted to *IEEE Trans. Image Proc.*

[29] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[30] T. Goldstein and S. Osher, "The split Bregman method for L1-regularized problems," *SIAM Journal on Imaging Sciences*, vol. 2, no. 2, pp. 323–343, 2009.

[31] S. Setzer, "Operator splittings, Bregman methods and frame shrinkage inimage processing," *International Journal of Computer Vision*, vol. 92, no. 3, pp. 265–280, 2011.

[32] E. Esser, "Applications of Lagrangian-based alternating direction methods and connections to split Bregman," 2009, tech. Rep. 09-31, ULCA.

[33] F. Soulez and É. Thiébaut, "Joint deconvolution and demosaicing," in *Proc. of IEEE ICIP*, Cairo, Egypt, Nov. 2009, pp. 145–148.

[34] L. Condat and S. Mosaddegh, "Joint demosaicking and denoising by total variation minimization," in *Proc. of IEEE ICIP*, Orlando, USA, Sept. 2012.