A Bivariate Threshold For ℓ_1 Regularized Problems

İlker Bayram

***Abstract*—We introduce a bivariate threshold function for ℓ_1 penalized minimization problems. The threshold function is derived by solving an ℓ_1 penalized inverse problem of size two. The bivariate function can be used to accelerate the convergence of iterative algorithms that are often used in sparse signal processing. We discuss how to use the function in practice and demonstrate the improvement in convergence rate with numerical experiments.**

I. Introduction

A central problem in sparse signal processing is

$$\min_x \left\{ h(x) = \frac{1}{2}\|y - Hx\|_2^2 + \|x\|_1 \right\}, \tag{1}$$

where y denotes data (given) and H is an observation matrix. A widely used algorithm for solving this minimization problem is the iterative shrinkage/thresholding algorithm (ISTA) [15], [19]. ISTA can be derived within different frameworks such as majorization-minimization (MM) [15], [14], sequential unconstrained minimization [8], [7], forward-backward splitting [11], [10]. In all of these frameworks, the idea is to replace the original minimization problem (1) with a sequence of easier problems and construct a sequence or iterates that converges to a minimizer. In particular, at the k^{th} iteration, MM employs a non-negative convex function $g_k(x)$ defined for $\alpha \geq \sigma(H^T H)$ as

$$g_k(x) = \left\langle x - x^k, (\alpha I - H^T H)(x - x^k) \right\rangle, \tag{2}$$

and sets

$$x^{k+1} = \arg\min_x \; h(x) + g_k(x). \tag{3}$$

For the special choice of g_k in (2), the minimization problem in (3) can be written, after some arrangements, as

$$x^{k+1} = \arg\min_x \; \frac{\alpha}{2}\|x - z^k\|_2^2 + \|x\|_1, \tag{4}$$

where

$$z^k = x^k - \alpha^{-1} H^T \left(H\, x^k - y\right). \tag{5}$$

The solution of (4) is obtained by soft-thresholding z^k [14].

The special choice of g_k given in (2) is not mandatory for MM [17] but it is not arbitrary either. This special g_k ensures that the subsequent minimization step (3) has a simple solution, namely soft-thresholding. In contrast, if we set

$$g_k(x) = \left\langle x - x^k, (M - H^T H)(x - x^k) \right\rangle \tag{6}$$

for some positive definite M with $M > H^T H$, which is otherwise arbitrary, the subsequent minimization problem might be as hard as the original problem. In this letter, we propose to employ a block-diagonal M with 2×2 blocks. For such a choice, the subsequent minimization problem is solved by a bivariate penalty, which will be derived in the sequel. Using a block-diagonal M allows to find a tighter upper bound for $H^T H$. This in turn leads to non-negative functions g_k closer to zero and the subsequent problems (3) get closer to the original. We demonstrate numerically that such a modification can lead to noticeable acceleration of the algorithm. We also discuss how to select the block diagonal M for a given $H^T H$.

The idea of accelerating ISTA using an M other than αI has been explored in the context of wavelet-regularized deconvolution in [21], [22], [2], [23]. Specifically, [21] considers a tighter diagonal matrix upper bound for $H^T H$ when H is the composition of a convolutive blur operator and the Shannon wavelet analysis operator. [2] presents a generalization of this approach for an arbitrary wavelet frame (see also [22] for an alternative scheme). [23] discusses how to find a tighter diagonal upper bound, thereby accelerating convergence. All of these papers essentially seek diagonal M matrices and consequently employ soft-thresholding. In contrast, the current letter employs a bivariate threshold derived from a block-diagonal M. Such an extension has not previously appeared in the literature as far as we are aware.

The proposed bivariate threshold function can be used in other iterative algorithms for ℓ_1 regularized problems such as the Douglas-Rachford algorithm [10], [12], alternating direction method of multipliers [5] or saddle-point algorithms [9], [13]. However, we will restrict our attention to ISTA in this letter.

In the following section, we derive the bivariate threshold and provide an algorithm that uses it for solving (1). We discuss how to find block-diagonal M matrices in Section III. In Section IV, we demonstrate numerically the acceleration gained by using the bivariate threshold. Section V is the conclusion.

II. The Bivariate Penalty

For the choice of g_k in (6), the minimization problem in (3) becomes

$$x^{k+1} = \arg\min_x \; \frac{1}{2}\left\langle x - x^k, M(x - x^k)\right\rangle + \left\langle x, H^T(H\,x^k - y)\right\rangle + \|x\|_1. \tag{7}$$

The optimality conditions for (7) are

$$0 = Mx^{k+1} - \left(Mx^k - H^T(Hx^k - y)\right) + s, \quad \text{for some } s \in \operatorname{sgn}(x^{k+1}), \tag{8}$$

İ. Bayram is with the Dept. of Electronics and Telecommunications Eng., Istanbul Technical University, Istanbul, Turkey. E-mail : ibayram@itu.edu.tr.

where $\operatorname{sgn}(\cdot)$ is a set-valued separable mapping defined on the real line as

$$\operatorname{sgn}(t) = \begin{cases} \{-1\}, & \text{if } t < 0, \\ [-1,1], & \text{if } t = 0, \\ \{1\}, & \text{if } 0 < t. \end{cases} \tag{9}$$

Notice that if M is block-diagonal, then the minimization problem (7) and the optimality conditions (8) are block-separable. Therefore, for a block-diagonal M with $2{\times}2$ blocks, it suffices to consider problems of size two.

A. A Bivariate Problem

For ease of notation, consider the variational inclusion problem,

$$z = B\,x + s, \qquad \text{for some } s \in \operatorname{sgn}(x), \tag{OC}$$

where B is a 2×2 positive definite (p.d.) matrix.

Proposition 1. Suppose B is positive definite. Given any z, there is a unique x that satisfies (OC).

Proof. (OC) form the optimality conditions of the problem

$$x \in \arg\min_t \frac{1}{2}\left\langle t - B^{-1}z, B\,(t - B^{-1}z)\right\rangle + \|t\|_1. \tag{10}$$

Since B is p.d., the problem (10) is strictly convex, and has a unique minimum, for any z. Thus follows the claim. □

This proposition justifies the following definition.

Definition 1. Suppose B is positive definite. $T_B : \mathbb{R}^2 \to \mathbb{R}^2$ is defined as the mapping that takes $z \in \mathbb{R}^2$ to $x \in \mathbb{R}^2$, such that (OC) is satisfied.

The formulation in (1) sometimes employs a multiplicative factor for weighting the ℓ_1 term. Such terms can be dealt with easily by a change of variables. In the context of the bivariate problem, we have the following result. The proof can be obtained by a change of variables and is omitted.

Proposition 2. Suppose B is p.d. and

$$z = B\,x + \tau\,s, \qquad \text{for some } s \in \operatorname{sgn}(x), \tag{11}$$

where $\tau > 0$. Then, $x = \tau\,T_B(z/\tau)$.

We now provide an expression for T_B.

Proposition 3. Suppose B is a positive definite matrix given as $B = \begin{bmatrix} a & b \\ b & c \end{bmatrix}$, and $x = T_B(z)$. The following cases are mutually exclusive and they cover all possibilities for z.

(a) If $|z_i| \le 1$, for $i = 1, 2$, then $x_1 = x_2 = 0$.

(b) If

$$|z_1| > 1, \tag{12a}$$

$$\left| z_2 - \frac{b}{a}\big(z_1 - \operatorname{sgn}(z_1)\big) \right| \le 1, \tag{12b}$$

then

$$x_1 = \big(z_1 - \operatorname{sgn}(z_1)\big)/a, \tag{13a}$$

$$x_2 = 0. \tag{13b}$$

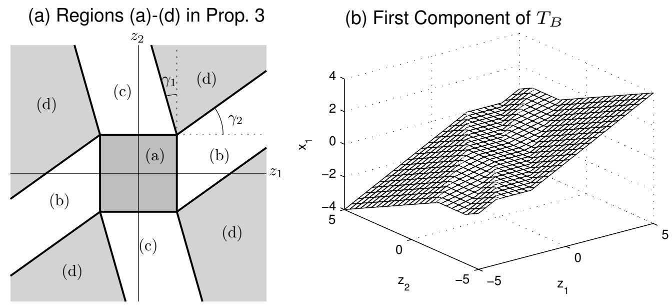

Fig. 1. (a) The regions described in (a)-(d) in Prop. 3 that partition $\mathbb{R}^2$. (b) The first component of T_B for a special choice of B.

(c) If

$$\left| z_1 - \frac{b}{c}\big(z_2 - \operatorname{sgn}(z_2)\big) \right| \le 1, \tag{14a}$$

$$|z_2| > 1, \tag{14b}$$

then

$$x_1 = 0, \tag{15a}$$

$$x_2 = \big(z_2 - \operatorname{sgn}(z_2)\big)/a. \tag{15b}$$

(d) If for some (r, s) pair taking values in $\{0, 1\}$, we have

$$\begin{bmatrix} (-1)^r & 0 \\ 0 & (-1)^s \end{bmatrix} B^{-1} \left(z - \begin{bmatrix} (-1)^r \\ (-1)^s \end{bmatrix} \right) > 0 \tag{16}$$

then

$$x = B^{-1} \left(z - \begin{bmatrix} (-1)^r \\ (-1)^s \end{bmatrix} \right). \tag{17}$$

Proof. See the appendix. □

In the proposition, the four cases (a)-(d) partition $\mathbb{R}^2$ into nine regions, as depicted in Fig. 1a. In this figure, the two angles γ_1 and γ_2 depend on B and satisfy

$$\cot\gamma_1 = \frac{b}{a}, \quad \cot\gamma_2 = \frac{b}{c}. \tag{18}$$

In the special case $a = c$, we have $\gamma_1 = \gamma_2$ and the lines delimiting the regions defined in (d) become perpendicular. We remark that in general, the signs of z may not coincide with the signs of x. This is also evident from Fig. 1a, where the two regions in the lower left and upper right contain $z \in \mathbb{R}^2$ with different signs, even though the signs of x remain constant ($|x_i| > 0$ in these regions). The first component of T_B is shown in Fig. 1b,c for $B = \begin{bmatrix} 4 & 3 \\ 3 & 4 \end{bmatrix}$.

Pseudocode for the modified ISTA that uses T_B is provided in Algorithm 1. In this algorithm, we assume that M is block-diagonal with 2×2 blocks and the n^{th} block is denoted by B_n. The number of blocks is equal to N, so that the number of variables is $2N$.

B. Convergence of the Algorithm

If $M \ge H^T T$, the discussion in the beginning of Sec. II implies that Algorithm 1 decreases the cost monotonically. In fact, it can be further shown that if a minimizer is not reached, the descent is strict. It then follows by the global convergence theorem [18] that the cost values visited by the

Algorithm 1 ISTA with a Bivariate Threshold

repeat

$z \leftarrow (M - H^T H)x + H^T y$

for n = 1:N **do**

$\begin{bmatrix} x_{2n-1} \\ x_2 n \end{bmatrix} \leftarrow T_{B_n}\left(\begin{bmatrix} z_{2n-1} \\ z_{2n} \end{bmatrix}\right)$

until some convergence criterion is met

algorithm converge to the minimum possible. This observation alone does not automatically imply that the sequence of x's converges. We remark that this is also the case for regular ISTA – see e.g. the discussion in [14] which refers to [11] for the convergence of the actual iterates. The convergence proof of ISTA in [11] (see also [1]), does not directly cover Algorithm 1. However, if we work with a modified inner product defined as $\langle x, y\rangle_M = \langle x, My\rangle$, then the proof in [11], [1] can be adapted (this is also noted in [2], without going into details). In fact, it can further be shown by the study in [11], [1] that Algorithm 1 converges even if $M > \frac{1}{2} H^T H$. However, this discussion is beyond the scope of the letter.

III. Selection of the Upper Bound Matrix

The only point left out in the discussion so far is the selection of M with $M > H^T H$. We propose two methods for this problem in this section.

A. Semi-Definite Programming

Given $H^T H$, we would like M to satisfy $M > H^T H$ but we also would like M to be 'small'. One way to achieve this is to look for a block-diagonal M with $M \geq H^T H$ such that if $M' \neq M$ is another block-diagonal matrix with $M' \geq H^T H$, then $(M - M')$ is not positive semi-definite. This is a vector optimization problem [6]. In order to obtain a Pareto-optimal solution, we consider the modified convex problem, aiming to minimize the trace of M, as [1]

$$\min \operatorname{tr}(M) \text{ subject to } \begin{cases} M \geq H^T H, \\ M \text{ is block-diagonal.} \end{cases} \tag{19}$$

This problem is an instance of a semi-definite program [6] and it can be solved with available software, such as SeDuMi [20].

B. Gershgorin Discs

An alternative, that is especially useful for diagonally dominant $H^T H$ is based on Gershgorin's theorem. For symmetric matrices, Gershgorin's theorem can be stated as follows.

Theorem 1. [16] Suppose A is a symmetric matrix and define

$$b_k = \sum_{j \neq k} |A_{k,j}|. \tag{20}$$

Then all of the eigenvalues of A lie in the intervals $[A_{k,k} - b_k, A_{k,k} + b_k]$.

[1]To ensure the convergence of the algorithm, we actually need $M > H^T H$ (the inequality should be strict) but this can be achieved by adding ϵI to M obtained via (19).

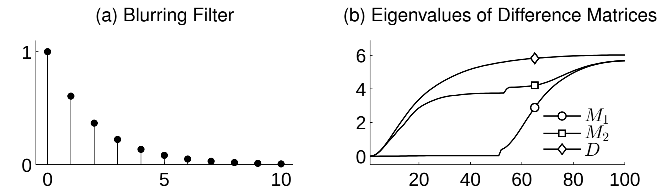

Fig. 2. (a) The filter used in the experiments, (b) the eigenvalues of the difference matrices $M_1 - \tau^2 H^T H$, $M_2 - \tau^2 H^T H$, $D - \tau^2 H^T H$–see the text for the description of these matrices.

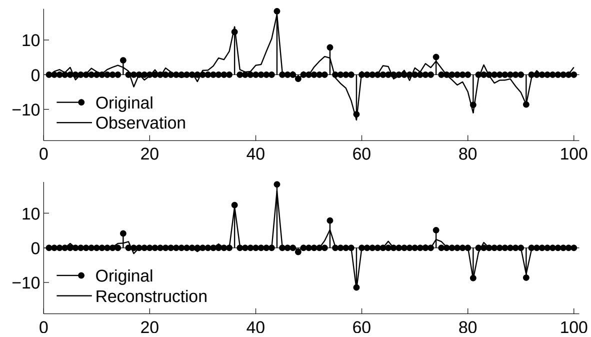

Fig. 3. (a) Desired original object and the noisy observations, (b) original object and reconstruction found by ISTA.

As a corollary, if $A_{k,k} > b_k$ for all k, then A is positive definite. Based on this observation, we set,

$$a_k = \sum_{j \neq 2k} \left|(H^T H)_{2k-1,j}\right|, \tag{21a}$$

$$c_k = \sum_{j \neq 2k-1} \left|(H^T H)_{2k,j}\right|, \tag{21b}$$

$$b_k = (H^T H)_{2k-1,2k}, \tag{21c}$$

and construct $B_k = \begin{bmatrix} a_k & b_k \\ b_k & c_k \end{bmatrix}$. We then define M to be the block-diagonal matrix whose k^{th} diagonal-block is B_k. It follows by Gershgorin's theorem that $M \geq H^T H$.

IV. Experiment

In this section, we demonstrate the utility of the proposed threshold function via a numerical experiment. Matlab code for the experiment is available at "`http://web.itu.edu.tr/ibayram/Bivariate`".

We set H to be a convolution matrix derived from the filter shown in Fig. 2a. This filter is an exponentially decaying sequence with 11 elements.

We use the sparse signal x shown in Fig. 3 as the object to be recovered. We form the observations as $y = \tau H x + n$, where n is white Gaussian noise with $\sigma = 1/3$ and τ is adjusted such that the SNR is 10 dB. The noisy observation y is shown in Fig. 3a.

In order to find block-diagonal matrices that upper bound $\tau^2 H^T H$, we used both methods discussed in Sec. III and obtained the matrices M_1 (using the method in Sec. III-A), M_2 (using the method in Sec. III-B). For comparison, we also computed a diagonal matrix D with $D \geq \tau^2 H^T H$ via semidefinite programming as discussed in Sec. III-A. To see how well these matrices approximate $\tau^2 H^T H$, we

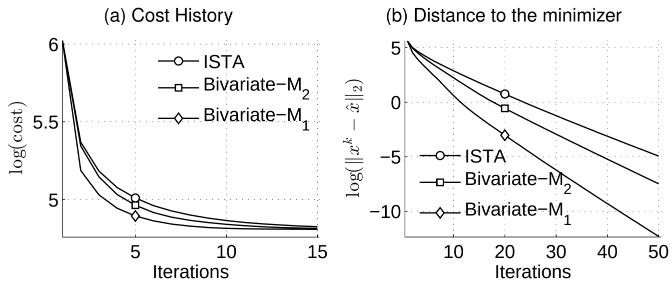

Fig. 4. Performance comparison for ISTA and Algorithm 1 with M_1 and M_2. (a) Cost history, (b) distance to the minimizer.

computed the eigenvalues of the difference of M_1, M_2, D from $\tau^2\, H^T\, H$. These eigenvalues are shown in increasing order in Fig. 2b. The figure suggests that the best fit is M_1, followed by M_2, in agreement with our expectation.

In order to assess the convergence speed of the algorithms, we first ran ISTA (using D) for 10K iterations to obtain an estimate of the limit. This sequence is shown in Fig. 3b. Then we ran Algorithm 1 using M_1, M_2. The history of the cost function as well as the distance to the limit is shown in Fig. 4. We see from this figure that using Algorithm 1 instead of ISTA (with D) leads to a clear improvement in speed, both in terms of reducing cost and convergence rate.

It is possible to accelerate ISTA using combinations of subsequent iterates [3], [4]. These acceleration schemes are also formally applicable for Algorithm 1. In our experiments with FISTA [3], we found that all of the algorithms discussed above can be accelerated noticeably. However, the relative convergence rates remain more or less the same.

V. Conclusion

We derived a bivariate threshold function motivated by an ℓ_1 regularized minimization problem. We demonstrated that this bivariate function can be used to accelerate iterative shrinkage type algorithms by allowing to use majorizers that provide a better fit to the quadratic term.

A natural extension of this work would be to seek analytic expressions for threshold functions containing more variables. An alternative is to seek threshold functions that allow to work with banded matrices. The challenge is to find expressions that are easy to interpret and/or implement. We plan to investigate such extensions in future work.

The proposed threshold function might also be useful in understanding how bivariate threshold functions should behave. In particular, the function derived in this paper behaves quite differently than a soft-threshold and it contains non-rectangular deadzones (see Fig. 1). This might provide some intuition for designing threshold functions from scratch, without necessarily going through a minimization problem.

Appendix
Proof of Prop. 3

The following four mutually exclusive cases exhaust all possibilities for x : (i) $\{x_1 = 0, x_2 = 0\}$, (ii) $\{x_1 \neq 0, x_2 = 0\}$, (iii) $\{x_1 = 0, x_2 \neq 0\}$, (iv) $\{x_1 \neq 0, x_2 \neq 0\}$. We study these cases separately to reach an explicit expression for the bivariate threshold function.

Lemma 1. $\{x_1 = 0,\, x_2 = 0\}$ iff $\{|z_1| \leq 0,\, |z_2| \leq 1\}$.

Proof. This follows directly from (OC). □

Lemma 2. $\{x_1 \neq 0,\ x_2 = 0\}$ if and only if

$$|z_1| > 1, \tag{22a}$$

$$\left| z_2 - \frac{b}{a}\big(z_1 - \operatorname{sgn}(z_1)\big) \right| \leq 1. \tag{22b}$$

In this case, x and z are related as $x_1 = (z_1 - \operatorname{sgn}(z_1))/a$.

Proof. Suppose $\{x_1 > 0,\ x_2 = 0\}$. By (OC), we must have

$$z_1 = a\, x_1 + \operatorname{sgn}(x_1) \tag{23a}$$

$$z_2 = b\, x_1 + s_2, \text{ for } s_2 \in [-1, 1]. \tag{23b}$$

Observe from the first equation that since $a > 0$, in order for equality to hold, we must have $\operatorname{sgn}(x_1) = \operatorname{sgn}(z_1)$. Solving for x_1 and s_2, we obtain

$$x_1 = (z_1 - \operatorname{sgn}(z_1))/a, \tag{24a}$$

$$s_2 = z_2 - \frac{b}{a}(z_1 - \operatorname{sgn}(z_1)). \tag{24b}$$

Since $x_1 \neq 0$, we must have $|z_1| > 1$. Also, the restriction $s_2 \in [-1, 1]$ can be stated equivalently as $\left| z_2 - \frac{b}{a}(z_1 - 1)\right| \leq 1$.

For the converse, suppose (22) hold. Set x_1 and s_2 as in (24) and observe that the optimality conditions in (23) hold. Thus the optimality conditions are satisfied for $x_1 \neq 0$, $x_2 = 0$. □

Lemma 3. $\{x_1 = 0,\ x_2 \neq 0\}$ if and only if

$$\left| z_1 - \frac{b}{c}\big(z_2 - \operatorname{sgn}(z_2)\big) \right| \leq 1, \tag{25a}$$

$$|z_2| > 1. \tag{25b}$$

In this case, x and z are related as $x_2 = (z_2 - \operatorname{sgn}(z_2))/a$.

Proof. Follows by symmetry from Lemma 2. □

Lemma 4. $\{x_1 \neq 0,\ x_2 \neq 0\}$ if and only if there exists r, s taking values in $\{0, 1\}$ such that

$$\begin{bmatrix} (-1)^r & 0 \\ 0 & (-1)^s \end{bmatrix} B^{-1} \left(z - \begin{bmatrix} (-1)^r \\ (-1)^s \end{bmatrix} \right) > 0. \tag{26}$$

In this case, x and z are related as

$$x = B^{-1} \left(z - \begin{bmatrix} (-1)^r \\ (-1)^s \end{bmatrix} \right). \tag{27}$$

Proof. Notice that a set of conditions in terms of B^{-1}, equivalent to (OC) are,

$$B^{-1}\, z = x + B^{-1} s, \qquad \text{for some } s \in \operatorname{sgn}(x). \tag{OC$_2$}$$

Suppose $x_i \neq 0$. By (OC$_2$), for $d = \begin{bmatrix} \operatorname{sgn}(x_1) & \operatorname{sgn}(x_2) \end{bmatrix}^T$, we must have $B^{-1}\,(z - d) = x$. But since $x \neq 0$, we also have $\operatorname{diag}(d)\, B^{-1}(z - d) > 0$. Setting $(-1)^r = \operatorname{sgn}(x_1)$, $(-1)^s = \operatorname{sgn}(x_2)$, (26) follows.

For the converse, suppose (26) holds for some r, s pair taking values in $\{0, 1\}$. Set x as in (27). It follows by (26) that $x_1 \neq 0$, $x_2 \neq 0$ and $\operatorname{sgn}(x_1) = (-1)^r$, $\operatorname{sgn}(x_2) = (-1)^s$. Rearranging, we obtain (OC$_2$). Thus follows the claim in the reverse direction. □

Prop. 3 follows as a corollary of these lemmas and the fact that the four cases (i)-(iv) above form a partition of the unity.

REFERENCES

[1] H. H. Bauschke and P. L. Combettes. *Convex analysis and monotone operator theory in Hilbert spaces*. Springer, 2011.

[2] İ. Bayram and I. W. Selesnick. A subband adaptive iterative shrinkage/thresholding algorithm. *IEEE Trans. Signal Processing*, 58(3):1131–1143, March 2010.

[3] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 1:183–202, 2009.

[4] J. M. Bioucas-Dias and M. A. T. Figueiredo. A new TwIST: Two-step iterative shrinkage/thresholding algorithms for image restoration. *IEEE Trans. Image Proc.*, 16(12):2992–3003, December 2007.

[5] 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*, 3(1):1–122, 2011.

[6] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[7] C. L. Byrne. An elementary proof of convergence for the forward-backward splitting algorithm. *Journal of Nonlinear and Convex Analysis*, 15(4):681–691, 2014.

[8] C. L. Byrne. *Iterative Optimization in Inverse Problems*. Taylor and Francis, 2014.

[9] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145, May 2011.

[10] P. L. Combettes and J.-C. Pesquet. Proximal splitting methods in signal processing. In H. H. Bauschke, R. S. Burachik, P. L. Combettes, V. Elser, D. R. Luke, and H. Wolkowicz, editors, *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*. Springer, New York, 2011.

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

[12] J. Eckstein and D. P. Bertsekas. On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators. *Mathematical Programming*, 55(3):293–318, 1992.

[13] E. Esser, X. Zhang, and T. F. Chan. A general framework for a class of first order primal-dual algorithms for convex optimization in imaging science. *SIAM J. Imaging Sciences*, 3(4):1015–1046, November 2010.

[14] M. A. T. Figueiredo, J. M. Bioucas-Dias, and R. D. Nowak. Majorization-minimization algorithms for wavelet-based image restoration. *IEEE Trans. Image Proc.*, 16(12):2980–2991, December 2007.

[15] M. A. T. Figueiredo and R. D. Nowak. An EM algorithm for wavelet-based image restoration. *IEEE Trans. Image Proc.*, 12(8):906–916, August 2003.

[16] R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, 1990.

[17] K. Lange. *Optimization*. Springer, 2004.

[18] D. G. Luenberger. *Linear and Nonlinear Programming*. Addison-Wesley, 1984.

[19] C. De Mol and M. Defrise. A note on wavelet-based inversion methods. In M. Z. Nashed and O. Scherzer, editors, *Inverse Problems, Image Analysis and Medical Imaging*. American Mathematical Society, 2002.

[20] J.F. Sturm. Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11–12:625–653, 1999. Version 1.05 available from `http://fewcal.kub.nl/sturm`.

[21] C. Vonesch and M. Unser. A fast thresholded Landweber algorithm for wavelet-regularized multidimensional deconvolution. *IEEE Trans. Image Proc.*, 17(4):539–549, April 2008.

[22] C. Vonesch and M. Unser. A fast multilevel algorithm for wavelet-regularized image restoration. *IEEE Trans. Image Proc.*, 18(3):509–523, March 2009.

[23] Y. Zhang and N. Kingsbury. Improved bounds for subband-adaptive iterative shrinkage/thresholding algorithms. *IEEE Trans. Image Proc.*, 22(4):1373–1381, April 2013.