Fast Constrained Least Squares Spectral Unmixing using Primal-Dual Interior-Point Optimization

Emilie Chouzenoux, *Member, IEEE*, Maxime Legendre,
Saïd Moussaoui, *Member, IEEE* and Jérôme Idier, *Member, IEEE*

Abstract—Hyperspectral data unmixing aims at identifying the components (endmembers) of an observed surface and at determining their fractional abundances inside each pixel area. Assuming that the spectral signatures of the surface components have been previously determined by an endmember extraction algorithm, or to be part of an available spectral library, the main problem is reduced to the estimation of the fractional abundances. For large hyperspectral image data sets, the estimation of the abundance maps requires the resolution of a large-scale optimization problem subject to linear constraints such as non-negativity and sum less or equal to one. This paper proposes a primal-dual interior-point optimization algorithm allowing a constrained least squares estimation approach. In comparison with existing methods, the proposed algorithm is more flexible since it can handle any linear equality and/or inequality constraint and has the advantage of a reduced computational cost. It also presents an algorithmic structure suitable for a parallel implementation on modern intensive computing devices such as Graphics Processing Units (GPU). The implementation issues are discussed and the applicability of the proposed approach is illustrated with the help of examples on synthetic and real hyperspectral data.

Index Terms—spectral unmixing, constrained least squares, interior-point optimization, primal-dual algorithm, GPU computing.

I. INTRODUCTION

Hyperspectral imaging corresponds to the measurement of the incident light reflection at the ground surface of an observed scene in several contiguous spectral bands. Despite of the high spatial resolution that can be attained by recent imaging devices, the surface area covered by any pixel of the image may contain different components. Therefore, the measured reflectance spectrum in each pixel can be explained as a mixture of the individual component reflectance spectra weighted by the proportion (abundance) of each component in this pixel area.

Unmixing hyperspectral data aims at the identification of the observed surface components (endmembers) and the determination of their fractional abundances inside each pixel area [1, 2]. Fast hyperspectral data unmixing approaches are supervised, by assuming that the endmember spectra are part of an available spectral library or can be provided by an endmember extraction algorithm [3, 4]. Then, the remaining step of the unmixing is the estimation of the fractional abundances. Actually, there is an increasing interest to joint estimation

methods based either on non-negative source separation [5, 6] or constrained non-negative matrix factorization [7, 8]. However, the purpose of this paper is to focus on the second step of the supervised approach with the aim to present a fast computation method adapted to the case of large data sets.

Usual algorithms for solving the spectral unmixing problem consist in minimizing a data fitting measure (generally a least squares criterion) under the physical constraints of non-negativity and sum-to-one. For instance, the former constraint leads to the *non-negative least squares* algorithm (NNLS) [9, 10], and the latter is handled by the *sum-to-one constrained least squares* (SCLS) method [11]. Both constraints are accounted for by the *fully constrained least squares* (FCLS) algorithm [12]. In [13], a Bayesian inference algorithm incorporating jointly these constraints is proposed. It is based on Monte Carlo Markov chain methods and offers the advantage of estimating the number of components. However, all these mentioned methods suffer from a significant increase of the computation time in the case of large data sets (in terms of image size, number of components or number of spectral bands). In order to reduce the computation time, many recent contributions have investigated the use of parallel computing tools [14] such as graphics processing units (GPUs) [15] and FPGA based-design [16]. A geometrical formulation of the abundance estimation step has been recently proposed in [17], the computation cost being reduced by retrieving some quantities computed during the endmember extraction step or by using simplex projection methods [18]. However, the geometrical formulation is restricted to the case of full-additivity and is not suitable for general linear constraints such as partial additivity (sum less than or equal to one) or bound constraints on the abundances. In [19], a modern convex optimization approach based on the alternating method of multipliers [20] was adopted for solving the constrained optimization problem arising in spectral unmixing.

In this paper, we propose a new flexible spectral unmixing algorithm based on constrained least squares estimation and interior-point optimization [21, 22]. The main originality of our approach is to exploit the potential of primal-dual interior-point techniques, which have shown their efficiency for solving large-scale constrained signal and image processing problems [23, 24]. The proposed optimization method allows to minimize any convex objective function under equality (e.g., sum-to-one) and inequality (e.g., non-negativity or sum-less-than-one) constraints. From the numerical optimization point of view, the choice of a primal-dual interior-point optimization scheme leads to an algorithm that can be implemented effi-

ciently using modern parallel computing tools such as GPUs.

The rest of this paper is organized as follows: Section II formulates the constrained optimization problem arising in spectral unmixing. Section III presents the adopted interior-point optimization scheme for the estimation of the abundance maps. Its implementation issues accounting for memory storage and computing time are discussed in section IV. Finally, Section V illustrates the performances of the proposed approach in terms of computation time and unmixing accuracy, through applications to both synthetic and real data.

II. PROBLEM STATEMENT

Let us consider N pixels of a hyperspectral image acquired in L spectral bands and assume a linear mixing model. This linear model is widely accepted in many practical situations since it offers a first-order approximation of the radiative transfer model [25]. According to this model, the observed spectrum $\boldsymbol{y}_n \in \mathbb{R}^L$ in the n-th pixel is explained as a linear combination of P endmember spectra and corrupted by an additive noise $\boldsymbol{\epsilon}_n$,

$$\boldsymbol{y}_n = \boldsymbol{S}\,\boldsymbol{a}_n + \boldsymbol{\epsilon}_n, \tag{1}$$

where $\boldsymbol{S} = [\boldsymbol{s}_1, \ldots, \boldsymbol{s}_P] \in \mathbb{R}^{L \times P}$ contains the P endmember spectra and $\boldsymbol{a}_n = [a_{n,1}, \ldots, a_{n,P}]^{\mathrm{t}} \in \mathbb{R}^P$ is the vector of endmember abundances in the n-th pixel.

Using matrix notations, the mixing model is rewritten as,

$$\boldsymbol{Y} = \boldsymbol{S}\,\boldsymbol{A} + \boldsymbol{E}, \tag{2}$$

where $\boldsymbol{Y} \in \mathbb{R}^{L \times N}$ is the observation data matrix, $\boldsymbol{A} \in \mathbb{R}^{P \times N}$ the fractional abundance matrix and $\boldsymbol{E} \in \mathbb{R}^{L \times N}$ the measurement noise.

The abundance matrix should satisfy the non-negativity constraint

$$(\forall n \in \{1, \ldots, N\})\,(\forall p \in \{1, \ldots, P\}) \qquad A_{pn} \geqslant 0. \tag{3}$$

This constraint will be denoted as $\boldsymbol{A} \geqslant \boldsymbol{0}$. Moreover, under the assumption that all the endmembers comprising the pixel spectrum in $\boldsymbol{Y}$ are present in the columns of $\boldsymbol{S}$, the abundances coefficients should satisfy the full additivity constraint,

$$(\forall n \in \{1, \ldots, N\}) \qquad \sum_{p=1}^{P} A_{pn} = 1, \tag{4}$$

which can be summarized by $\boldsymbol{1}_P^{\mathrm{t}}\boldsymbol{A} = \boldsymbol{1}_N^{\mathrm{t}}$, where $\boldsymbol{1}_N^{\mathrm{t}}$ denotes a vector of $\mathbb{R}^N$ with all entries equal to one.

When the set of endmembers is incomplete, or when the pixel area are subject ot illumination variability or attenuation, only partial additivity should be required, i.e.:

$$(\forall n \in \{1, \ldots, N\}) \qquad \sum_{p=1}^{P} A_{pn} \leqslant 1, \tag{5}$$

which can be noted shortly by $\boldsymbol{1}_P^{\mathrm{t}}\boldsymbol{A} \leqslant \boldsymbol{1}_N^{\mathrm{t}}$.

The estimation of $\boldsymbol{A}$ given $\boldsymbol{S}$ and $\boldsymbol{Y}$ is firstly formulated as the minimization of a convex criterion $F(.)$, under linear inequality constraints such as non-negativity and partial additivity. Then, the case of the sum-to-one constraint is addressed. Finally, an interior-point algorithm based on a primal-dual approach is proposed for the resolution of the constrained optimization problem.

A. Criterion formulation

The criterion $F(\cdot)$ to minimize results from the statistical modeling of the observation process and the sought abundances properties. Adopting the well-known least-squares approach leads to define $F(\cdot)$ as the quadratic function:

$$F(\boldsymbol{A}) = \frac{1}{2} \sum_{\ell=1}^{L} \sum_{n=1}^{N} ((\boldsymbol{S}\boldsymbol{A})_{\ell n} - Y_{\ell n})^2 . \tag{6}$$

In a statistical estimation framework, (6) corresponds to the neg-log-likelihood associated to a spatially and spectrally uncorrelated Gaussian noise $\boldsymbol{E}$.

Note that the proposed approach can be adapted to a wider class of convex criteria which can be expressed as:

$$F(\boldsymbol{A}) = \frac{1}{2} \sum_{n=1}^{N} (\boldsymbol{S}\boldsymbol{a}_n - \boldsymbol{y}_n)^{\mathrm{t}} \boldsymbol{\Sigma}_n (\boldsymbol{S}\boldsymbol{a}_n - \boldsymbol{y}_n) + \sum_{n=1}^{N} \varphi(\boldsymbol{a}_n), \tag{7}$$

where $\boldsymbol{\Sigma}_n \in \mathbb{R}^{L \times L}$ is a positive spectral covariance matrix and φ is a convex regularization function. However, in the sequel, the presentation will be focused on the case of the least squares criterion (6) since it is widely used in hyperspectral imaging with reflectance spectroscopy.

B. Constraint formulation

We focus on the following general formulation of the constrained optimization problem for the estimation of the abundances maps:

$$\min_{\boldsymbol{A} \in \mathbb{R}^{P \times N}} F(\boldsymbol{A}) \quad \text{s.t.} \quad \boldsymbol{T}_1 \boldsymbol{A} + \boldsymbol{T}_0 \geqslant \boldsymbol{0}, \tag{8}$$

where $\boldsymbol{T}_1 \in \mathbb{R}^{Q \times P}$ and $\boldsymbol{T}_0 \in \mathbb{R}^{Q \times N}$. This formulation allows to take into account

- constraint (3) when $Q = P$, $\boldsymbol{T}_1 = \boldsymbol{I}_P$ and $\boldsymbol{T}_0 = \boldsymbol{0}_P$,
- constraint (5) when $Q = 1$, $\boldsymbol{T}_1 = -\boldsymbol{1}_P^{\mathrm{t}}$ and $\boldsymbol{T}_0 = \boldsymbol{1}_N^{\mathrm{t}}$,
- constraints (3) and (5) jointly, by setting $Q = P + 1$, $\boldsymbol{T}_1 = [\boldsymbol{I}_P \mid -\boldsymbol{1}_P]^{\mathrm{t}}$ and $\boldsymbol{T}_0 = [\boldsymbol{0}^{\mathrm{t}} \mid \boldsymbol{1}_N]^{\mathrm{t}}$,

where $\boldsymbol{I}_N$ denotes the identity matrix of $\mathbb{R}^N$. The equality constraint (4) can be implicitly handled by introducing a reparametrization so that the optimization problem is reduced to an inequality constrained minimization [26].

Property 1. *For each matrix $\boldsymbol{A}^{(0)} \in \mathbb{R}^{P \times N}$ satisfying the equality constraint (4), the transformed vector $\boldsymbol{A} = \boldsymbol{A}^{(0)} + \boldsymbol{Z}\boldsymbol{U}$ also satisfies (4) as soon as the columns of matrix $\boldsymbol{Z} \in \mathbb{R}^{P \times P-1}$ are formed from vectors into the null space of $\boldsymbol{1}_P^{\mathrm{t}}$.*

For the sum-to-one constraint, a null space matrix can be defined by,

$$Z_{ij} = \begin{cases} 1 & \text{if } i = j, \\ -1 & \text{if } i = j + 1, \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

According to this reparametrization, the constrained optimization problem when constraints (3) and (4) are imposed becomes equivalent to

$$\min_{\boldsymbol{U} \in \mathbb{R}^{(P-1) \times N}} F(\boldsymbol{A}^{(0)} + \boldsymbol{Z}\boldsymbol{U}) \quad \text{s.t.} \quad \boldsymbol{T}_1^u \boldsymbol{U} + \boldsymbol{T}_0^u \geqslant \boldsymbol{0}, \tag{10}$$

which takes the general form (8) with $\boldsymbol{T}_1^u = \boldsymbol{T}_1 \boldsymbol{Z}$ and $\boldsymbol{T}_0^u = \boldsymbol{T}_1 \boldsymbol{A}^{(0)} + \boldsymbol{T}_0$.

III. PRIMAL-DUAL OPTIMIZATION FOR ABUNDANCE MAPS ESTIMATION

The main feature of interior-point optimization is to keep the solution inside the strictly feasible domain [21, 22]. At each iteration, the constraint fulfillment is ensured by adding a logarithmic barrier function making the criterion unbounded at the boundary of the feasible solution domain. Let us present our *Interior Point Least Squares* (IPLS) algorithm to solve problem (8). By introducing the operator $m = \text{vec}(M)$ which corresponds to the transformation of a matrix M to a vector m in the lexicographic order, problem (8) is equivalent to the standard form inequality constrained optimization problem:

$$\min_{a \in \mathbb{R}^{PN}} \Phi(a) \quad \text{s.t.} \quad Ta + t \geqslant 0, \quad (11)$$

with $a = \text{vec}(A)$, $T = I_N \otimes T_1$ and $t = \text{vec}(T_0)$, where $\otimes$ is the Kronecker product.

The IPLS algorithm is based on a primal-dual interior-point approach which consists in jointly estimating $a \in \mathbb{R}^{PN}$, and their associated Lagrange multipliers $\lambda \in \mathbb{R}^{QN}$ through the resolution of a sequence of optimization problems obtained from perturbed versions of the Karush-Kuhn-Tucker (KKT) optimality conditions for problem (11):

$$\begin{cases} \nabla\Phi(a) - T^{\text{t}}\lambda &=& 0, \\ \Lambda(Ta + t) &=& \mu_k, \\ Ta + t &\geqslant& 0, \\ \lambda &\geqslant& 0, \end{cases} \quad (12)$$

where $\Lambda = \text{Diag}(\lambda)$ and $\mu_k = \mu_k \mathbf{1}_{QN}$ results from a sequence of perturbation parameters $\{\mu_k\}_{k \in \mathbb{N}}$ converging to 0 as k is growing.

At each iteration k of the algorithm, a_{k+1} and λ_{k+1} are firstly calculated from the perturbed KKT conditions. The perturbation parameter μ_{k+1} is then updated in order to ensure the algorithm convergence. More precisely, an approximate solution of (12) is retained from a Newton algorithm step on the equality conditions, in association with a linesearch strategy allowing to ensure the inequality conditions [22, Chap.11]. The update strategy is then given by

$$(a_{k+1}, \lambda_{k+1}) = (a_k + \alpha_k d_k^a, \lambda_k + \alpha_k d_k^\lambda), \quad (13)$$

where α_k is the step size and (d_k^a, d_k^λ) are the primal and dual Newton directions.

Based on the iterative scheme (13), several primal-dual interior point methods have been proposed in the literature, each of them calling for its own strategy for the computation of the primal-dual directions, the derivation of a suitable step size, and the update of the perturbation parameter (See [27, 28] for a review). The proposed IPLS algorithm for spectral unmixing relies on the iterative scheme of [26] into which additional tools, that are described in the following, have been included to accelerate the practical convergence, as well as to reduce the computational cost per iteration.

1) Primal-dual directions: The Newton directions (d_k^a, d_k^λ) are obtained by solving the linear system,

$$\begin{bmatrix} \nabla^2\Phi(a_k) & -T^{\text{t}} \\ \Lambda_k T & \text{Diag}(Ta_k + t) \end{bmatrix} \begin{bmatrix} d_k^a \\ d_k^\lambda \end{bmatrix} = -r_{\mu_k}(a_k, \lambda_k), \quad (14)$$

where $\nabla\Phi(\cdot)$ and $\nabla^2\Phi(\cdot)$ are, respectively, the gradient and the Hessian of criterion $\Phi(\cdot)$, and $r_\mu(a, \lambda)$ is the primal-dual residual defined by,

$$r_\mu(a, \lambda) = \begin{pmatrix} \nabla\Phi(a) - T^{\text{t}}\lambda \\ \Lambda(Ta + t) - \mu \end{pmatrix} = \begin{pmatrix} r_\mu^{\text{prim}}(a, \lambda) \\ r_\mu^{\text{dual}}(a, \lambda) \end{pmatrix}. \quad (15)$$

As pointed out in [29, 30], the primal-dual matrix in the left side of equation (14) suffers from ill-conditioning as soon as $(Ta_k + t)_i \ll 1$ or $\lambda_i \ll 1$. Moreover, this matrix is not guarenteed to be symmetric or definite positive [28], so that the linear system (14) is difficult to solve. Therefore, rather than solving directly (14), [26, 31] propose to proceed by variable substitution. From the second equation of (14) one deduces,

$$d_k^\lambda = \text{Diag}(Ta_k + t)^{-1}[\mu_k - \Lambda_k(Ta_k + t) - \Lambda_k Td_k^a]. \quad (16)$$

Then, the primal direction d_k^a is obtained by solving the reduced linear system

$$\begin{aligned} [\nabla^2\Phi(a_k) + T^{\text{t}}\text{Diag}(Ta_k + t)^{-1}\Lambda_k T]\, d_k^a = \\ -\nabla\Phi(a_k) + T^{\text{t}}\text{Diag}(Ta_k + t)^{-1}\mu_k. \end{aligned} \quad (17)$$

Finally, the dual direction d_k^λ is calculated according to (16). Note that our computation of the primal direction differs from [26]. Indeed, instead of a low rank approximation of $\nabla^2\Phi(a_k)$, we keep the true Hessian matrix in (17), with the will to accelerate the convergence of our algorithm (see remark 1 at the end of this section).

2) Linesearch: The step size value α_k should be chosen so as to ensure the convergence of the IPLS algorithm and the fulfillment of the inequalities of the pertubed KKT system (12). The convergence study of the primal-dual algorithm presented in [26] requests that α_k ensures a sufficient decrease of the primal-dual merit function $\Psi_{\mu_k}(a, \lambda)$,

$$\begin{aligned} \Psi_{\mu_k}(a, \lambda) = \Phi(a) - \mu \sum_{i=1}^{QN} \ln([Ta + t]_i) \\ + \lambda^{\text{t}}(Ta + t) - \mu \sum_{i=1}^{QN} \ln(\lambda_i[Ta + t]_i). \end{aligned} \quad (18)$$

One can note that (18) contains two logarithmic barrier functions enforcing the fulfillment of the KKT inequalities. The sufficient decrease is assessed using the Armijo condition,

$$\psi_{\mu_k}(\alpha_k) - \psi_{\mu_k}(0) \leqslant \sigma \alpha_k \nabla\psi_{\mu_k}(0) \quad \text{with} \quad \sigma \in (0, 1/2), \quad (19)$$

where $\psi_{\mu_k}(\alpha) \triangleq \Psi_{\mu_k}(a_k + \alpha d_k^a, \lambda_k + \alpha d_k^\lambda)$. A step size α_k satisfying (19) is obtained by a backtracking algorithm [22]: starting from an initial step size α_k^0, and if the latter does not satisfy (19), smaller values are tested, $\alpha_k^0 \tau$, $\alpha_k^0 \tau^2$, ..., $\tau \in (0, 1)$, until (19) holds.
In order to ensure that $\psi_{\mu_k}(\cdot)$ remains finite valued, the backtracking strategy is initialized as follows,

$$\begin{cases} \alpha_k^0 = 1 & \text{if} \quad \alpha_k^+ = +\infty, \\ \alpha_k^0 = \min(1, 0.99\,\alpha_k^+) & \text{elsewhere}, \end{cases} \quad (20)$$

where α_k^+ is the largest positive value such that,

$$\lambda_k + \alpha d_k^\lambda > 0, \qquad T(a_k + \alpha d_k^a) + t > 0. \quad (21)$$

3) Perturbation parameter update: According to [26], the convergence is ensured as soon as the sequence $\{\mu_k\}_{k\in\mathbb{N}}$ tends to 0 when k tends to infinity. We propose to update the parameter μ_k by using the μ-criticity rule defined in [32] by:

$$\mu_k = \theta \frac{\delta_k}{QN}, \qquad (22)$$

where $\delta_k = (\boldsymbol{T}\boldsymbol{a}_k + \boldsymbol{t})^{\mathrm{t}}\boldsymbol{\lambda}_k$ is the duality gap and $\theta \in (0,1)$.

4) Stopping criteria: The main steps of the proposed optimization method are summarized in Algorithm 1. Following [23, 31], the accuracy of the primal and dual directions (inner loop) is controlled by:

$$\|\boldsymbol{r}_{\mu_k}^{\mathrm{prim}}\|_\infty \leqslant \epsilon_k^{\mathrm{prim}} \text{ and } \delta_k/QN \leqslant \epsilon_k^{\mathrm{dual}}, \qquad (23)$$

where $\boldsymbol{r}_{\mu_k}^{\mathrm{prim}}$ is the primal residual at $\boldsymbol{a}_k$, $\epsilon_k^{\mathrm{prim}} = \eta^{\mathrm{prim}}\mu_k$, $\epsilon_k^{\mathrm{dual}} = \eta^{\mathrm{dual}}\mu_k$ with $\eta^{\mathrm{prim}} > 0$ and $\eta^{\mathrm{dual}} \in (1, \theta^{-1})$. The outer iterations of Algorithm 1 are run until the fulfillment of the stopping condition proposed in [22, Chap.11]:

$$\mu_k \leqslant \mu_{\min} \quad \text{or} \quad \left(\|\boldsymbol{r}_0^{\mathrm{prim}}\| + \|\boldsymbol{r}_0^{\mathrm{dual}}\|\right) \leqslant \epsilon_0. \qquad (24)$$

Require: Initial values $\boldsymbol{\lambda}_0 > \boldsymbol{0}$ and $\boldsymbol{a}_0$ such that $\boldsymbol{T}\boldsymbol{a}_0 + \boldsymbol{t} > \boldsymbol{0}$
Ensure: Resolution of (11)
While (condition (24) is not satisfied) **do**
    While (condition (23) is not satisfied) **do**
        Calculate $\boldsymbol{d}_k^a$ by solving (17)
        Deduce $\boldsymbol{d}_k^\lambda$ from (16)
        Find $\alpha_k > 0$ satisfying (19)
        Update $(\boldsymbol{a}_{k+1}, \boldsymbol{\lambda}_{k+1})$ according to (13)
    done
    Define μ_{k+1} according to (22).
done

Algorithm 1: Interior Point Least Squares algorithm.

5) Convergence result: The convergence of Algorithm 1 is guaranteed by the following result.

Theorem III.1. *Let $\Phi(\cdot)$ a twice differentiable convex function on $\mathbb{R}^{PN}$. Assume that the set $\mathcal{S} = \left\{\boldsymbol{a} \in \mathbb{R}^{PN} \mid \boldsymbol{T}\boldsymbol{a} + \boldsymbol{t} > \boldsymbol{0}\right\}$ is nonempty and bounded, and that either $\Phi(\cdot)$ is strictly convex or $\boldsymbol{T}^{\mathrm{t}}\boldsymbol{T}$ is invertible. Then, for every fixed $\mu > 0$, there exists k_μ such that the sequence $\{(\boldsymbol{a}_k, \boldsymbol{\lambda}_k)\}_{k \geqslant k_\mu}$ generated by (13) converges q-superlinearly to the unique minimizer of Ψ_μ. Moreover, the outer loop of Algorithm 1 generates a bounded sequence $\{(\boldsymbol{a}_k, \boldsymbol{\lambda}_k)\}$ whose accumulation points are primal-dual solutions of problem (11). Finally, if $\Phi(\cdot)$ is strictly convex, the outer iterates $\{\boldsymbol{a}_k\}$ converge to the unique solution of (11).*

    Proof: See Appendix A. ∎

We now comment the differences between Theorem III.1 and the convergence results in [26].

Remark 1.

(i) *The convergence result of [26] is established under the assumption that the criterion and the constraints are convex, and at least one of them is strongly convex. In our study, the convexity of Φ is sufficient, under the additional assumption that the constraints are linearly independant (i.e., $\boldsymbol{T}^{\mathrm{t}}\boldsymbol{T}$ inversible) and that the set $\mathcal{S}$ is nonempty and bounded. Note that these assumptions hold in particular for the constraints (3)-(4) or (3)-(5).*

(ii) *The q-superlinear convergence rate of the inner loop of our algorithm is mainly due to the use of the exact Hessian matrix in the primal system (17). A weaker result in terms of convergence speed is obtained in [26], since a quasi-Newton approximation of the Hessian matrix is considered.*

IV. MEMORY REQUIREMENT AND COMPUTATION TIME REDUCTION FOR LARGE SCALE SPECTRAL UNMIXING

According to the expression of the least squares criterion (6), the constrained optimization problem (8) is separable with respect to the image pixels. A first implementation strategy, denoted hereafter by *pixel-based* strategy, is to solve problem (2) by applying Algorithm 1 for unmixing each pixel n individually. A second approach is to adopt an *image-based* strategy, that is, solving the whole problem (8) with the primal-dual algorithm. A discussion on the numerical efficiency of both strategies will be given in Section V-A.

When the *image-based* strategy is adopted, the numerical complexity of Algorithm 1 is highly dominated by the primal direction calculation through the resolution of the linear system (17). This section presents an analysis of the structure of this system with the aim to reduce the computational cost and the memory requirement of Algorithm 1.

A. Primal system structure

The linear system (17) can be expressed as

$$\boldsymbol{H}_k \boldsymbol{d}_k^a = -\boldsymbol{g}_k, \qquad (25)$$

where

$$\boldsymbol{H}_k = \nabla^2\Phi(\boldsymbol{a}_k) + \boldsymbol{T}^{\mathrm{t}}\mathrm{Diag}(\boldsymbol{T}\boldsymbol{a}_k + \boldsymbol{t})^{-1}\boldsymbol{\Lambda}_k\boldsymbol{T}, \qquad (26)$$

$$\boldsymbol{g}_k = \nabla\Phi(\boldsymbol{a}_k) - \boldsymbol{T}^{\mathrm{t}}\mathrm{Diag}(\boldsymbol{T}\boldsymbol{a}_k + \boldsymbol{t})^{-1}\boldsymbol{\mu}_k. \qquad (27)$$

An analysis of the structure of matrix $\boldsymbol{H}_k$ is necessary in order to find an appropriate implementation strategy.

Firstly, we recall that $\boldsymbol{T} = \boldsymbol{I}_N \otimes \boldsymbol{T}_1$. Thus, $\boldsymbol{T}$ is block-diagonal composed by N identical blocks equal to $\boldsymbol{T}_1$. The notation $\boldsymbol{T} = \mathrm{Bdiag}_N(\boldsymbol{T}_1)$ is used in the sequel. For every $n \in \{1, \cdots, N\}$, let $\boldsymbol{a}_{n,k} \in \mathbb{R}^P$ (resp. $\boldsymbol{\lambda}_{n,k} \in \mathbb{R}^Q$) be the n-th column of $\boldsymbol{A}_k = \mathrm{mat}(\boldsymbol{a}_k)^{P \times N}$ (resp. of $\Lambda_k = \mathrm{mat}(\boldsymbol{\lambda}_k) \in \mathbb{R}^{Q \times N}$) where $\mathrm{mat}(.)$ is the reciprocal operator of $\mathrm{vec}(.)$. Moreover, let $\boldsymbol{t}_{0,n} \in \mathbb{R}^Q$ be the n-th column of $\boldsymbol{T}_0$. It follows that,

$$\mathrm{Diag}(\boldsymbol{T}\boldsymbol{a}_k + \boldsymbol{t})^{-1}\boldsymbol{\Lambda}_k = \mathrm{Bdiag}_N(\boldsymbol{D}_{n,k}), \qquad (28)$$

where $\boldsymbol{D}_{n,k}$ is a diagonal matrix of size $P \times P$ whose diagonal elements are $\mathrm{Diag}\left(\boldsymbol{T}_1 \boldsymbol{a}_{n,k} + \boldsymbol{t}_{0,n}\right)^{-1} \boldsymbol{\lambda}_{n,k}$. Therefore,

$$
\begin{aligned}
\boldsymbol{T}^{\mathrm{t}} \mathrm{Diag}(\boldsymbol{T}\boldsymbol{a}_k + \boldsymbol{t})^{-1} \boldsymbol{\Lambda}_k \boldsymbol{T} = & \\
\mathrm{Bdiag}_N(\boldsymbol{T}_1)^{\mathrm{t}} \mathrm{Bdiag}_N(\boldsymbol{D}_{n,k}) & \mathrm{Bdiag}_N(\boldsymbol{T}_1), \\
= \mathrm{Bdiag}_N(\boldsymbol{T}_1^{\mathrm{t}} \boldsymbol{D}_{n,k} \boldsymbol{T}_1). & \quad (29)
\end{aligned}
$$

Secondly, the Hessian $\nabla^2 \Phi(\boldsymbol{a}_k)$ of the least squares criterion reads

$$
\begin{aligned}
\nabla^2 \Phi(\boldsymbol{a}_k) &= (\boldsymbol{I}_N \otimes \boldsymbol{S})^{\mathrm{t}} (\boldsymbol{I}_N \otimes \boldsymbol{S}), \\
&= \mathrm{Bdiag}_N(\boldsymbol{S}^{\mathrm{t}} \boldsymbol{S}), \quad (30)
\end{aligned}
$$

where (30) is a consequence of Kronecker product properties [33]:

$$
\begin{cases}
(\boldsymbol{A} \otimes \boldsymbol{B})^{\mathrm{t}} = \boldsymbol{A}^{\mathrm{t}} \otimes \boldsymbol{B}^{\mathrm{t}}, \\
(\boldsymbol{A} \otimes \boldsymbol{B})(\boldsymbol{C} \otimes \boldsymbol{D}) = \boldsymbol{A}\boldsymbol{C} \otimes \boldsymbol{B}\boldsymbol{D}.
\end{cases} \quad (31)
$$

Finally, (29) and (30) yield,

$$
\boldsymbol{H}_k = \mathrm{Bdiag}_N(\boldsymbol{S}^{\mathrm{t}} \boldsymbol{S} + \boldsymbol{T}_1^{\mathrm{t}} \boldsymbol{D}_{n,k} \boldsymbol{T}_1). \quad (32)
$$

Consequently, $\boldsymbol{H}_k$ is a block-diagonal matrix formed by N blocks of size $P \times P$. Note that, in the case of problem (10), a similar analysis leads to

$$
\boldsymbol{H}_k = \mathrm{Bdiag}_N(\boldsymbol{Z}^{\mathrm{t}}(\boldsymbol{S}^{\mathrm{t}} \boldsymbol{S} + \boldsymbol{T}_1^{\mathrm{t}} \boldsymbol{D}_{n,k} \boldsymbol{T}_1)\boldsymbol{Z}), \quad (33)
$$

that is, $\boldsymbol{H}_k$ block-diagonal with N blocks of size $(P-1) \times (P-1)$.

B. Memory issues

When applying the *image-based* strategy to large scale problems, the memory space required to store matrix $\boldsymbol{H}_k$ can exceed the available memory, even when using a sparse coding. A less memory demanding calculation of the primal direction can be achieved by solving separately, for each iteration, the N lower-size linear systems

$$
(\forall n \in \{1, \ldots, N\}) \qquad \boldsymbol{H}_{n,k} \boldsymbol{d}_{n,k}^a = -\boldsymbol{g}_{n,k}, \quad (34)
$$

where $\boldsymbol{d}_{n,k}^a$ (resp. $\boldsymbol{g}_{n,k}$) is the n-th column of the matrix $\mathrm{mat}(\boldsymbol{d}_k^a)$ (resp. $\mathrm{mat}(\boldsymbol{g}_k)$). This implementation will now be referred to as the *image-based pixel-wise* implementation, as opposed to the *image-based full-wise* implementation where $\boldsymbol{H}_k$ is entirely built.

The pixel-wise strategy being based on the resolution at each iteration of N independent linear systems, it is straightforward to implement in parallel. An intermediate implementation dividing system (32) (or (33)) into $1 \leqslant K \leqslant N$ blocks can also be considered, with the advantage to adapt the normal equations size K to the available memory. This latter approach will be referred to as *image-based block-wise* implementation. The performance of each implementation strategy will be discussed in Section V-A.

V. EXPERIMENTAL RESULTS

This section discusses the performances and illustrates the applicability of Algorithm 1. The latter is referred to as IPLS, the type of constraints being indicated in prefix, namely NN for non-negativity constraint (3), STO for sum-to-one and non-negativity constraints (3)-(4) and SLO for sum-lower-than-one and non-negativity constraints (3)-(5).

We first consider synthetic data in order to discuss the choice of implementation strategy, to perform a comparative analysis with existing unmixing methods, and to illustrate the relevance of the partial additivity constraints. Then, the parallel implementation of IPLS using GPUs is adressed. Finally, its applicability is emphasized through the processing of real hyperspectral data.

The computation of the proposed primal-dual algorithm requires specifying the parameters $(\eta^{\mathrm{prim}}, \eta^{\mathrm{dual}}, \theta)$ and $(\mu_{\min}, \epsilon_0)$, controlling the precision of the inner and outer loops, respectively. Following [23, 31], we set:

$$
\eta^{\mathrm{prim}} = 100, \quad \eta^{\mathrm{dual}} = 1.9, \quad \theta = 0.5. \quad (35)
$$

Moreover, the values $\mu_{\min} = 10^{-9}$ and $\epsilon_0 = 10^{-7}$ are retained for the stopping condition (24).

A. Synthetic data

In order to simulate realistic synthetic hyperspectral data, reflectance spectra from the USGS (U.S. Geological Survey) spectral library [34] are retained[1]. These reflectance spectra contain $L = 224$ spectral bands from 383 nm to 2508 nm. A subset of P spectra is then randomly picked up to create synthetic mixtures with abundances simulated from a Dirichlet distribution. Only realizations with maximum abundance value lower than a specified level $A_{\max}$ are retained. Finally, a random Gaussian noise is added to each resulting mixture spectrum, in order to get a signal to noise ratio (SNR) of 30 dB. The unmixing algorithms are implemented on Matlab 2012b and the calculations are performed using a HP Compaq Elite desktop having an Intel Core i7 3.4 GHz processor and 8 GB of RAM.

The first step of the experiment consists in choosing the best implementation strategy adapted to this hardware and software configuration. Then, some comparisons are performed between our method and NNLS, FCLS and ADMM algorithms, in terms of computation time and estimation accuracy. Finally, we illustrate through two examples, the relevance of the partial additivity constraint over the full additivity and the non-negativity constraints.

1) Pixel-based or image-based unmixing: In order to compare the performances of pixel-based and image-based full-wise implementations, we consider the unmixing of synthetic images of size $N = 64^2$, with different number of end-members, and $A_{\max} = 1$. The unmixing is performed under constraints (3)-(4), using either the exact endmembers or those extracted from the image using the N-FINDR method [35]. For each test realized, Table I reports the computation time

---

[1]available at http://pubs.usgs.gov/of/2003/ofr-03-395/datatable.html

per pixel, the average number of iterations (outer iterations of Algorithm 1), and the average residual norm:

$$r = \frac{1}{N} \sum_{n=1}^{N} \frac{1}{L} \|\boldsymbol{y}_n - \boldsymbol{S}\boldsymbol{a}_n\|_2 . \quad (36)$$

It can be noted that the image-based implementation is significantly faster than the pixel-based. This is explained by the better management of the vectorized calculations compared to sequential ones in Matlab. Moreover, less iterations are required to reach the stopping criterion in the case of the image-based implementation. The average residual norm of the two strategies certifies that the quality of the reconstruction is equivalent in both cases. Finally, let us emphasize that the performances of IPLS are not degraded when replacing the exact endmembers by their estimation with N-FINDR.

Endmembers	P	Time (μs)		Iterations		r ($\times 10^{-4}$)	
		PXL	IMG	PXL	IMG	PXL	IMG
LIB	3	2150	15	25.7	20.3	3.14	3.14
	6	2198	49	25.2	20.4	3.38	3.38
	10	2307	118	24.6	20.9	3.65	3.65
	15	2649	266	24.0	21.1	3.75	3.76
EEA	3	2138	15	25.7	20.2	3.15	3.14
	6	2192	48	25.2	20.2	3.38	3.39
	10	2302	117	24.5	20.9	3.69	3.70
	15	2643	277	23.9	21.2	3.86	3.86

TABLE I
AVERAGE TIME PER PIXEL, NUMBER OF ITERATIONS, AND RESIDUAL NORM OVER 100 MONTE-CARLO SIMULATIONS, FOR DIFFERENT NUMBER OF ENDMEMBERS USING ACTUAL (LIB) OR ESTIMATED ENDMEMBERS (EEA): COMPARISON BETWEEN PIXEL-BASED (PXL) AND IMAGE-BASED (IMG) IMPLEMENTATIONS OF IPLS.

2) Image-based unmixing alternatives: We now analyse the influence of the block-size parameter K on the performance of the image-based block-wise implementation. In that respect, we consider a hyperspectral image of size $N = 128^2$ built using a subset of P endmembers from the USGS library, and $A_{\max} = 1$. The computation time required by Algorithm 1 to unmix this image under full additivity constraints (3)-(4), is presented in Figure 1 for several numbers of endmembers and different values of K. Note that a block size $K = 1$ corresponds to the pixel-wise implementation and the full-size strategy is obtained by setting $K = N$. The other configurations correspond to intermediate block-wise alternatives. The memory space required for the unmixing function in Matlab is also reported.

Ideally, the best implementation should correspond to both a low computing time and a low memory usage. According to our results, the computation time decreases as the block size rises, reaching a minimum for a block size above 100, whatever the value of P. On the other hand, as expected, the memory usage grows with the block size. Consequently, the block size should be set to intermediate value in order to achieve a best computing time. The block-wise implementation with a block size $K = 256$ allowing to get the best compromise between computing time and memory requirement is retained for the remaining experiments presented in the paper.
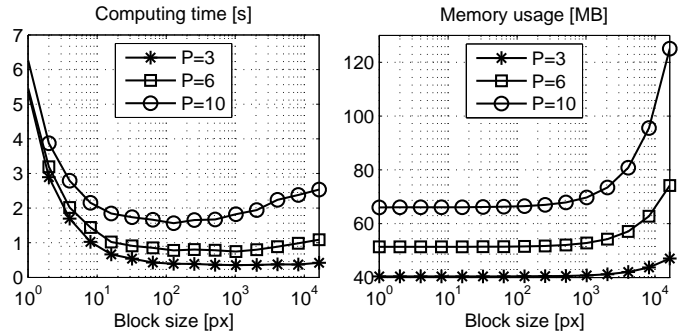

Fig. 1. Unmixing computation time (in seconds) and memory usage (MB) for different block sizes of the block-wise implementation

3) Comparison with state-of-the-art unmixing algorithms: IPLS is now compared with FCLS when both non-negativity constraint (3) and full additivity constraint (4) are imposed, and with NNLS when only non-negativity is considered. We also compare our algorithm with the alternating method of multipliers (ADMM) from [19], using the Matlab code available at http://www.lx.it.pt/~bioucas. Synthetic hyperspectral images of size $N = 64^2$ are generated, using different number P of endmembers and $A_{\max} = 1$. For each image, the set of spectra employed to perform the spectral unmixing is either the one used to create the image or is estimated using the N-FINDR endmember extraction algorithm.

The three methods have led to the same unmixing quality in terms of residual value r. The results in terms of average computation time per pixel over 100 Monte-Carlo simulations are reported in Table II. For all the tests realized, both STO-IPLS and STO-ADMM appear to be faster than FCLS. The ratio between STO-IPLS or STO-ADMM and FCLS computation times seems to be independent from the number of endmember used. NN-IPLS and NN-ADMM are also faster than NNLS under the conditions tested. This superiority tends to decrease as the number of endmembers increases. Finally, the ranking between IPLS and ADMM methods depends on the experimental conditions. According to our tests, STO-IPLS seems slightly faster than STO-ADMM, while NN-IPLS and NN-ADMM perform similarly in terms of computation time.

4) Relevance of the partial additivity constraint: When dealing with real data, the abundance estimation performances depend on the used endmember spectra and on the constraints that are imposed on the abundance values. The aim of this section is to show that it may be suitable, in some situations, to relax the sum-to-one constraint (4) and, eventually, to replace it by the partial additivity constraint (5). A hyperspectral image of size $N = 128^2$ built using a subset of $P = 6$ endmembers from the USGS library is considered. The accuracy of the abundance maps estimation is assessed using the normalized mean square error

$$\text{NMSE}(\%) = \frac{100}{P} \sum_{p=1}^{P} \left(\|\boldsymbol{m}_p - \hat{\boldsymbol{m}}_p\|^2 / \|\boldsymbol{m}_p\|^2 \right), \quad (37)$$

which measures the relative mean difference between the actual abundances maps $\boldsymbol{m}_p = [A_{p1}, \ldots, A_{pN}]^{\mathrm{t}}$ and the

Endmembers	P	Time (μs)		
		FCLS	STO-ADMM	STO-IPLS
LIB	3	46	22	**18**
	6	84	65	**45**
	10	210	124	**90**
	15	479	198	**177**
EEA	3	45	**18**	**18**
	6	84	71	**42**
	10	144	132	**89**
	15	314	197	**179**

Endmembers	P	Time (μs)		
		NNLS	NN-ADMM	NN-IPLS
LIB	3	66	**18**	20
	6	117	53	**46**
	10	177	109	**94**
	15	246	**183**	190
EEA	3	67	**15**	20
	6	118	54	**45**
	10	175	121	**93**
	15	237	189	**187**

TABLE II

AVERAGE TIME PER PIXEL FOR DIFFERENT NUMBER OF ENDMEMBERS USING ACTUAL (LIB) OR ESTIMATED ENDMEMBERS (EEA): COMPARISON BETWEEB FCLS, ADMM AND IPLS FULLY CONSTRAINED (STO) AND NNLS, ADMM AND IPLS WITH NON-NEGATIVITY CONSTRAINT ONLY (POS).

Endmembers	$A_{\max}$	μ_η	Constraints		
			STO	SLO	NN
LIB	(1, 0.95, 0.9)	1.00	**0.04**	0.10	0.19
		0.95	3.83	**0.64**	0.69
		0.90	12.36	**2.03**	2.06
EEA	1	1.00	**0.47**	0.58	0.72
		0.95	4.57	3.39	**1.44**
		0.90	16.45	12.71	**5.08**
	0.95	1.00	**0.63**	0.74	**0.90**
		0.95	5.65	4.47	**1.94**
		0.90	17.18	13.28	**5.02**
	0.9	1.00	**0.99**	**1.12**	1.24
		0.95	4.67	3.45	**1.42**
		0.90	17.25	13.25	**5.72**

TABLE III

ACCURACY (NMSE) OF ABUNDANCE ESTIMATION FROM EITHER ACTUAL (LIB) OR EXTRACTED ENDMEMBERS (EEA), USING IPLS UNDER FULL ADDITIVITY (STO), PARTIAL ADDITIVITY (SLO) AND NON-NEGATIVITY (NN) CONSTRAINTS.

the VCA algorithm. The unmixing is performed under the full additivity (3)-(4), partial additivity (3)-(5) or non-negativity (3) constraints. From Table IV, one can note that imposing the partial additivity constraint is very useful when the number of unmixed endmembers is lower than the number of actual endmembers.

P	$\hat{P}$	LIB			EEA		
		STO	SLO	NN	STO	SLO	NN
6	6	**0.06**	0.11	0.17	**0.47**	0.51	0.60
	5	18.22	**15.39**	19.34	18.28	**15.47**	19.66
	4	48.31	**36.00**	39.15	48.17	**35.99**	39.19
10	10	**0.14**	0.21	0.31	**1.68**	1.84	2.15
	9	10.13	**7.41**	8.69	11.64	**9.05**	10.17
	8	22.47	**16.40**	22.41	23.66	**17.98**	23.50
15	15	**0.52**	0.73	1.57	**6.43**	6.64	9.23
	14	6.17	**5.65**	16.19	15.79	**12.31**	19.20
	13	12.32	**10.97**	26.75	21.56	**16.77**	27.96

TABLE IV

ACCURACY (NMSE IN (%)) OF ABUNDANCE ESTIMATION USING IPLS UNDER FULL ADDITIVITY (STO), PARTIAL ADDITIVITY (SLO) AND NON-NEGATIVITY (NN) CONSTRAINTS: EFFECT OF AN INCOMPLETE SET OF ACTUAL (LIB) OR EXTRACTED ENDMEMBERS (EEA).

estimated ones $\hat{m}_p$.

a) Effect of illumination variability: We first analyse the relevance of the full additivity constraint when the image pixels are subject to illumination variability. In that respect, each pixel spectrum generated from the linear mixing model is multiplied by a scale factor η modeling the illumination variability due to surface topography or atmospheric attenuation [36]. As in [37], this scale factor is simulated from a Beta distribution with a specified mean value μ_η in the interval [0.9, 1]. The hyperspectral image is then unmixed using the IPLS algorithm on the exact endmembers, with either constraint (4) or (5) in addition to the non-negativity constraint (3).

Table III summarizes our results for different maximum abundance values and attenuation levels. The number of pixels was set to $N = 50^2$ and 100 Monte-Carlo simulations have been considered. It can be noted that the full additivity constraint leads to the best estimation results in the absence of illumination variability ($\mu_\eta = 1$) and endmember spectra taken either from the library or extracted from the image using an endmember extraction algorithm (VCA in this experiment). However, the performances decrease when the value of μ_η equals 0.95 or 0.9. It can be, for instance, noted that the partial additivity is relevant when the endmembers are taken from the library and that the non-negativity constraint alone leads to the best results when endmembers are extracted from the image.

b) Effect of an incomplete set of endmembers: We now consider the case when the set of endmembers used to unmix the image spectra is incomplete. Such situation arises, for instance, when the number of components is underestimated. An image of size $N = 50^2$ containing P endmembers is simulated using the same strategy as in the previous experiment, using $A_{\max} = 1$ and $\mu_\eta = 1$. The unmixing is performed using a subset of $\hat{P} \leqslant P$ endmembers arbitrarily taken from the actual set of endmembers or estimated using

B. Parallel implementation

A parallel implementation of Algorithm 1, for both image-based and pixel-based strategies, has been realized using CUDA (for *Compute Unified Device Architecture*), a programming model created by Nvidia based on a language designed as an extension of the C language.

In the pixel-based implementation, the entire algorithm is run independently for each pixel. One thread per pixel is used, each thread containing the whole IPLS algorithm. The image-based implementation does not present such a degree of parallelization since some steps, namely the linesearch, the perturbation parameter update, and the convergence check, require the computation of one variable for the entire image. These global steps, called *reductions*, are optimized using the combination of the GPU and the CPU as it is described in [38, Chap. 6].
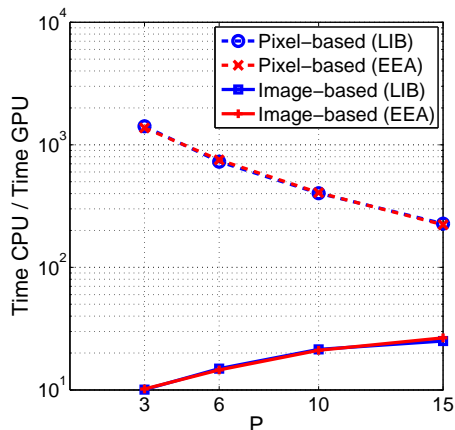
Fig. 2.   Comparison in terms of computation time per pixel, between GPU and CPU implementation for pixel-based and image-based implementations of IPLS algorithm. Average results over 100 Monte-Carlo simulations, for different number of endmembers using actual (LIB) or estimated endmembers (EEA).
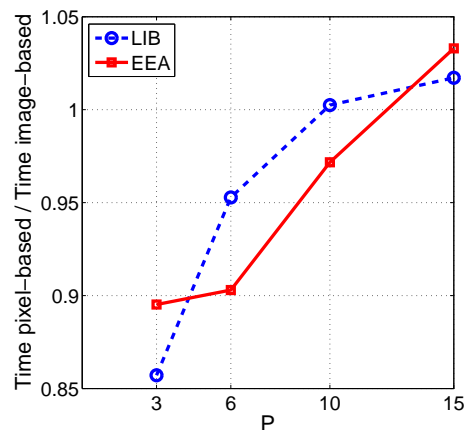

Fig. 3.   Comparison in terms of computation time per pixel, between pixel-based and image-based GPU implementations of IPLS algorithm. Average results over 100 Monte-Carlo simulations, for different number of endmembers using actual (LIB) or estimated endmembers (EEA).

For the same experiments than those conducted in Section V-A1, we present in Figure 2 the speed up in terms of average computation time per pixel obtained when using parallel programming. The IPLS algorithm was run on a Dell Precision T7400 having an Intel Xeon X5472 3 GHz processor and 16 GB of RAM. It embeds the Nvidia Tesla C1060 GPU (Graphics Processing Unit) allowing to do parallel computation on its 240 processor cores running at 1.3 GHz. Note that the iterations number and the residual norms resulting from these tests were the same than those presented in Table I, which shows the validity of our GPU program. The ratio between GPU and CPU computation time follows different behaviour for pixel-based and image-based implementations. When the first approach is retained, the gain of GPU computing tends to decrease, as the number of endmembers grows. On the opposite, the image-based GPU implementation tends to be more efficient when P increases. Up to our knowledge, this difference could be due to the use of the strategy by [38, Chap. 6] in our GPU programs of the image-based implementation. Indeed, it implies that the computing time necessary for performing data transfers remains constant, whatever the value of P. For small size unmixing problems, this transfer time becomes preponderant over other operations, thus limiting the GPU speed-up.

Figure 3 illustrates the ratio between the average computation time per pixel for pixel-based and image-based GPU implementations. Although the pixel-based approach would seem better suited for parallelization, it presents similar computation time than the image-based approach, when GPU programming is employed. This can be explained by the fact that, according to the CUDA model [39], the threads are actually processed by groups of 32 called *warps*. Each warp works as an SIMD (Single Instruction Multiple Data) unit. At one given instant, all the threads of one warp are necessarily executing the same instruction. In the case of a conditional structure such as *if, then, else*, if two conditions are satisfied by different threads within a warp, then two series of instructions corresponding to

these conditions are executed by all the threads of this warp, although some of the results are ignored. Therefore, during the execution of the pixel-based IPLS, the computing time depends on the pixels having the slowest convergence rate in each group of 32 consecutive pixels. The gain when using the pixel-based approach can thus be small if the convergence rates highly differ from one pixel to another. Another reason is that, as emphasized in Section V-A1, the IPLS algorithm requires more iterations to reach convergence in its pixel-based version.

C. Real data processing

We consider in this section the unmixing of the well known AVIRIS Cuprite dataset available online[2]. This image originally contains 250×191 pixels and 224 spectral bands between 0.4 and $2.5 \mu m$. Only 188 bands are preserved after removing the corrupted ones.

1) Number of endmembers estimation: The number of endmembers is estimated with the SGDE method proposed in [40] based on Gerschgorin disks' radii is used, leading to the reasonable number of 14 endmembers. This estimated number is used during the rest of the experiment. Other methods such as Virtual Dimentionality estimation [41] or ELM [42] could have been used, possibly leading to a different number.

2) Endmembers extraction: The endmembers are extracted from the scene using the N-FINDR algorithm. For each endmember, Table V gives the two closest components of the USGS library according to the Spectral Information Divergence (SID) [43]. Other endmember extraction algorithms and spectral distance measurements could have been used, possibly leading to different substances. A survey on EEA algorithms is conducted in [44].

3) Abundance estimation: Computing times for different constraints are reported in Table VI, for an image-based block-wise implementation of the IPLS algorithm, run on Matlab R2011b codes, using the same architecture than in Section

---

[2]http://aviris.jpl.nasa.gov/html/aviris.freedata.html.

Index	Mineral	SID ($\times 10^{-3}$)
1	Pyrope WS474	6.52
	Sphene HS189.3B	8.36
2	Buddingtonite GDS85 D-206	6.51
	Kaolin/Smect KLF511 12%K	7.39
3	Nontronite SWa-1.a	13.31
	Kaolin/Smect H89-FR-5 30K	13.73
4	Nontronite NG-1.a	6.33
	Montmorillonite+Illi CM37	8.63
5	Nontronite SWa-1.a	15.39
	Kaolin/Smect KLF508 85%K	17.51
6	Rectorite ISR202 (RAr-1)	8.46
	Montmorillonite+Illi CM42	8.49
7	Montmorillonite+Illi CM42	9.52
	Kaolin/Smect H89-FR-5 30K	10.36
8	Kaolin/Smect KLF511 12%K	3.28
	Rectorite ISR202 (RAr-1)	4.08
9	Montmorillonite CM20	5.38
	Alunite GDS82 Na82	6.91
10	Thenardite GDS146	3.60
	Kaolin/Smect H89-FR-2 50K	3.35
11	Cookeite CAr-1.c ¡30um	2.54
	Thenardite GDS146	2.65
12	Montmorillonite+Illi CM42	8.22
	Rectorite ISR202 (RAr-1)	9.37
13	Kaolin/Smect KLF511 12%K	1.82
	Montmorillonite+Illi CM37	3.21
14	Barite HS79.3B	5.36
	Richterite HS336.3B	5.51

TABLE V
SPECTRAL INFORMATION DIVERGENCE (SID) BETWEEN EXTRACTED ENDMEMBERS AND LABORATORY REFLECTANCES.

NN   SLO   STO

Fig. 4. Cuprite residual norm per pixel after unmixing with IPLS subject to different constraints.

NN   SLO

Fig. 5. Abundance sum per pixel after unmixing Cuprite scene subject to different constraints: maps and probability density functions.

V-B. We can note on both Table VI and Figure 4 that the residual error is not strongly affected by the constraint choice.

Figure 5 illustrates the effect of the constraint choice on the distribution of the abundance sum per pixel. With positivity only, a sum higher than one is observed in a significant part of the image, which has no physical meaning. Adding the partial additivity constraint provides a sum close to one in most of the pixels. Those who have an abundance sum far from one may reveal a lack of luminosity, an underestimated number of endmembers, or a non-linear phenomenon that cannot be handled with the proposed mixing model.

Using the GPU implementation described in Section V-B, the computational time for unmixing the real data under constraints (3) and (4) becomes 0.50s for the image-based version, and 0.59s for the pixel-based version. The speed up of 33 compared to the CPU version exhibits the suitability of our method for parallel programming.

Constraint	NN	SLO	STO
Time (s)	13.2	17.5	16.4
r ($\times 10^{-4}$)	7.37	8.34	8.52

TABLE VI
COMPUTING TIME AND RESIDUAL NORM AFTER UNMIXING CUPRITE SCENE WITH IPLS SUBJECT TO DIFFERENT CONSTRAINTS.

VI. CONCLUSION

We have proposed in this paper a spectral unmixing algorithm allowing to estimate the abundance maps using a primal-dual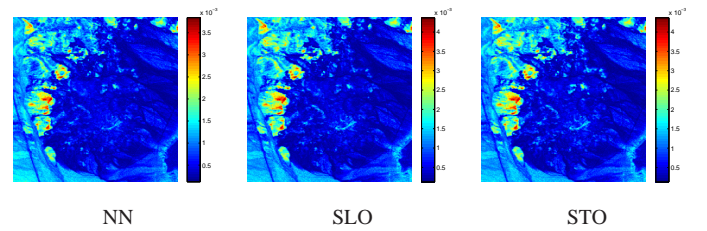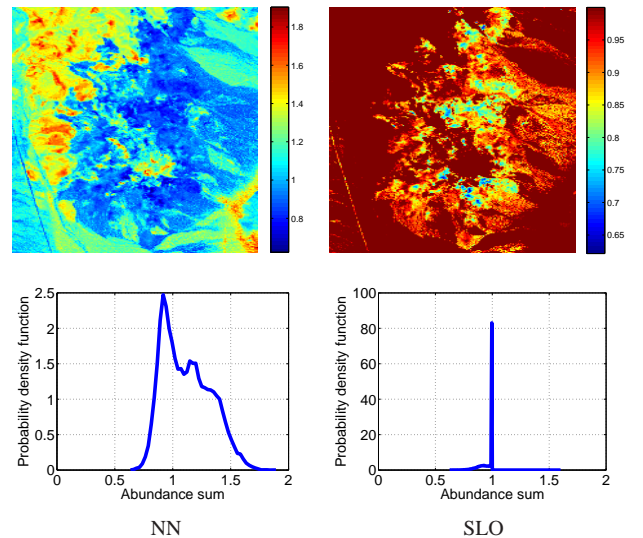 interior-point optimization method. The main feature of the proposed approach is to handle various linear constraints such as full additivity, partial additivity and non-negativity. The second advantage of this approach is its suitability for an efficient parallel implementation using GPUs. These features have been illustrated by processing a real hyperspectral data using two GPU variants of the proposed method (pixel-based or image-based). Implementing the pixel-based version of the method can be extended to the case of sparse unmixing of a single pixel spectrum sing either a sparse recovery approach on a large library or to the case of a large number of spectral bands. On the other hand, the image-based implementation opens the way to fast processing methods including spatial penalization on the abundance maps.

In that respect, the proposed method can be naturally extended to the case of abundance estimation using penalized or weighted least squares estimation, when the regularization function preserves the block diagonal structure of the Hessian matrix. This is for instance the case with Tikhonov regularization or sparse regularization approaches. Future works will be directed to addressing the case of penalization functions that incorporate a spatial regularization of the abundance maps, such as total variation [45] and roughness penalties [46]. Our preliminary results, presented in [47], have shown that the spatial regularization enhances the estimation quality at the price of a significant increase of the numerical complexity. Additional mathematical development are required in order to adapt the primal-dual approach to solve the minimization

problem in this context with a reduced memory requirement and computation cost.

APPENDIX

A. Proof of Theorem III.1

1) Convergence of the inner loop: Let us firstly prove the convergence of the inner loop of Algorithm 1 for a fixed perturbation parameter $\mu > 0$. According to the definition (18), we have, for all $k \in \mathbb{N}$, $\nabla \Psi_\mu(\boldsymbol{a}_k, \boldsymbol{\lambda}_k) =$

$$\begin{bmatrix} \nabla \Phi(\boldsymbol{a}_k) - 2\boldsymbol{T}^{\mathrm{t}} \mathrm{Diag}(\boldsymbol{T}\boldsymbol{a}_k + \boldsymbol{t})^{-1} \boldsymbol{\mu} + \boldsymbol{T}^{\mathrm{t}} \boldsymbol{\lambda}_k \\ \boldsymbol{T}\boldsymbol{a}_k + \boldsymbol{t} - \boldsymbol{\Lambda}_k^{-1} \boldsymbol{\mu} \end{bmatrix} \quad (38)$$

and $\nabla^2 \Psi_\mu(\boldsymbol{a}_k, \boldsymbol{\lambda}_k) =$

$$\begin{bmatrix} \nabla^2 \Phi(\boldsymbol{a}_k) + 2\boldsymbol{T}^{\mathrm{t}} \mathrm{Diag}(\boldsymbol{T}\boldsymbol{a}_k + \boldsymbol{t})^{-2} \boldsymbol{T} \boldsymbol{\mu} & \boldsymbol{T}^{\mathrm{t}} \\ \boldsymbol{T} & \boldsymbol{\Lambda}_k^{-2} \boldsymbol{\mu} \end{bmatrix} \quad (39)$$

For all $k \in \mathbb{N}$, the linesearch ensures that $\boldsymbol{T}\boldsymbol{a}_k + \boldsymbol{t} > \boldsymbol{0}$ and $\boldsymbol{\lambda}_k > \boldsymbol{0}$. Since either $\Phi(\cdot)$ is strictly convex or $\boldsymbol{T}^{\mathrm{t}}\boldsymbol{T}$ is inversible, $\nabla^2 \Psi_\mu(\boldsymbol{a}_k, \boldsymbol{\lambda}_k)$ is positive definite so that Ψ_μ has a unique minimizer $(\hat{\boldsymbol{a}}_\mu, \hat{\boldsymbol{\lambda}}_\mu)$. Then, the same analysis as in [26, Sec.3] allows us to deduce that the sequence $\{(\boldsymbol{a}_k, \boldsymbol{\lambda}_k)\}_{k \in \mathbb{N}}$ resulting from the update equation (13) converges to $(\hat{\boldsymbol{a}}_\mu, \hat{\boldsymbol{\lambda}}_\mu)$.

2) Convergence rate of the inner loop: Lengthy but straightforward calculations show that (16), (17), (38) and (39) lead to

$$\nabla^2 \Psi_\mu(\boldsymbol{a}_k, \boldsymbol{\lambda}_k)\boldsymbol{d}_k + \nabla \Psi_\mu(\boldsymbol{a}_k, \boldsymbol{\lambda}_k) =$$
$$\begin{bmatrix} \mu \boldsymbol{T}^{\mathrm{t}} \mathrm{Diag}(\boldsymbol{T}\boldsymbol{a}_k + \boldsymbol{t})^{-1} \left(\mathrm{Diag}(\boldsymbol{\Lambda}_k^{-1}\boldsymbol{\mu} - (\boldsymbol{T}\boldsymbol{a}_k + \boldsymbol{t})) \right)^{-1} \boldsymbol{T} \boldsymbol{d}_k^a \\ \boldsymbol{\Lambda}_k^{-1} \left(\mathrm{Diag}(\boldsymbol{\Lambda}_k^{-1}\boldsymbol{\mu} - (\boldsymbol{T}\boldsymbol{a}_k + \boldsymbol{t})) \right) \boldsymbol{d}_k^\lambda \end{bmatrix}$$

with $\boldsymbol{d}_k = \begin{bmatrix} (\boldsymbol{d}_k^a)^{\mathrm{t}} & (\boldsymbol{d}_k^\lambda)^{\mathrm{t}} \end{bmatrix}^{\mathrm{t}}$. Let us study the behaviour of $\|\nabla^2 \Psi_{\mu_k}(\boldsymbol{a}_k, \boldsymbol{\lambda}_k)\boldsymbol{d}_k + \nabla \Psi_{\mu_k}(\boldsymbol{a}_k, \boldsymbol{\lambda}_k)\|$ for large values of k. According to [26, Lem.3.1], $\{\boldsymbol{\lambda}_k\}_{k \in \mathbb{N}}$ and $\{\boldsymbol{T}\boldsymbol{a}_k + \boldsymbol{t}\}_{k \in \mathbb{N}}$ are bounded, and bounded away from zero. Moreover, the gradient of Ψ_μ tends to $\boldsymbol{0}$ as k goes to infinity, so that (38) yields $\lim_{k \to \infty} \boldsymbol{\Lambda}_k^{-1}\boldsymbol{\mu} - (\boldsymbol{T}\boldsymbol{a}_k + \boldsymbol{t}) = \boldsymbol{0}$. Therefore,

$$\|\nabla^2 \Psi_{\mu_k}(\boldsymbol{a}_k, \boldsymbol{\lambda}_k)\boldsymbol{d}_k + \nabla \Psi_{\mu_k}(\boldsymbol{a}_k, \boldsymbol{\lambda}_k)\| \underset{k \to \infty}{=} o(\|\boldsymbol{d}_k\|), \quad (40)$$

Hence, applying [48, Th.3.5], there exists k_μ such that the stepsize $\alpha_k = 1$ is admissible for all $k \geqslant k_\mu$ and the sequence $\{(\boldsymbol{a}_k, \boldsymbol{\lambda}_k)\}_{k \geqslant k_\mu}$ resulting from (13) converges q-superlinearly to $(\hat{\boldsymbol{a}}_\mu, \hat{\boldsymbol{\lambda}}_\mu)$.

3) Convergence of the outer loop: For all k such that (23) holds,

$$\|\boldsymbol{r}_{\mu_k}^{\mathrm{prim}}\| \leqslant \sqrt{PN}\|\boldsymbol{r}_{\mu_k}^{\mathrm{prim}}\|_\infty \leqslant \sqrt{PN}\eta^{\mathrm{prim}}\mu_k, \quad (41)$$
$$\|\boldsymbol{r}_{\mu_k}^{\mathrm{dual}}\| \leqslant \delta_k + \mu_k \leqslant (QN\eta^{\mathrm{dual}} + 1)\mu_k. \quad (42)$$

Furthermore,

$$\mu_{k+1} \leqslant \theta\eta^{\mathrm{dual}}\mu_k, \quad (43)$$

with $\theta\eta^{\mathrm{dual}} \in (0, 1)$ and $\mu_0 > 0$ so that $\{\mu_k\}_{k \in \mathbb{N}}$ converges to 0 as k tends to infinity. Thus, according to (41), (42) and [26, Th.5.1], the outer loop of Algorithm 1 generates a bounded sequence $\{(\boldsymbol{a}_k, \boldsymbol{\lambda}_k)\}$ whose accumulation points are primal-dual solutions of problem (11). Finally, if $\Phi(\cdot)$ is strictly convex, the solution $\hat{\boldsymbol{a}}$ of (11) is unique, and the outer iterates $\{\boldsymbol{a}_k\}$ converge to $\hat{\boldsymbol{a}}$.

REFERENCES

[1] J. R. Scott, *Remote Sensing: The Image Chain Approach*, New York: Oxford Univ. Press, 1997.

[2] C.-I. Chang, *Hyperspectral Data Exploitation*, Wiley Interscience, 2007.

[3] J. M. Bioucas-Dias, A. Plaza, N. Dobigeon, M. Parente, D. Qian, P. Gader, and J. Chanussot, "Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 5, no. 2, pp. 354–379, April 2012.

[4] N. Keshava and J. F. Mustard, "Spectral unmixing," *IEEE Signal Processing Magazine*, vol. 19, no. 1, pp. 44 –57, January 2002.

[5] S. Moussaoui, H. Hauksdóttir, F. Schmidt, C. Jutten, J. Chanussot, D. Brie, S. Douté, and J. Benediktsson, "On the decomposition of Mars hyperspectral data by ICA and Bayesian positive source separation," *Neurocomputing*, vol. 71, no. 10-12, pp. 2194–2208, June 2008.

[6] N. Dobigeon, S. Moussaoui, J.-Y. Tourneret, and C. Carteret, "Bayesian separation of spectral sources under non-negativity and full additivity constraints," *Signal Processing*, vol. 89, no. 12, pp. 2657 – 2669, December 2009.

[7] S. Jia and Y. Qian, "Constrained nonnegative matrix factorization for hyperspectral unmixing," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, no. 1, pp. 161 –173, 2009.

[8] A. Huck, M. Guillaume, and J. Blanc-Talon, "Minimum dispersion constrained nonnegative matrix factorization to unmix hyperspectral data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 6, pp. 2590–2602, 2010.

[9] C. L. Lawson and R. J. Hanson, *Solving Least-Squares Problems*, Prentice-Hall, Englewood Cliffs, New Jersey, 1974.

[10] R. Bro and S. De Jong, "A fast non-negativity constrained least squares algorithm," *Journal of Chemometrics*, vol. 11, pp. 393–401, 1997.

[11] J. J. Settle and N. A. Drake, "Linear mixing and the estimation of ground cover proportions," *International Journal of Remote Sensing*, vol. 14, no. 6, pp. 1159–1177, 1993.

[12] D. C. Heinz and C.-I. Chang, "Fully constrained least squares linear spectral mixture analysis method for material quantification in hyperspectral imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 39, no. 3, pp. 529 –545, 2001.

[13] N. Dobigeon, J.-Y. Tourneret, and C.-I. Chang, "Semi-supervised linear spectral unmixing using a hierarchical Bayesian model for hyperspectral imagery," *IEEE Transactions on Signal Processing*, vol. 56, no. 7, pp. 2684–2695, 2008.

[14] A. Plaza, J. Plaza, A. Paz, and S. Sanchez, "Parallel hyperspectral image and signal processing," *IEEE Signal Processing Magazine*, vol. 28, pp. 119–126, May 2011.

[15] S. Sanchez, A. Paz, G. Martin, and A. Plaza, "Parallel unmixing of remotely sensed hyperspectral images on commodity graphics processing units," *Concurrency and Computation: Practice and Experience*, vol. 23, pp. 1538–1557, 2011.

[16] C. González, J. Resano, A. Plaza, and D. Mozos, "FPGA implementation of abundance estimation for spectral unmixing of hyperspectral data using the image space reconstruction algorithm," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 5, no. 1, pp. 248–261, 2012.

[17] P. Honeine and C. Richard, "Geometric unmixing of large hyperspectral images: A barycentric coordinate approach," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 50, no. 6, pp. 2185–2195, June 2012.

[18] R. Heylen, D. Burazerovic, and P. Scheunders, "Fully constrained least squares spectral unmixing by simplex projection," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 11, pp. 4112–4122, November 2011.

[19] J. M. Bioucas-Dias and M. A. T. Figueiredo, "Alternating direction algorithms for constrained sparse regression: Application to hyperspectral unmixing," in *Proceedings of the 2nd IEEE Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS'10)*, Raykjavik, Iceland, 2010.

[20] M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo, "An augmented Lagrangian aproach to the constrained optimization formulation of imaging inverse problems," *IEEE Transactions on Image Processing*, vol. 20, no. 3, pp. 681–695, 2011.

[21] M. H. Wright, "Interior methods for constrained optimization," in *Acta Numerica 1992*, pp. 341–407. Cambridge University Press, 1991.

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

[23] C. A. Johnson, J. Seidel, and A. Sofer, "Interior-point methodology for 3-D PET reconstruction," *IEEE Transactions on Medical Imaging*, vol. 19, no. 4, April 2000.

[24] S. Bonettini and T. Serafini, "Non-negatively constrained image deblurring with an inexact interior point method," *Journal of Computational and Applied Mathematics*, vol. 231, no. 1, pp. 236–248, 2009.

[25] N. Keshava, "A survey of spectral unmixing," *Lincoln Laboratory Journal*, vol. 14, no. 1, pp. 55–78, 2003.

[26] P. Armand, J. C. Gilbert, and S. Jan-Jégou, "A feasible BFGS interior point algorithm for solving strongly convex minimization problems," *SIAM Journal on Optimization*, vol. 11, pp. 199–222, 2000.

[27] S. J. Wright, *Primal-Dual Interior-Point Methods*, SIAM, Philadelphia, PA, 1997.

[28] A. Forsgren, P. E. Gill, and M. H. Wright, "Interior methods for nonlinear optimization," *SIAM Review*, vol. 44, no. 4, pp. 525–597, 2002.

[29] M. H. Wright, "Some properties of the Hessian of the logarithmic barrier function," *Mathematical Programming*, vol. 67, no. 2, pp. 265–295, 1994.

[30] M. H. Wright, "Ill-conditioning and computational error in interior methods for nonlinear programming," *SIAM Journal on Optimization*, vol. 9, no. 1, pp. 84–111, 1998.

[31] A. Conn, N. Gould, and P. L. Toint, "A primal-dual algorithm for minimizing a nonconvex function subject to bounds and nonlinear constraints," in *Nonlinear Optimization and Applications*, G. Di Pillo and F. Giannessi, Eds. Kluwer Academic Publishers, 2 edition, 1996.

[32] A. S. El-Bakry, R. A. Tapia, T. Tsuchiya, and Y. Zhang, "On the formulation and theory of the Newton interior-point method for nonlinear programming," *Journal of Optimization Theory and Applications*, vol. 89, pp. 507–541, June 1996.

[33] C. F. Van Loan, "The ubiquitous Kronecker product," *Journal of Computational and Applied Mathematics*, vol. 123, no. 1-2, pp. 85–100, 2000.

[34] R. N. Clark, G. A. Swayze, A. Gallagher, T. V. King, and W. M. Calvin, "The U.S. geological survey digital spectral library: version 1: 0.2 to 3.0 μm," *U.S. Geological Survey, Denver, CO, Open File Rep. 93-592*, 1993.

[35] A. Plaza and C.-I. Chang, "An improved N-FINDR algorithm in implementation," in *Proc. of SPIE Vol*, 2005, vol. 5806, p. 299.

[36] L. R. Gaddis, L. A. Soderblom, H. H. Kieffer, K. J. Becker, J. Torsen, and K. F. Mullins, "Decomposition of AVIRIS spectra: Extraction of spectral reflectance, atmospheric, and instrumental components," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 34, no. 1, pp. 163–178, January 1996.

[37] J. M. P. Nascimento and J. M. Bioucas-Dias, "Vertex component analysis: a fast algorithm to unmix hyperspectral data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, no. 4, pp. 898 – 910, April 2005.

[38] D. B. Kirk and W. H. Wen-Mei, *Programming massively parallel processors: a hands-on approach*, Morgan Kaufmann, 2010.

[39] C. Nvidia, "NVIDIA CUDA C programming guide version 4.2," 2012.

[40] O. Caspary, P. Nus, and T. Cecchin, "The source number estimation based on gerschgorin radii," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'98)*, 1998, vol. 4, pp. 1993–1996.

[41] C.-I. Chang and D. Qian, "Estimation of number of spectrally distinct signal sources in hyperspectral imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 42, no. 3, pp. 608–619, 2004.

[42] B. Luo, J. Chanussot, D. Sylvain, and L. Zhang, "Empirical automatic estimation of the number of endmembers in hyperspectral images," *IEEE Geoscience and Remote Sensing Letters*, vol. 10, no. 1, pp. 24–28, January 2013.

[43] C.-I. Chang, "Spectral information divergence for hyperspectral image analysis," in *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS'99)*, 1999, vol. 1, pp. 509–511.

[44] A. Plaza, P. Martinez, R. Perez, and J. Plaza, "A quantitative and comparative analysis of endmember extraction algorithms from hyperspectral data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 42, no. 3, pp. 650–663, March 2004.

[45] M. D. Iordache, J. M. Bioucas-Dias, and A. Plaza, "Total variation spatial regularization for sparse hyperspectral unmixing," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 50, no. 11, pp. 4484–4502, 2012.

[46] J. Idier, "Convex half-quadratic criteria and interacting auxiliary variables for image restoration," *IEEE Transactions on Image Processing*, vol. 10, no. 7, pp. 1001–1009, 2001.

[47] S. Moussaoui, E. Chouzenoux, and J. Idier, "Primal-dual interior point optimization for penalized least squares estimation of abundance maps in hyperspectral imaging," in *Proceedings of the IEEE Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS'12)*, Shangai, China, June 2012.

[48] J. Nocedal and S. J. Wright, *Numerical Optimization*, Springer-Verlag, New York, NY, 1999.

Émilie Chouzenoux received the engineering degree from École Centrale, Nantes, France, in 2007, and the Ph. D. degree in signal processing from the Institut de Recherche en Communications et Cybernétique, Nantes, France, in 2010.

She is currently an Assistant Professor with the University of Paris-Est, Champs-sur-Marne, France (LIGM, UMR CNRS 8049). Her research interests are in convex and nonconvex optimization algorithms for large scale problems of image and signal reconstruction.

Maxime Legendre received the engineering degree from École Centrale, Nantes, France, in 2011.

He is currently a Ph.D. student with the Institut de Recherche en Communications et Cybernétique, Nantes (IRCCYN, UMR CNRS 6597). His research interests are in constrained optimization algorithms for inverse problems of image processing, and their applications to spatial imaging.

Saïd Moussaoui received the State engineering degree from Ecole Nationale Polytechnique, Algiers, Algeria, in 2001, and the Ph.D. degree in automatic control and signal processing from Université Henri Poincaré, Nancy, France, in 2005.

He is currently Assistant Professor with École Centrale de Nantes, Nantes, France. Since September 2006, he has been with the Institut de Recherche en Communications et Cybernétique, Nantes (IRCCYN, UMR CNRS 6597). His research interests are in statistical signal and image processing including source separation, Bayesian estimation, and their applications to spectrometry and hyperspectral imaging.

Jérôme Idier was born in France in 1966. He received the Diploma degree in electrical engineering from École Supérieure d'Électricité, Gif-sur-Yvette, France, in 1988, and the Ph.D. degree in physics from University of Paris-Sud, Orsay, France, in 1991.

In 1991, he joined the Centre National de la Recherche Scientifique. He is currently a Senior Researcher with the Institut de Recherche en Communications et Cybernétique, Nantes, France. His major scientific interests are in probabilistic approaches to inverse problems for signal and image processing.

Dr. Idier is currently elected member of the French National Committee for Scientific Research.