Partially Asynchronous Distributed Unmixing of Hyperspectral Images

Pierre-Antoine Thouvenin, *Member, IEEE*, Nicolas Dobigeon, *Senior Member, IEEE* and
Jean-Yves Tourneret, *Senior Member, IEEE*

***Abstract*—So far, the problem of unmixing large or multitemporal hyperspectral datasets has been specifically addressed in the remote sensing literature only by a few dedicated strategies. Among them, some attempts have been made within a distributed estimation framework, in particular relying on the alternating direction method of multipliers (ADMM). In this paper, we propose to study the interest of a partially asynchronous distributed unmixing procedure based on a recently proposed asynchronous algorithm. Under standard assumptions, the proposed algorithm inherits its convergence properties from recent contributions in non-convex optimization, while allowing the problem of interest to be efficiently addressed. Comparisons with a distributed synchronous counterpart of the proposed unmixing procedure allow its interest to be assessed on synthetic and real data. Besides, thanks to its genericity and flexibility, the procedure investigated in this work can be implemented to address various matrix factorization problems.**

***Index Terms*—Partially asynchronous distributed estimation, hyperspectral unmixing, non-convex optimization.**

I. Introduction

Acquired in hundreds of contiguous spectral bands, hyperspectral (HS) images present a high spectral resolution, which is mitigated by a lower spatial resolution in specific applications such as airborne remote sensing. The observed spectra are thus represented as mixtures of signatures corresponding to distinct materials. Spectral unmixing then consists in estimating the reference signatures associated with each material, referred to as endmembers, and their relative fractions in each pixel of the image, referred to as abundances, according to a predefined mixture model. In practice, a linear mixing model (LMM) is traditionally adopted when the declivity of the scene and microscopic interactions between the observed materials are negligible [1]. Per se, HS unmixing can be cast as a blind source separation problem and, under the above assumptions, can be formulated as a particular instance of matrix factorization.

For this particular application, using distributed procedures can be particularly appealing to estimate the abundances since the number of pixels composing the HS images can be orders of magnitude larger than the number of spectral bands in which the images are acquired. In this context, distributed unmixing methods previously proposed in the remote sensing literature essentially rely on synchronous algorithms [2]–[5] with limited convergence guarantees. A different approach consists in resorting to a proximal alternating linearized minimization (PALM) [6], [7] to estimate the mixture parameters (see, e.g., [8]–[10] in this context), which leads to an easily distributable optimization problem when considering the update of the abundances, and benefits from well established convergence results.

While a synchronous distributed variant of the PALM algorithm is particularly appealing to address HS unmixing, this algorithm does not fully exploit the difference in the computing performance of the involved computing units, which is precisely the objective pursued by the numerous asynchronous optimization techniques proposed in the optimization literature (e.g., [11]–[19]). For distributed synchronous algorithms, a master node waits for the information brought by all the available computation nodes (referred to as *workers*) before proceeding to the next iteration (e.g., updating a variable shared between the different nodes, see Fig. 1). On the contrary, asynchronous algorithms offer more flexibility in the sense that they allow more frequent updates to be performed by the computational nodes, thus reducing their idleness time. In particular, asynchronous algorithms can lead to a significant speed up in the algorithm computation time by allowing the available computational units (i.e., cores and machines) to work in parallel, with as few synchronizations (i.e., memory locks) as possible [20]–[22]. For some practical problems, there is no master node, and the workers can become active at any time and independently from the other nodes [21]–[23]. For other applications, a master node first assigns different tasks to all the available workers, then aggregates information from a given node as soon as it receives its information, and launches a new task on this specific node (see Fig. 2). In this partially asynchronous setting, the workers may make use of out-of-date information to perform their local updates [19]. Given the possible advantages brought by the asynchronicity, we propose an asynchronous unmixing procedure based on recent non-convex optimization algorithms. To this end, we consider a centralized architecture as in [24], composed of a master node in charge of a variable shared between the different workers, and Ω workers which have access to a local variable (i.e., only accessible from a given worker) and a (possibly out-of-date) local copy of the shared variable.

Asynchronous methods adapted to the aforementioned con-

This work was supported in part by the Hypanema ANR Project no. ANR-12-BS03-003, by the MapInvPlnt ERA-NET MED Project no. ANR-15-NMED-0002-02, by the Thematic Trimester on Image Processing of the CIMI Labex under Grant ANR-11-LABX-0040-CIMI within the Program ANR-11-IDEX-0002-02 and by the Direction Générale de l'Armement, French Ministry of Defence.

This work has been conducted while P.-A. T. was working with the University of Toulouse. N. D. and J.-Y. T. are with the University of Toulouse, IRIT/INP-ENSEEIHT, 31071 Toulouse, France. (e-mail: pierreantoine.thouvenin@gmail.com, {Nicolas.Dobigeon, Jean-Yves.Tourneret}@enseeiht.fr

text include many recent papers, e.g., [20]–[22], [24], [25]. For HS image unmixing, Gauss-Seidel optimization schemes have proved convenient to decompose the original optimization task into simpler sub-problems, which can be solved or distributed efficiently [26]. We may mention the recently proposed partially asynchronous distributed alternating direction method of multipliers (ADMM) [24], used to solve a distributed optimization task reformulated as a consensus problem. However, HS unmixing does not allow traditional block coordinate descent (BCD) methods (such as the ADMM [27], [28]) to be efficiently applied due to the presence of sub-problems which require iterative solvers. In such cases, the PALM algorithm [6] and its extensions [7], [29], which are sequential algorithms, combine desirable convergence guarantees for non-convex problems with an easily distributable structure in a synchronous setting. Recently, PALM has been extended to accommodate asynchronous updates [21], and analyzed in a stochastic and a deterministic framework. More specifically, the author in [21] considers the general case where all the variables to be estimated are shared by the different workers. However, the explicit presence of a maximum allowable delay in the update steps is problematic, since this parameter is not explicitly controlled by the algorithm. In addition, the residual terms resulting from the allowed asynchronicity have a significant impact on the step-size prescribed to ensure the convergence of the algorithm. In practice, the use of this step-size does not lead to a reduction of the computation time needed to reach convergence, as it will be illustrated in Section IV. From this practical point of view, the algorithm proposed in [24], where the maximum delay is explicitly controlled, appears to be more convenient. However, the use of this ADMM-based algorithm does not ensure that the constraints imposed on the shared variables are satisfied at each iteration, and the sub-problems derived in the context of HS unmixing require the use of iterative procedures. Finally, the strategy developed in [22] allows more flexibility in the allowed asynchronicity, while requiring slightly more stringent assumptions on the penalty functions when compared to [21].

Consequently, this paper proposes to adapt the framework introduced in [22], which encompasses the system structure described in [24], to HS unmixing. Indeed, given the preceding remarks, the framework introduced in [22] appears as one of the most flexible to address HS unmixing in practice. This choice is partly justified by the possible connections between the PALM algorithm and [22]. Indeed, the PALM algorithm enables a synchronous distributed algorithm to be easily derived for matrix factorization problems, which then offers an appropriate reference to precisely evaluate the relevance of the asynchronicity tolerated by the approach described in [22]. Another contribution of this paper consists in assessing the interest of asynchronicity for HS unmixing, in comparison with recently proposed synchronous distributed unmixing procedures.

The paper is organized as follows. The problem addressed in this paper is introduced in Section II. The proposed unmixing procedure is detailed in Section III, along with the assumptions required from the problem structure to recover appropriate convergence guarantees. Simulation results illustrating the performance of the proposed approach on synthetic and real data are presented in Sections IV and V. Finally, Section VI concludes this work and outlines possible research perspectives.

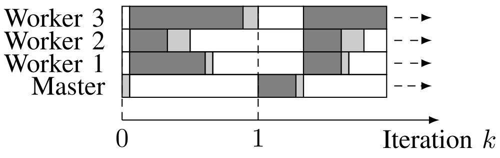


Fig. 1. Illustration of a synchronous distributed mechanism (idle time in white, transmission delay in light gray, computation delay in gray). The master is triggered once it has received information from all the workers.

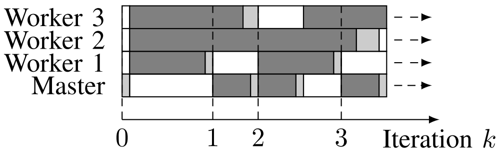


Fig. 2. Illustration of an asynchronous distributed mechanism (idle time in white, transmission delay in light gray, computation delay in gray). The master node is triggered whenever it has received information from K workers ($K = 1$ in the illustration).

II. Problem Formulation

The LMM consists in representing each acquisition by a linear combination of the endmembers $\mathbf{m}_r$, which are present in unknown proportions. Assuming the data are composed of R endmembers, where R is *a priori* known, and considering that the image is divided into Ω subsets of pixels (see Remark 1 for details) to distribute the data between several workers, the LMM can be defined as

$$\mathbf{Y}_\omega = \mathbf{M}\mathbf{A}_\omega + \mathbf{B}_\omega,\ \omega \in \{1,\dots,\Omega\} \tag{1}$$

where $\mathbf{Y}_\omega = [\mathbf{y}_{1,\omega},\dots,\mathbf{y}_{N,\omega}]$ is an $L \times N$ matrix whose columns are the spectral signatures acquired for each pixel of the ωth pixel subset. Note that each group can be assigned a different number of pixels if needed. The columns $\mathbf{m}_r$ of the matrix $\mathbf{M} \in \mathbb{R}^{L\times R}$ are the different endmembers, and the columns $\mathbf{a}_{n,\omega}$ of the abundance matrix $\mathbf{A}_\omega \in \mathbb{R}^{R\times N}$ gather the proportion of the endmembers within $\mathbf{y}_{n,\omega}$. Finally, the matrix $\mathbf{B}_\omega \in \mathbb{R}^{L\times N}$ represents an additive noise resulting from the data acquisition and the modeling errors. The following constraints, aimed at ensuring a physical interpretability of the results, are usually considered

$$\mathbf{A}_\omega \succeq \mathbf{0}_{R,N},\quad \mathbf{A}_\omega^{\mathrm{T}}\mathbf{1}_R = \mathbf{1}_N,\quad \mathbf{M} \succeq \mathbf{0}_{L,R} \tag{2}$$

where $\succeq$ denotes a term-wise inequality. Assuming the data are corrupted by a white Gaussian noise leads to the following data fitting term

$$f_\omega(\mathbf{A}_\omega,\mathbf{M}) = \frac{1}{2}\left\|\mathbf{Y}_\omega - \mathbf{M}\mathbf{A}_\omega\right\|_{\mathrm{F}}^2. \tag{3}$$

In addition, the constraints summarized in (2) are taken into account by defining

$$g_\omega(\mathbf{A}_\omega) = \iota_{\mathcal{A}_N}(\mathbf{A}_\omega) \tag{4}$$

$$\mathcal{A}_N = \left\{ \mathbf{X} \in \mathbb{R}^{R \times N} \mid \mathbf{X}^T \mathbf{1}_R = \mathbf{1}_N, \mathbf{X} \succeq \mathbf{0}_{R,N} \right\} \tag{5}$$

$$r(\mathbf{M}) = \iota_{\{\cdot \succeq \mathbf{0}\}}(\mathbf{M}) \tag{6}$$

where $\iota_{\mathcal{S}}$ denotes the indicator function of a set $\mathcal{S}$ ($\iota_{\mathcal{S}}(\mathbf{x}) = \mathbf{0}$ if $\mathbf{x} \in \mathcal{S}$, $+\infty$ otherwise). This leads to the following optimization problem

$$(\mathbf{A}^*, \mathbf{M}^*) \in \underset{\mathbf{A},\mathbf{M}}{\arg\min}\, \Psi(\mathbf{A}, \mathbf{M}) \tag{7}$$

with

$$\Psi(\mathbf{A}, \mathbf{M}) = F(\mathbf{A}, \mathbf{M}) + G(\mathbf{A}) + r(\mathbf{M}) \tag{8}$$

$$F(\mathbf{A}, \mathbf{M}) = \sum_{\omega=1}^{\Omega} f_\omega(\mathbf{A}_\omega, \mathbf{M}), \quad G(\mathbf{A}) = \sum_{\omega=1}^{\Omega} g_\omega(\mathbf{A}_\omega). \tag{9}$$

With these notations, $\mathbf{A}_\omega$ denotes a *local* variable (i.e., which will be accessed by a single worker), and $\mathbf{M}$ is a global variable (i.e., shared between the different workers, see Fig. 3). More generally, f_ω plays the role of a data fitting term, whereas g_ω and r can be regarded as regularizers or constraints. The structure of the proposed unmixing algorithm, inspired by [22], is detailed in the following section.

Remark 1. In the initial formulation of the mixing model (1), the indexes ω and Ω refer to subsets of pixels. A direct interpretation of this statement can be obtained by dividing a unique (and possibly large) hyperspectral image into Ω non-overlapping tiles of smaller (and possibly different) sizes. In this case, each tile is individually unmixed by a given worker. Another available interpretation allows multitemporal analysis to be conducted. Indeed, in practice, distributed unmixing procedures are of particular interest when considering the unmixing of a sequence of several HS images, acquired by possibly different sensors at different dates, but sharing the same materials [30]–[32]. In this case, ω and Ω could refer to time instants. Each worker ω is then dedicated to the unmixing of a unique HS image acquired at a given time instant. The particular applicative challenge of distributed unmixing of multitemporal HS images partly motivates the numerical experiments on synthetic (yet realistic) and real data presented hereafter.

Remark 2. Even if the work reported in this work has been partly motivated by the particular application of HS unmixing, the problem formulated in this section is sufficiently generic to encompass a wider class of matrix factorization tasks, as those encountered in audio processing [33], machine learning [34], [35].

III. A partially asynchronous unmixing algorithm

A. Algorithm description

Reminiscent of [24], the proposed algorithm relies on a star topology configuration in which a master node supervises an optimization task distributed between several workers. The master node also updates and transmits the endmember matrix $\mathbf{M}$ shared by the different workers. In fact, the computation time of synchronous algorithms is essentially conditioned by the speed of the slowest worker (see Figs. 1 and 2). Consequently, relaxing the synchronization requirements (by allowing bounded delays between the information brought by each worker) allows a significant decrease in the computation time to reach convergence, which can scale almost linearly with the number of workers [21], [24]. Note that, even though asynchronous optimization schemes may require more iterations than their synchronous counterparts to reach a given precision, allowing more frequent updates generally compensates this drawback in terms of computation time [24].

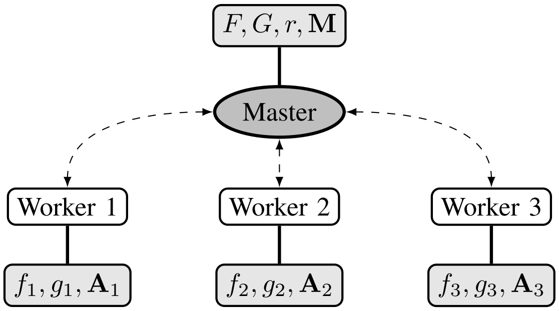


Fig. 3. Illustration of the master-slave architecture considered for the unmixing problem (7) with $\Omega = 3$ workers (the function and variables available at each node are given in light gray rectangles).

In the partially asynchronous setting considered, the master node updates the variable shared by the workers once it has received information from at least $K \ll \Omega$ workers. The new state of the shared variable $\mathbf{M}$ is then transmitted to the K available workers, which can individually proceed to the next step. As in [22], a relaxation step with decreasing stepsizes ensures the convergence of the algorithm (see Algo. 1). In order to clarify to which extent the convergence analysis introduced in [22] is applicable to the present setting, we consider $K = 1$ in the rest of this paper. However, other values of K could be considered without loss of generality. Details on the operations performed by the master node and each worker are detailed in Algos. 1 and 2 respectively.

Remark 3. (a) The parameter γ_k is essentially instrumental to ensure the global convergence of the partially asynchronous unmixing algorithm described in this work, following the general framework introduced in [22]. For simplicity, we have directly adopted the expression proposed in [16] [22, Assumption D., p. 18] which has been reported to yield satisfactory results in practice [16]. Evaluating the practical interest of different expressions for the relaxation parameters in terms of the convergence speed of the algorithm is an interesting prospect, which is however beyond the scope of this paper.

(b) Note that a synchronous distributed counterpart of Algo. 1 can be easily derived for Problem (7), which partly justifies the form chosen for Algo. 1. This version consists in setting $\gamma_k = 1$, and waiting for the updates performed by all the workers (i.e., $K = \Omega$, see Step 1 of Algo. 1) before updating the shared variable $\mathbf{M}$. This implementation will be taken as a reference to evaluate the computational efficiency of the proposed algorithm in Sections IV and V.

Algorithm 1: Master node update.

```
Data: $\mathbf{A}^{(0)}$, $\mathbf{M}^{(0)}$, $\gamma_0 \in (0,1]$, $\mu \in (0,1)$, $N_{\text{iter}}$, $K$.
Broadcast $\mathbf{M}^{(0)}$ to the $\Omega$ workers ;
$k \leftarrow 0$ ;
$\mathcal{T}_k \leftarrow \emptyset$ ;
while $k < N_{iter}$ do
Step 1  Wait for $\hat{\mathbf{A}}^k_{\omega^k}$ from any worker ;
        $\mathcal{T}_k = \mathcal{T}_k \cup \{\omega^k\}$ ;
        $d^{k+1}_\omega = \begin{cases} 0 \text{ if } \omega \in \mathcal{T}_k \\ d^k_\omega + 1 \text{ otherwise} \end{cases}$ ;
        $\mathbf{A}^{k+1}_\omega = \begin{cases} \mathbf{A}^k_\omega + \gamma_k(\hat{\mathbf{A}}_\omega - \mathbf{A}^k_\omega) \text{ if } \omega \in \mathcal{T}_k \\ \mathbf{A}^k_\omega \text{ otherwise} \end{cases}$ ;
        if $(\sharp\mathcal{T}_k < K)$ then
            Go to step Step 1 ;                    // wait until $\sharp\mathcal{T}_k \geq K$
        else
            $\hat{\mathbf{M}}^k \in \text{prox}_{c^k_{\mathbf{M}} r}\Big(\mathbf{M}^k + \frac{1}{c^k_{\mathbf{M}}}\nabla_{\mathbf{M}} F(\mathbf{A}^{k+1}, \mathbf{M}^k)\Big)$;
            $\mathbf{M}^{k+1} = \hat{\mathbf{M}}^k + \gamma_k(\hat{\mathbf{M}}^k - \mathbf{M}^k)$;
            $\gamma_{k+1} = \gamma_k(1 - \mu\gamma_k)$;
            $\mathcal{T}_{k+1} \leftarrow \emptyset$ ;
            $k \leftarrow k+1$;
Result: $\mathbf{A}^{N_{\text{iter}}}$, $\mathbf{M}^{N_{\text{iter}}}$.
```

B. Parameter estimation

A direct application of the algorithm described in Algo. 2 under the constraints (2) leads to the following update rule for the abundance matrix $\mathbf{A}_{\omega^k}$

$$\widehat{\mathbf{A}}^k_{\omega^k} = \text{prox}_{\iota_{\mathcal{A}_N}}\left(\mathbf{A}^k_\omega - \frac{1}{c^k_{\mathbf{A}_{\omega^k}}}\nabla_{\mathbf{A}_\omega} f_\omega\big(\mathbf{A}^k_{\omega^k}, \mathbf{M}^{k-d^k_{\omega^k}}\big)\right) \tag{10}$$

where $\text{prox}_{\iota_{\mathcal{A}_N}}$ denotes the proximal operator of the indicator function $\iota_{\mathcal{A}_N}$ (see, e.g., [36]), and

$$\nabla_{\mathbf{A}_\omega} f_\omega(\mathbf{A}_\omega, \mathbf{M}) = \mathbf{M}^{\mathrm{T}}\big(\mathbf{M}\mathbf{A}_\omega - \mathbf{Y}_\omega\big). \tag{11}$$

The step-size $c^k_{\mathbf{A}_{\omega^k}}$ is chosen as in the standard PALM algorithm, i.e.,

$$c^k_{\mathbf{A}_{\omega^k}} = L^k_{\mathbf{A}_{\omega^k}} = \left\|(\mathbf{M}^{k-d^k_{\omega^k}})^{\mathrm{T}}\mathbf{M}^{k-d^k_{\omega^k}}\right\|_2 \tag{12}$$

where $L^k_{\mathbf{A}_{\omega^k}}$ denotes the Lipschitz constant of $\nabla_{\mathbf{A}_\omega} f_\omega(\cdot, \mathbf{M}^{k-d^k_{\omega^k}})$ (see [6, Remark 4 (iv)]). Note that the projection $\text{prox}_{\iota_{\mathcal{A}_N}}(\cdot)$ can be exactly computed (see [37], [38] for instance). Similarly, the update rule for the endmember matrix $\mathbf{M}$ is

$$\widehat{\mathbf{M}}^k = \text{prox}_{\iota_{\{\cdot \succeq 0\}}}\left(\mathbf{M}^k - \frac{1}{c^k_{\mathbf{M}}}\nabla_{\mathbf{M}} F\big(\mathbf{A}^{k+1}, \mathbf{M}^k\big)\right) \tag{13}$$

with

$$\nabla_{\mathbf{M}} F\big(\mathbf{A}, \mathbf{M}\big) = \sum_\omega (\mathbf{M}\mathbf{A}_\omega - \mathbf{Y}_\omega)\mathbf{A}^{\mathrm{T}}_\omega \tag{14}$$

$$c^k_{\mathbf{M}} = L^k_{\mathbf{M}} = \|\sum_\omega \mathbf{A}^{k+1}_\omega(\mathbf{A}^{k+1}_\omega)^{\mathrm{T}}\|_2. \tag{15}$$

and $L^k_{\mathbf{M}}$ is the Lipschitz constant of $\nabla_{\mathbf{M}} F\big(\mathbf{A}^k, \cdot\big)$.

C. Convergence guarantees

In general, the proposed algorithm requires the following assumptions, based on the convergence results given in [6, Theorem 1] and [22, Theorem 1].

Algorithm 2: ωth worker update (since the shared variable $\mathbf{M}$ may have been updated by the master node in the meantime, $\tilde{\mathbf{M}}$ corresponds to a possibly delayed version of the current $\mathbf{M}^k$). From the master point of view, $\tilde{\mathbf{M}} = \mathbf{M}^{k-d^k_\omega}$.

```
Data: $\tilde{\mathbf{M}}$, $\tilde{\mathbf{A}}_\omega$.
begin
    Wait for $(\tilde{\mathbf{M}}, \tilde{\mathbf{A}}_\omega)$ from the master node;
    $\hat{\mathbf{A}}_\omega \in \text{prox}_{c_{\mathbf{A}_\omega} g_\omega}\Big(\tilde{\mathbf{A}}_\omega - \frac{1}{c_{\mathbf{A}_\omega}}\nabla_{\mathbf{A}_\omega} f_\omega(\tilde{\mathbf{A}}_\omega, \tilde{\mathbf{M}})\Big)$;
    Transmit $\hat{\mathbf{A}}_\omega$ to the master node;
Result: $\hat{\mathbf{A}}_\omega$.
```

Assumption 1 (Algorithmic assumption)**.** Let $(\omega_k, d^k_{\omega^k}) \in \{1, \dots, \Omega\} \times \{1, \dots, \tau\}$ denote the couple composed of the index of the worker transmitting information to the master at iteration k, and the delay between the (local) copy $\tilde{\mathbf{M}}^k$ of the endmember matrix $\mathbf{M}$ and the current state $\mathbf{M}^k$ (i.e., $\tilde{\mathbf{M}}^k \triangleq \mathbf{M}^{k-d^k_{\omega^k}}$). The allowable delays $d^k_{\omega^k}$ are assumed to be bounded by a constant $\tau \in \mathbb{N}^*$. In addition, each couple $(\omega_k, d^k_{\omega^k})$ represents a realization of a random vector within the probabilistic model introduced in [22, Assumption C].

Assumption 2 (Inherited from PALM [6])**.**

(i) For any $\omega \in \{1, \dots, \Omega\}$, $g_\omega : \mathbb{R}^{R\times N} \to (-\infty, +\infty]$ and $r : \mathbb{R}^{L\times R} \to (-\infty, +\infty]$ are proper, convex lower semi-continuous (l.s.c.) functions;

(ii) For $\omega \in \{1, \dots, \Omega\}$, $f_\omega : \mathbb{R}^{R\times N} \times \mathbb{R}^{L\times R} \to \mathbb{R}$ is a $\mathcal{C}^1$ function, and is convex with respect to each of its variables when the other is fixed;

(iii) Ψ, f_ω, g_ω, and r are lower bounded, i.e., $\inf_{\mathbb{R}^{R\times N}\times\mathbb{R}^{L\times R}} \Psi > -\infty$, $\inf_{\mathbb{R}^{R\times N}\times\mathbb{R}^{L\times R}} f_\omega > -\infty$, $\inf_{\mathbb{R}^{R\times N}} g_\omega > -\infty$, and $\inf_{\mathbb{R}^{L\times R}} r > -\infty$;

(iv) Ψ is a coercive semi-algebraic function (see [6]);

(v) For all $\omega \in \{1, \dots, \Omega\}$, $\mathbf{M} \in \mathbb{R}^{L\times R}$, $\mathbf{A}_\omega \mapsto f_\omega(\mathbf{A}_\omega, \mathbf{M})$ is a $\mathcal{C}^1$ function, and the partial gradient $\nabla_{\mathbf{A}_\omega} f_\omega(\cdot, \mathbf{M})$ is Lipschitz continuous with Lipschitz constant $L_{\mathbf{A}_\omega}(\mathbf{M})$. Similarly, $\mathbf{M} \mapsto f_\omega(\mathbf{A}_\omega, \mathbf{M})$ is a $\mathcal{C}^1$ function, and the partial gradient $\nabla_{\mathbf{M}} f_\omega(\mathbf{A}_\omega, \cdot)$ is Lipschitz continuous, with constant $L_{\mathbf{M},\omega}(\mathbf{A}_\omega)$;

(vi) the Lipschitz constants used in the algorithm, i.e., $L_{\mathbf{A}_{\omega^k}}(\mathbf{M}^k)$ and $L_{\mathbf{M},\omega_k}(\hat{\mathbf{A}}^k_{\omega^k})$ (denoted by $L^k_{\mathbf{A}_{\omega^k}}$ and $L^k_{\mathbf{M},\omega_k}$ in the following) are bounded, i.e. there exists appropriate constants such that for all iteration index k

$$0 < L^-_{\mathbf{A}} \leq L^k_{\mathbf{A}_{\omega^k}} \leq L^+_{\mathbf{A}}, \quad 0 < L^-_{\mathbf{M}} \leq L^k_{\mathbf{M},\omega^k} \leq L^+_{\mathbf{M}}.$$

(vii) ∇F is Lipschitz continuous on bounded subsets.

Assumption 3 (Additional assumptions)**.** (i) For all $\omega \in \{1, \dots, \Omega\}$, $\mathbf{A}_\omega \in \mathbb{R}^{R\times N}$, $\nabla_{\mathbf{A}_\omega} f_\omega(\mathbf{A}_\omega, \cdot)$ is Lipschitz continuous with Lipschitz constant $L_{\mathbf{A}_\omega,\mathbf{M}}(\mathbf{A}_\omega)$;

(ii) The Lipschitz constants $L_{\mathbf{A}_{\omega_k},\mathbf{M}}(\hat{\mathbf{A}}^k_{\omega^k})$ (denoted by $L^k_{\mathbf{A}_{\omega_k},\mathbf{M}}$ in the following) is bounded, i.e. there exists appropriate positive constants such that for all $k \in \mathbb{N}$:

$$0 < L^-_{\mathbf{A},\mathbf{M}} \leq L^k_{\mathbf{A}_{\omega_k},\mathbf{M}} \leq L^+_{\mathbf{A},\mathbf{M}}.$$

Assumption 1 summarizes standard algorithmic assumptions to ensure the convergence of Algo. 1. Besides, Assumption 2 gathers requirements of the traditional PALM algorithm [6], under which the distributed synchronous version of the proposed algorithm can be ensured to converge.

Note that the non-convex problem (7) obviously satisfies Assumptions 2 to 3 for the functions defined in Section II (see [6] for examples of semi-algebraic functions). In particular, the bounds on the Lipschitz constants involved in Assumptions 2(vii) and 3(ii) are satisfied in practice, considering the fact that hyperspectral unmixing is generally conducted on reflectance data (implying $\mathbf{Y}_\omega \in [0,1]^{L\times N}$), and given the constraints imposed on $\mathbf{A}_\omega$ and $\mathbf{M}$ respectively.

Under Assumptions 1 to 3, the analysis led in [22] allows the following convergence result to be satisfied.

Proposition 1. *Suppose that Problem (7) satisfies the requirements specified in Assumptions 1 to 3. Define the sequence $\{\mathbf{v}^k\}_{k\in\mathbb{N}}$ of the iterates generated by Algos. 1 and 2, with $\mathbf{v}^k \triangleq (\mathbf{A}^k, \mathbf{M}^k)$ and the parameters in Algo. 2 chosen as*

$$c^k_{\mathbf{A}_{\omega^k}} = L^k_{\mathbf{A}_{\omega^k}}, \quad c^k_{\mathbf{M}} = L^k_{\mathbf{M}}.$$

Then, the following convergence results are obtained:

(i) the sequence $\{\Psi(\mathbf{v}^k)\}_{k\in\mathbb{N}}$ converges almost surely;

(ii) every limit point of the sequence $\{\mathbf{v}^k\}_{k\in\mathbb{N}}$ is a critical point of Ψ almost surely.

Proof. See sketch of proof in the Appendix. □

The convergence analysis is conducted using an auxiliary function (introduced in Lemma 2 in Appendix) to handle asynchronicity [21]. The resulting convergence guarantees then allow convergence results associated with the original problem (7) to be recovered.

Besides, the following result ensures a stronger convergence guarantee for the synchronous counterpart of Algo. 1.

Proposition 2 (Finite length property, following from [6])**.** *Suppose that Problem (7) satisfies the requirements specified in Assumptions 2 to 3. Define the sequence $\{\mathbf{v}^k\}_{k\in\mathbb{N}}$ of the iterates generated by the synchronous version of Algo. 1, with $\mathbf{v}^k \triangleq (\mathbf{A}^k, \mathbf{M}^k)$ and*

$$c^k_{\mathbf{A}_{\omega^k}} = L^k_{\mathbf{A}_{\omega^k}}, \quad c^k_{\mathbf{M}} = L^k_{\mathbf{M}}, \quad \gamma_k = 1, \quad K = \Omega.$$

Then, the following properties can be proved:

(i) the sequence $\{\mathbf{v}^k\}_{k\in\mathbb{N}}$ has finite length

$$\sum_{k=1}^{+\infty} \left\|\mathbf{v}^{k+1} - \mathbf{v}^k\right\| < +\infty$$

where

$$\left\|\mathbf{v}^{k+1} - \mathbf{v}^k\right\| = \sqrt{\left\|\mathbf{A}^{k+1} - \mathbf{A}^k\right\|_F^2 + \left\|\mathbf{M}^{k+1} - \mathbf{M}^k\right\|_F^2};$$

(ii) the sequence $\{\mathbf{v}^k\}_{k\in\mathbb{N}}$ converges to a critical point of Ψ.

Proof. These statements result from a direct application of [6, Theorem 1, Theorem 3] and [6, Remark 4 (iv)]. □

Note that an additional volume regularization can be considered, as long as it satisfies the conditions given in Assumption 2, and more specifically the convexity Assumption 2(i). For instance, the mutual distance between the endmembers introduced in [39] can be easily accounted for.

IV. Experiments with synthetic data

To illustrate the interest of the allowed asynchronicity, we compare the estimation performance of Algo. 1 to the performance of its synchronous counterpart (described in Section III), and evaluate the resulting unmixing performance in comparison with three unmixing methods proposed in the literature. We propose to consider the context of multitemporal HS unmixing, which is of particular interest for recent remote sensing applications [30]–[32]. For this application, a natural way of distributing the data consists in assigning a single HS image to each worker. To this end, we generated synthetic data composed of $\Omega = 3$ HS images resulting from linear mixtures of $R \in \{3, 6, 9\}$ endmembers acquired in $L = 413$ bands. The generated abundance maps vary smoothly over time (i.e., from one image to another) to reproduce a realistic evolution of the scene of interest. As in [40, Section V], the abundance maps were obtained by multiplying reference abundance coefficients with trigonometric functions to ensure a sufficiently smooth temporal evolution. For the dataset with $R = 3$, the reference abundance map was obtained by unmixing the Moffett scene (same area as in [41]). For the datasets composed of $R \in \{6, 9\}$ endmembers, we directly used the synthetic abundance maps introduced in [42] as a reference[1]. Each image, composed of $10,000$ pixels, was then corrupted by an additive white Gaussian noise whose variance ensures a signal-to-noise ratio (SNR) of 30 dB.

Note that the distributed methods were run on a single computer for illustration purposes using the built-in low level distributed computing instructions available in Julia [43] (which provide an MPI-like interface). In this case, the workers are independent processes.

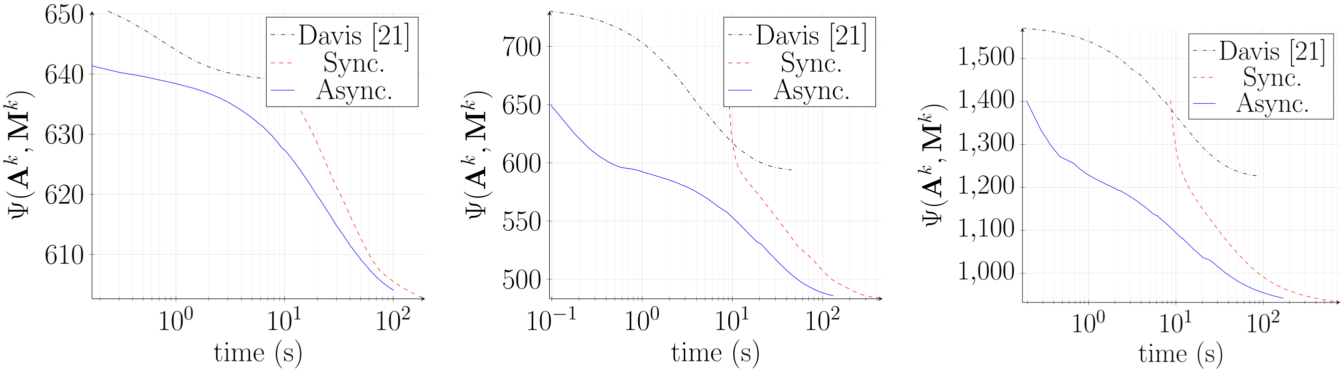
As is common with many blind unmixing algorithms, the performance of the proposed approach is expected to be limited in cases where the initial endmember matrix does not properly represent the observed materials. This observation essentially results from the nonconvex nature of the problem presently addressed, and is not specific to the proposed approach. To the best of the authors' knowledge, no blind unmixing algorithm can systematically ensure the convergence of the generated iterates to a "satisfactory" critical point of the objective function in cases where the initialization is relatively poor.

A. Compared methods

The estimation performance of the proposed algorithm has been compared to those of several unmixing methods from the literature. Note that only the computation times associated with Algo. 1 and its synchronous version, implemented in Julia [43], can lead to a consistent comparison in this experiment. Indeed, some of the other unmixing methods have been implemented in MATLAB by their respective authors. In the following lines, implementation details specific to each of these methods are given.

1) VCA/FCLS: the endmembers are first extracted on each image using the vertex component analysis (VCA) [44],

[1] Abundance maps available at http://www.umbc.edu/rssipl/people/aplaza/fractals.zip.



(a) $R = 3$

(b) $R = 6$

(c) $R = 9$

Fig. 4. Evolution of the objective function for the synthetic datasets, obtained for Algo. 1 and its synchronous version until convergence (model (1)).

which requires pure pixels to be present. The abundances are then estimated for each pixel by solving a fully constrained least squares problem (FCLS) using the ADMM algorithm described in [45];

2) SISAL/FCLS: the endmembers are extracted on each image by the simplex identification via split augmented Lagrangian (SISAL) [46], and the abundances are estimated for each pixel by FCLS. The tolerance for the stopping rule is set to 10^{-4};
3) Proposed method (referred to as ASYNC): the endmembers are initialized with the signatures obtained by VCA on the first image of the sequence, and the abundances are initialized by FCLS. The synchronous and asynchronous algorithms are stopped when the relative decrease of the objective function between two consecutive iterations is lower than 10^{-5}, with a maximum of 100 and 500 iterations respectively. Its synchronous counterpart is referred to as SYNC. The relaxation parameter γ_k ($k \in \mathbb{N}^*$) is updated as in [22] with $\gamma_0 = 1$ and $\mu = 10^{-6}$ (see Algo. 1). In the absence of any temporal or spatial reglarization, the lexicographically ordered pixels composing the datasets are evenly distributed between $\Omega = 3$ workers;
4) DAVIS [21]: this asynchronous algorithm only differs from the previous algorithm, in that no relaxation step is considered, and in the expression of the descent stepsize used to ensure the algorithm convergence. To ensure a fair comparison, it has been run in the same setting as the proposed asynchronous method;
5) DSPLR [5]: the DSPLR algorithm is considered with the stopping criterion proposed in [5] (set to $\varepsilon = 10^{-5}$), with a maximum of 100 iterations. The same initialization as the two previous distributed algorithms is used.

The estimation performance reported in Table I are evaluated in terms of

(i) endmember estimation and spectral reconstruction through the average spectral angle mapper (aSAM)

$$\mathrm{aSAM}(\mathbf{M}) = \frac{1}{R}\sum_{r=1}^{R} \arccos\left(\frac{\mathbf{m}_r^T \widehat{\mathbf{m}}_r}{\|\mathbf{m}_r\|_2 \|\widehat{\mathbf{m}}_r\|_2}\right) \tag{16}$$

$$\mathrm{aSAM}(\mathbf{Y}) = \frac{1}{N\Omega}\sum_{n,\omega} \arccos\left(\frac{\mathbf{y}_{n,\omega}^T (\widehat{\mathbf{M}}\hat{\mathbf{a}}_{n,\omega})}{\|\mathbf{y}_{n,\omega}\|_2 \|\widehat{\mathbf{M}}\hat{\mathbf{a}}_{n,\omega}\|_2}\right); \tag{17}$$

TABLE I
SIMULATION RESULTS ON SYNTHETIC DATA (GMSE($\mathbf{A}$)×10^{-3}, RE ×10^{-4}).

	Algorithm		aSAM($\mathbf{M}$) (°)	GMSE($\mathbf{A}$)	RE	aSAM($\mathbf{Y}$) (°)	time (s)
$R = 3$	VCA/FCLS	[44]	1.82	1.27	0.64	1.45	**1**
	SISAL/FCLS	[46]	1.55	0.94	0.62	1.43	2
	DSPLR	[5]	0.84	2.76	**0.59**	**1.41**	139
	DAVIS	[21]	1.44	0.92	0.63	1.45	10
	SYNC		**0.76**	**0.33**	0.60	**1.41**	197
	ASYNC		0.85	0.38	0.60	**1.41**	101
$R = 6$	VCA/FCLS	[44]	2.55	1.08	1.11	1.64	**1**
	SISAL/FCLS	[46]	1.65	0.50	**0.91**	1.53	2.5
	DSPLR	[5]	3.64	4.65	7.73	**1.45**	116
	DAVIS	[21]	1.87	1.22	0.96	1.58	45.3
	SYNC		**0.63**	**0.28**	**0.78**	**1.45**	462
	ASYNC		1.09	0.59	0.81	1.48	46
$R = 9$	VCA/FCLS	[44]	3.07	2.59	6.75	2.37	**2**
	SISAL/FCLS	[46]	2.17	1.77	5.11	2.14	4
	DSPLR	[5]	8.52	6.53	1.48	**1.56**	153
	DAVIS	[21]	1.57	1.27	1.98	1.69	84
	SYNC		**0.87**	**0.40**	**1.50**	1.57	762
	ASYNC		0.88	0.54	1.52	1.58	170

(ii) abundance estimation through the global mean square error (GMSE)

$$\mathrm{GMSE}(\mathbf{A}) = \frac{1}{\Omega R N}\sum_{\omega=1}^{\Omega} \|\mathbf{A}_\omega - \widehat{\mathbf{A}}_\omega\|_{\mathrm{F}}^2; \tag{18}$$

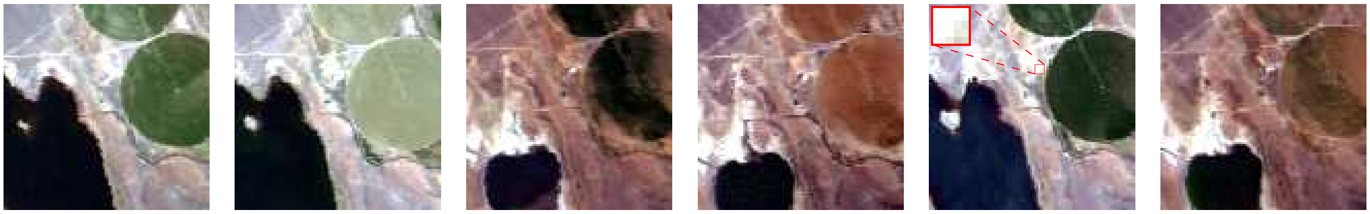
(iii) quadratic reconstruction error (RE)

$$\mathrm{RE} = \frac{1}{\Omega L N}\sum_{\omega=1}^{\Omega} \|\mathbf{Y}_\omega - \widehat{\mathbf{M}}\widehat{\mathbf{A}}_\omega\|_{\mathrm{F}}^2. \tag{19}$$

B. Results

The results reported in Table I correspond to a single trial of the different algorithms. More precisely, the results reported for VCA/FCLS are representative of the results obtained over multiple runs, which have not been observed to vary significantly from one run to another. A similar observation has been made for multiple runs of the asynchronous algorithms (ASYNC and DAVIS) whose performance does not change significantly over different runs for the simulation setting adopted in this paper, both in terms of estimation accuracy and computation time.

- **Endmember estimation:** the proposed asynchronous algorithm leads to competitive endmember estimation for the three synthetic datasets (in terms of aSAM and RE),



(a) 04/10/2014 (b) 06/02/2014 (c) 09/19/2014 (d) 11/17/2014 (e) 04/29/2015 (f) 10/13/2015

Fig. 5. Mud lake dataset used in the MTHS experiment with the corresponding acquisition dates. The area delineated in red in Fig. 5(e) highlights a region known to contain outliers (this observation results from a previous analysis led on this dataset in [31]).

notably in comparison with its synchronous counterpart. We can note that the DSPLR algorithm yields interesting estimation results for $R = 3$, which however significantly degrade as R increases. This partly results from the matrix inversions involved in the update steps of [5], which remain relatively sensitive to the conditioning of the involved matrices, and consequently to the choice of the regularization parameter of the augmented Lagrangian.

- **Abundance estimation:** the synchronous PALM algorithm leads to the best abundance estimation results, even in the absence of any additional regularization on the spatial distribution of the abundances. In this respect, we can note that the performance of PALM and its asynchronous version is relatively similar, and consistently outperforms the other unmixing methods.
- **Overall performance:** the performance measures reported in Table I show that the proposed distributed algorithm yields competitive estimation results, especially in terms of the required computational time when compared to its synchronous counterpart. To be more explicit, the evolution of the objective function versus the computation time shows the interest of the allowed asynchronicity to speed up the unmixing task, as illustrated in Fig. 4 (the computation time required by Algo. 1 is almost 4 times lower than the one of its synchronous counterpart).

Note that even though the SYNC and ASYNC algorithms start from the same initial point, there is no guarantee that both methods converge to the same critical point, which essentially accounts for the differences in the results reported for both methods in Table I. For the asynchronous algorithms, another potential source of variability comes from the variations in the order the updates are performed from one run to another. For the simulation setting adopted in this work, such variations have not been observed to lead to significant differences in the estimation results.

V. Experiments with real data

In practice, as emphasized earlier, distributed unmixing procedures are of particular interest when considering the unmixing of large HS images, or of a sequence of HS images acquired by possibly different sensors at different time instants [30]–[32], referred to as multitemporal HS (MTHS) images. The unmixing of two large real HS images is first proposed, whereas the application to MTHS images essentially motivates the last example addressed in this section. The experiments have been conducted in the same setting as in the previous section (the pixels composing the datasets are evenly distributed between $\Omega = 3$ workers).

A. Description of the datasets

a) Cuprite dataset (single HS image): the first dataset considered in this work consists of a 190×250 subset extracted from the popular Cuprite dataset. In this case, reference abundance maps are available from the literature (see for instance [44], [47]). After removing water-absorption and low SNR bands, 189 out of the 224 spectral bands initially available were exploited in the subsequent unmixing procedure. The data have been unmixed with $R = 10$ endmembers based on prior studies conducted on this dataset [44], [47].

b) Houston dataset (single HS image): the second dataset considered hereafter was acquired over the campus of the University of Houston, Texas, USA, in June 2012 [48]. The 152×108 scene of interest is composed of 144 bands acquired in the wavelength range 380nm to 1050nm. The data have been unmixed with $R = 4$ endmembers based on prior studies conducted on this dataset [49].

c) Mud lake dataset (MTHS images): we finally consider a real sequence of AVIRIS HS images acquired between 2014 and 2015 over the Mud Lake, located in the Lake Tahoe region (California, United States of America)[2]. The 100×100 scene of interest is in part composed of a lake and a nearby field displayed in Fig. 5. The images have been unmixed with $R = 3$ endmembers based on results obtained from prior studies conducted on these data [31], [50], and confirmed by the results of the noise-whitened eigengap algorithm (NWEGA) [51] reported in Table II. After removing the water absorption bands, 173 out of the 224 available spectral bands were finally exploited.

B. Results

Given the absence of ground truth for the different datasets (except the indications available in the literature for the

[2] The images from which the interest of interest is extracted are freely available from the online AVIRIS flight locator tool at http://aviris.jpl.nasa.gov/alt_locator/.

TABLE II
ENDMEMBER NUMBER R ESTIMATED BY NWEGA [51] ON EACH IMAGE OF THE MUD LAKE DATASET.

	04/10/2014	06/02/2014	09/19/2014	11/17/2014	04/29/2015	10/13/2015
NWEGA	3	3	3	4	3	4

Cuprite scene [44], [47]), the estimation results obtained by the proposed algorithms are compared to the other unmixing procedures in terms of the RE and the aSAM introduced in (17) and (19) respectively (see Table III). The consistency of the estimated abundance maps, reported in Figs. 6 to 10, is also considered when analyzing the different results.

a) Cuprite dataset: except for the DSPLR algorithm, whose scale indeterminacy leads to results somewhat harder to interpret for this dataset, the results obtained by the different methods are relatively similar, both in terms of the estimated abundance maps and the recovered endmembers (see Fig. 6).

b) Houston dataset: the distributed algorithms yield abundance maps in agreement with the VCA/FCLS and SISAL algorithms (see Fig. 7). We can note that the algorithms SYNC, ASYNC and DSPLR provide a more contrasted abundance map for the concrete than VCA/FCLS, SISAL/FCLS and DAVIS.

c) Mud lake dataset: the algorithms SYNC, DAVIS [21] and ASYNC lead to particularly convincing abundance maps, in the sense that the abundances of the different materials (containing soil, water and vegetation) are consistently estimated for each time instant, contrary to VCA/FCLS, SISAL/FCLS and DSPLR (see Figs. 8 to 10). At $\omega = 5$, VCA/FCLS and SISAL, which have been applied individually to each image of the sequence, appear to be particularly sensitive to the presence of outliers in the area delineated in red in Fig. 5(e) (see [31] for a previous study on this dataset). This observation is further confirmed by the abundance maps reported at $t = 5$ in Figs. 8 and 9, as well as the corresponding endmembers reported in Fig. 11 (whose amplitude is significantly greater than 1). This sensitivity notably results from the fact that each scene has been analyzed independently from the others in this specific context (note that the results would have been worse if these methods were applied to all the images at once).

d) Global reconstruction performance: the performance measures reported for the different datasets in Table III confirm the interest of the PALM algorithm and its asynchronous variant for unmixing applications. The asynchronous variant can be observed to lead to a notable reduction of the computation time (see also Fig. 12), while allowing a reconstruction performance similar to the classical PALM algorithm to be obtained.

VI. CONCLUSION AND FUTURE WORK

This paper focused on a partially asynchronous distributed unmixing algorithm based on recent contributions in nonconvex optimization [21], [22], [24], which proves convenient to address large scale hyperspectral unmixing problems. Under relatively standard conditions, the proposed approach inherits from the convergence guarantees studied in [22], and from those of the traditional PALM algorithm [6], [7] for its synchronous counterpart. Evaluated on synthetic and real data, the proposed approach provided competitive estimation results, while significantly reducing the computation time to reach convergence. From a computational point of view, implementing a fully functional, large scale asynchronous unmixing algorithm and assessing its scalability with respect to the volume of data involved is an interesting prospect. As with any distributed algorithm, the computation time required by the proposed method is expected to decrease linearly with the number of workers assigned to the unmixing task until the cost of the master/worker communications is comparable to the cost of the estimation task conducted on each worker. Future research perspectives also include the extension to different network topology as in [18], [23], or the use of variable metrics as described in [7], [8], [29], [52].

TABLE III
SIMULATION RESULTS ON REAL DATA (RE $\times 10^{-4}$).

	Algorithm		RE	aSAM($\mathbf{Y}$) (°)	time (s)
Cuprite	VCA/FCLS	[44]	0.51	0.96	**2**
	SISAL/FCLS	[46]	0.47	0.92	6
	DSPLR	[5]	1.25	1.42	20.2
	DAVIS	[21]	0.33	0.79	64.0
	SYNC		**0.15**	**0.55**	1290
	ASYNC		0.30	0.77	134
Houston	VCA/FCLS	[44]	22.5	3.31	**0.1**
	SISAL/FCLS	[46]	21.3	2.01	0.6
	DSPLR	[5]	**0.13**	**0.99**	51.5
	DAVIS	[21]	14.9	2.44	22.3
	SYNC		0.21	1.14	84.6
	ASYNC		0.24	1.17	24.9
Mud lake	VCA/FCLS	[44]	23.7	13.23	**1**
	SISAL/FCLS	[46]	**1.65**	**3.09**	2
	DSPLR	[5]	1.93	10.9	99.6
	DAVIS	[21]	17.61	6.27	58.9
	SYNC		5.05	5.88	70.4
	ASYNC		5.13	5.88	35.0

APPENDIX

The proposed sketch of proof adapts the first arguments developed in [22], in order to clarify that the proposed algorithm fits within this general framework. Note that a similar proof can be obtained by induction when J blocks have to be updated by each worker, and I blocks by the master node (corresponding to the situation described in (7)).

Lemma 1. *Under Assumptions 1 to 3, there exists two positive constants $c_{\mathbf{A}}$ and $c_{\mathbf{M}}$ such that*

$$\begin{aligned}\Psi(\mathbf{A}^{k+1}, \mathbf{M}^{k+1}) &\leq \Psi(\mathbf{A}^k, \mathbf{M}^k) \\ &- \frac{\gamma_k}{2}\big(c_{\mathbf{A}} - \gamma_k(L_{\mathbf{A}}^+ + L_{\mathbf{A},\mathbf{M}}^+)\big)\big\|\hat{\mathbf{A}}_{\omega^k}^k - \mathbf{A}_{\omega^k}^k\big\|^2 \\ &- \frac{\gamma_k}{2}\big(c_{\mathbf{M}} - \gamma_k L_{\mathbf{M}}^+\big)\big\|\hat{\mathbf{M}}^k - \mathbf{M}^k\big\|^2 \\ &+ \frac{1}{2}\tau L_{\mathbf{A},\mathbf{M}}^+ \sum_{q=k-\tau+1}^{k} \big\|\mathbf{M}^q - \mathbf{M}^{q-1}\big\|^2. \qquad (20)\end{aligned}$$

Proof. **Step 1:** Assumption 2(v) allows the descent lemma [53, p. 683] to be applied to $\mathbf{M} \mapsto F(\mathbf{A}, \mathbf{M})$, leading to

$$\begin{aligned}F(\mathbf{A}^{k+1},\mathbf{M}^{k+1}) &\leq F(\mathbf{A}^{k+1}, \mathbf{M}^k) + \frac{L_{\mathbf{M}}^k}{2}\big\|\mathbf{M}^{k+1} - \mathbf{M}^k\big\|^2 \\ &+ \big\langle \nabla_{\mathbf{M}} F(\mathbf{A}^{k+1}, \mathbf{M}^k), \mathbf{M}^{k+1} - \mathbf{M}^k \big\rangle \qquad (21)\end{aligned}$$

Thus,

$$\begin{aligned}\Psi(\mathbf{A}^{k+1},\mathbf{M}^{k+1}) &\leq F(\mathbf{A}^{k+1},\mathbf{M}^{k}) + G(\mathbf{A}^{k+1})\\ &\quad+ \big\langle \nabla_{\mathbf{M}} F(\mathbf{A}^{k+1},\mathbf{M}^{k}), \mathbf{M}^{k+1}-\mathbf{M}^{k}\big\rangle\\ &\quad+ \frac{L^k_{\mathbf{M}}}{2}\big\|\mathbf{M}^{k+1}-\mathbf{M}^{k}\big\|^2 + r(\mathbf{M}^{k+1})\end{aligned}$$

$$\begin{aligned}&= f_{\omega^k}(\mathbf{A}^{k+1}_{\omega^k},\mathbf{M}^{k}) + g_{\omega^k}(\mathbf{A}^{k+1}_{\omega^k}) + \frac{L^k_{\mathbf{M}}}{2}\big\|\mathbf{M}^{k+1}-\mathbf{M}^{k}\big\|^2\\ &\quad+ \sum_{q\neq\omega^k} f_q(\mathbf{A}^k_q,\mathbf{M}^k) + g_q(\mathbf{A}^k_q) + r(\mathbf{M}^{k+1})\\ &\quad+ \big\langle \nabla_{\mathbf{M}} F(\mathbf{A}^{k+1},\mathbf{M}^{k}), \mathbf{M}^{k+1}-\mathbf{M}^{k}\big\rangle. \qquad (22)\end{aligned}$$

Since $\mathbf{M}^{k+1} = \mathbf{M}^k + \gamma^k\big(\hat{\mathbf{M}}^k - \mathbf{M}^k\big)$, we further have

$$\begin{aligned}\Psi(\mathbf{A}^{k+1},\mathbf{M}^{k+1}) &\leq f_{\omega^k}(\mathbf{A}^{k+1}_{\omega^k},\mathbf{M}^{k}) + g_{\omega^k}(\mathbf{A}^{k+1}_{\omega^k})\\ &\quad+ \sum_{q\neq\omega^k} f_q(\mathbf{A}^k_q,\mathbf{M}^k) + g_q(\mathbf{A}^k_q) + r(\mathbf{M}^{k+1})\\ &\quad+ \big\langle \nabla_{\mathbf{M}} F(\mathbf{A}^{k+1},\mathbf{M}^{k}), \mathbf{M}^{k+1}-\mathbf{M}^{k}\big\rangle\\ &\quad+ \frac{(\gamma^k)^2 L^k_{\mathbf{M}}}{2}\big\|\hat{\mathbf{M}}^{k}-\mathbf{M}^{k}\big\|^2. \qquad (23)\end{aligned}$$

In addition, the optimality of $\hat{\mathbf{M}}^k$ implies

$$\begin{aligned}&r(\hat{\mathbf{M}}^k) + \frac{c^k_{\mathbf{M}}}{2}\big\|\hat{\mathbf{M}}^{k}-\mathbf{M}^{k}\big\|^2\\ &\quad+ \big\langle \nabla_{\mathbf{M}} F(\mathbf{A}^{k+1},\mathbf{M}^{k}), \hat{\mathbf{M}}^{k}-\mathbf{M}^{k}\big\rangle \leq r(\mathbf{M}^k)\end{aligned} \qquad (24)$$

and the convexity of r leads to

$$r(\mathbf{M}^{k+1}) \leq r(\mathbf{M}^k) + \gamma_k\big(r(\hat{\mathbf{M}}^k) - r(\mathbf{M}^k)\big). \qquad (25)$$

Combining (25), (24) and exploiting the expression $\mathbf{M}^{k+1} = \mathbf{M}^k + \gamma^k\big(\hat{\mathbf{M}}^k - \mathbf{M}^k\big)$ leads to

$$\begin{aligned}r(\mathbf{M}^{k+1}) &\leq r(\mathbf{M}^k) + \gamma_k\big(r(\hat{\mathbf{M}}^k) - r(\mathbf{M}^k)\big)\\ \text{(from (24))} &\leq r(\mathbf{M}^k) - \frac{\gamma^k c^k_{\mathbf{M}}}{2}\big\|\hat{\mathbf{M}}^{k}-\mathbf{M}^{k}\big\|^2\\ &\quad- \gamma^k\big\langle \nabla_{\mathbf{M}} F(\mathbf{A}^{k+1},\mathbf{M}^{k}), \hat{\mathbf{M}}^{k}-\mathbf{M}^{k}\big\rangle. \qquad (26)\end{aligned}$$

Combining (26) and (23) finally results in

$$\begin{aligned}\Psi(\mathbf{A}^{k+1},\mathbf{M}^{k+1}) &\leq f_{\omega^k}(\mathbf{A}^{k+1}_{\omega^k},\mathbf{M}^{k}) + g_{\omega^k}(\mathbf{A}^{k+1}_{\omega^k})\\ &\quad+ r(\mathbf{M}^k) + \sum_{q\neq\omega^k} f_q(\mathbf{A}^k_q,\mathbf{M}^k) + g_q(\mathbf{A}^k_q)\\ &\quad- \frac{\gamma_k}{2}(c^k_{\mathbf{M}} - \gamma^k L^k_{\mathbf{M}})\big\|\hat{\mathbf{M}}^{k}-\mathbf{M}^{k}\big\|^2. \qquad (27)\end{aligned}$$

Step 2: Arguments similar to those used in Step 1 above lead to

$$\begin{aligned}&f_{\omega^k}(\mathbf{A}^{k+1}_{\omega^k},\mathbf{M}^{k}) + g_{\omega^k}(\mathbf{A}^{k+1}_{\omega^k}) \leq f_{\omega^k}(\mathbf{A}^{k}_{\omega^k},\mathbf{M}^{k})\\ &+ \big\langle \nabla_{\mathbf{A}_{\omega^k}} f_{\omega^k}(\mathbf{A}^{k}_{\omega^k},\mathbf{M}^{k}) - \nabla_{\mathbf{A}_{\omega^k}} f_{\omega^k}(\mathbf{A}^{k}_{\omega^k},\hat{\mathbf{M}}^{k}), \mathbf{A}^{k+1}_{\omega^k} - \mathbf{A}^{k}_{\omega^k}\big\rangle\\ &- \frac{\gamma_k}{2}\big(c^k_{\mathbf{A}_{\omega^k}} - \gamma_k L^k_{\mathbf{A}_{\omega^k}}\big)\big\|\hat{\mathbf{A}}^{k}_{\omega^k} - \mathbf{A}^{k}_{\omega^k}\big\|^2 + g_{\omega^k}(\mathbf{A}^{k}_{\omega^k}).\end{aligned} \qquad (28)$$

Since $\nabla_{\mathbf{A}_\omega} f_\omega(\mathbf{A}_\omega,\cdot)$ is assumed to be Lipschitz continuous (see Assumption 3(ii)), we have

$$\begin{aligned}&\big\langle \nabla_{\mathbf{A}_{\omega^k}} f_{\omega^k}(\mathbf{A}^{k}_{\omega^k},\mathbf{M}^{k}) - \nabla_{\mathbf{A}_{\omega^k}} f_{\omega^k}(\mathbf{A}^{k}_{\omega^k},\hat{\mathbf{M}}^{k}), \mathbf{A}^{k+1}_{\omega^k} - \mathbf{A}^{k}_{\omega^k}\big\rangle\\ &\quad\leq L^k_{\mathbf{A},\mathbf{M}}\big\|\mathbf{M}^k - \hat{\mathbf{M}}^k\big\|\big\|\mathbf{A}^{k+1}_{\omega^k} - \mathbf{A}^{k}_{\omega^k}\big\|\end{aligned}$$

which, combined with (28), leads to

$$\begin{aligned}&f_{\omega^k}(\mathbf{A}^{k+1}_{\omega^k},\mathbf{M}^{k}) + g_{\omega^k}(\mathbf{A}^{k+1}_{\omega^k}) \leq f_{\omega^k}(\mathbf{A}^{k}_{\omega^k},\mathbf{M}^{k})\\ &\quad+ L^k_{\mathbf{A},\mathbf{M}}\big\|\mathbf{M}^k - \hat{\mathbf{M}}^k\big\|\big\|\mathbf{A}^{k+1}_{\omega^k} - \mathbf{A}^{k}_{\omega^k}\big\| + g_{\omega^k}(\mathbf{A}^{k}_{\omega^k})\\ &\quad- \frac{\gamma_k}{2}\big(c^k_{\mathbf{A}_{\omega^k}} - \gamma_k L^k_{\mathbf{A}_{\omega^k}}\big)\big\|\hat{\mathbf{A}}^{k}_{\omega^k} - \mathbf{A}^{k}_{\omega^k}\big\|^2.\end{aligned} \qquad (29)$$

Step 3: From this point, the product involving $\big\|\mathbf{M}^k - \tilde{\mathbf{M}}^k\big\|$ in (29) can be bounded as proposed in [21, Theorem 5.1]. To this end, we first note that

$$\begin{aligned}&L^k_{\mathbf{A},\mathbf{M}}\big\|\mathbf{M}^k - \hat{\mathbf{M}}^k\big\|\big\|\mathbf{A}^{k+1}_{\omega^k} - \mathbf{A}^{k}_{\omega^k}\big\|\\ &\leq \frac{L^k_{\mathbf{A},\mathbf{M}}}{2}\big\|\mathbf{M}^k - \hat{\mathbf{M}}^k\big\|^2 + \frac{L^k_{\mathbf{A},\mathbf{M}}}{2}\big\|\mathbf{A}^{k+1}_{\omega^k} - \mathbf{A}^{k}_{\omega^k}\big\|^2\\ &= \frac{L^k_{\mathbf{A},\mathbf{M}}}{2}\big\|\mathbf{M}^k - \hat{\mathbf{M}}^k\big\|^2 + \frac{L^k_{\mathbf{A},\mathbf{M}}\gamma_k^2}{2}\big\|\hat{\mathbf{A}}^{k}_{\omega^k} - \mathbf{A}^{k}_{\omega^k}\big\|^2\\ &\text{(using } \mathbf{A}^{k+1}_{\omega^k} = \mathbf{A}^{k}_{\omega^k} + \gamma_k(\hat{\mathbf{A}}^{k}_{\omega^k} - \mathbf{A}^{k}_{\omega^k})\text{)}. \qquad (30)\end{aligned}$$

Besides, using the fact that $d^k_{\omega^k} \leq \tau$ for any index k (see Assumption 1), we have

$$\begin{aligned}\big\|\mathbf{M}^k - \tilde{\mathbf{M}}^k\big\|^2 &= \Big\|\sum_{q=k-d^k_{\omega^k}+1}^{k}(\mathbf{M}^q - \mathbf{M}^{q-1})\Big\|^2\\ &\leq \tau\sum_{q=k-\tau+1}^{k}\big\|\mathbf{M}^q - \mathbf{M}^{q-1}\big\|^2. \qquad (31)\end{aligned}$$

Combining (29), (30), and (31) then leads to

$$\begin{aligned}&f_{\omega^k}(\mathbf{A}^{k+1}_{\omega^k},\mathbf{M}^{k}) + g_{\omega^k}(\mathbf{A}^{k+1}_{\omega^k}) \leq f_{\omega^k}(\mathbf{A}^{k}_{\omega^k},\mathbf{M}^{k})\\ &- \frac{\gamma_k}{2}\big(c^k_{\mathbf{A}_{\omega^k}} - \gamma_k(L^k_{\mathbf{A}_{\omega^k}} + L^k_{\mathbf{A},\mathbf{M}})\big)\big\|\hat{\mathbf{A}}^{k}_{\omega^k} - \mathbf{A}^{k}_{\omega^k}\big\|^2\\ &+ \tau L^k_{\mathbf{A},\mathbf{M}}\sum_{q=k-\tau+1}^{k}\big\|\mathbf{M}^q - \mathbf{M}^{q-1}\big\|^2 + g_{\omega^k}(\mathbf{A}^{k}_{\omega^k}).\end{aligned} \qquad (32)$$

Step 4: Combining (27), (32) and using the bounds on the different Lipschitz constants introduced in Assumptions 2(vi) and 3(ii) finally leads to the announced result. □

According to Lemma 1, the objective function Ψ is not necessarily decreasing from an iteration to another due to the presence of a residual term involving τ past estimates of $\mathbf{M}$. From this observation, an auxiliary function (whose derivation is reproduced in Lemma 2 for the sake of completeness) has been proposed in [21]. The introduction of such a function, which is eventually non-increasing between two consecutive iterations, is of particular interest for the convergence analysis. This function finally allows convergence guarantees related to the original problem (7) to be recovered.

Lemma 2 (Auxiliary function definition, adapted from [21, Proof of Theorem 5.1])**.** *Under the same assumptions as in Lemma 1, let Φ be the function defined by*

$$\begin{aligned}\Phi\big(\mathbf{A}(0),\mathbf{M}(0),\mathbf{M}(1),\ldots,\mathbf{M}(\tau)\big) &= \Psi\big(\mathbf{A}(0),\mathbf{M}(0)\big)\\ &\quad+ \frac{\beta}{2}\sum_{q=1}^{\tau}(\tau - q + 1)\big\|\mathbf{M}(q) - \mathbf{M}(q-1)\big\|^2\end{aligned} \qquad (33)$$

with $\beta = \tau L^+_{\mathbf{A},\mathbf{M}}$. Let $\mathbf{w}^k = (\mathbf{A}^k, \mathbf{M}^k, \check{\mathbf{M}}^k)$ and $\check{\mathbf{M}}^k = (\mathbf{M}^{k-1}, \ldots, \mathbf{M}^{k-\tau})$ for any iteration index $k \in \mathbb{N}$ (with the *convention* $\mathbf{M}^q = \mathbf{M}^0$ *if* $q < 0$*). Then,*

$$\begin{aligned}&\Phi(\mathbf{w}^{k+1}) \leq \Phi(\mathbf{w}^k) \\ &- \frac{\gamma_k}{2}\big(c_{\mathbf{A}} - \gamma_k(L^+_{\mathbf{A}} + L^+_{\mathbf{A},\mathbf{M}})\big)\big\|\hat{\mathbf{A}}^k_{\omega^k} - \mathbf{A}^k_{\omega^k}\big\|^2 \\ &- \frac{\gamma_k}{2}\big(c_{\mathbf{M}} - \gamma_k(L^+_{\mathbf{M}} + \tau^2 L^+_{\mathbf{A},\mathbf{M}})\big)\big\|\hat{\mathbf{M}}^k - \mathbf{M}^k\big\|^2. \end{aligned} \tag{34}$$

Proof. The expression of the auxiliary function proposed in [21] results from the following decomposition of the residual term $\sum_{q=k-\tau+1}^{k}\big\|\mathbf{M}^q - \mathbf{M}^{q-1}\big\|^2$. Introducing the auxiliary variables

$$\alpha^k = \sum_{q=k-\tau+1}^{k} (q-k+\tau)\big\|\mathbf{M}^q - \mathbf{M}^{q-1}\big\|^2$$

we can note that

$$\alpha^k - \alpha^{k+1} = \sum_{q=k-\tau+1}^{k} \big\|\mathbf{M}^q - \mathbf{M}^{q-1}\big\|^2 - \tau\big\|\mathbf{M}^{k+1} - \mathbf{M}^k\big\|^2. \tag{35}$$

Thus, using the upper bound $L^k_{\mathbf{A},\mathbf{M}} \leq L^+_{\mathbf{A},\mathbf{M}}$ (Assumption 3(ii)) and replacing (35) in (20) yields

$$\begin{aligned}&\Psi(\mathbf{A}^{k+1}, \mathbf{M}^{k+1}) + \beta\alpha^{k+1} \leq \Psi(\mathbf{A}^k, \mathbf{M}^k) + \beta\alpha^k \\ &- \frac{\gamma_k}{2}\big(c_{\mathbf{A}} - \gamma_k(L^+_{\mathbf{A}} + L^+_{\mathbf{A},\mathbf{M}})\big)\big\|\hat{\mathbf{A}}^k_{\omega^k} - \mathbf{A}^k_{\omega^k}\big\|^2 \\ &- \frac{\gamma_k}{2}\big(c_{\mathbf{M}} - \gamma_k(L^+_{\mathbf{M}} + \tau^2 L^+_{\mathbf{A},\mathbf{M}})\big)\big\|\hat{\mathbf{M}}^k - \mathbf{M}^k\big\|^2. \end{aligned}$$

Observing that $\Phi(\mathbf{w}^k) = \Psi(\mathbf{A}^k, \mathbf{M}^k) + \alpha^k$ finally leads to the announced result. □

The previous lemma makes clear that the proposed algorithm can be studied as a special case of [22]. The rest of the convergence analysis, which involves somewhat convoluted arguments, exactly follows [22] up to minor notational modifications.

References

[1] J. M. Bioucas-Dias, A. Plaza, N. Dobigeon, M. Parente, Q. Du, P. Gader, and J. Chanussot, "Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches," *IEEE J. Sel. Topics Appl. Earth Observ. in Remote Sens.*, vol. 5, no. 2, pp. 354–379, Apr. 2012.

[2] S. A. Robila and D. Ricart, "Distributed algorithms for unmixing hyperspectral data using nonnegative matrix factorization with sparsity constraints," in *Proc. IEEE Int. Conf. Geosci. Remote Sens. (IGARSS)*, Melbourne, Australia, Jul. 2013, pp. 2156–2159.

[3] J. Sigurdsson, M. O. Ulfarsson, J. R. Sveinsson, and J. M. Bioucas-Dias, "Sparse distributed hyperspectral unmixing," in *Proc. IEEE Int. Conf. Geosci. Remote Sens. (IGARSS)*, Beijing, China, Jul. 2016, pp. 6994–6997.

[4] ——, "Sparse distributed multitemporal hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 11, pp. 6069–6084, Nov. 2017.

[5] C. G. Tsinos, A. A. Rontogiannis, and K. Berberidis, "Distributed blind hyperspectral unmixing via joint sparsity and low-rank constrained non-negative matrix factorization," *IEEE Trans. Comput. Imag.*, vol. 3, no. 2, pp. 160–174, Apr. 2017.

[6] J. Bolte, S. Sabach, and M. Teboulle, "Proximal alternating linearized minimization for nonconvex and nonsmooth problems," *Mathematical Programming*, vol. 1-2, no. 146, pp. 459–494, Jul. 2013.

[7] E. Chouzenoux, J.-C. Pesquet, and A. Repetti, "A block coordinate variable metric forward-backward algorithm," *J. Glob. Optim*, vol. 66, no. 3, pp. 547–485, 2016.

[8] A. Repetti, E. Chouzenoux, and J.-C. Pesquet, "A preconditioned forward-backward approach with application to large-scale nonconvex spectral unmixing problems," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Processing (ICASSP)*, Florence, Italy, May 2014, pp. 1498–1502.

[9] J. Li, J. M. Bioucas-Dias, A. Plaza, and L. Liu, "Robust collaborative nonnegative matrix factorization for hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 10, pp. 6076–6090, Oct. 2016.

[10] P.-A. Thouvenin, N. Dobigeon, and J.-Y. Tourneret, "Estimation de variabilité pour le démélange non-supervisé d'images hyperspectrales," in *Actes du XXVième Colloque GRETSI*, Lyon, France, Sep. 2015, in French.

[11] P. Bianchi and J. Jakubowicz, "Convergence of a multi-agent projected stochastic gradient algorithm for non-convex optimization," *IEEE Trans. Autom. Control*, vol. 58, no. 2, pp. 391–405, Feb. 2013.

[12] J. Liang, M. Zhang, X. Zeng, and G. Yu, "Distributed dictionary learning for sparse representation in sensor networks," *IEEE Trans. Image Process.*, vol. 23, no. 6, pp. 2528–2541, Jun. 2014.

[13] J. Chen, Z. J. Towfic, and A. H. Sayed, "Dictionary learning over distributed models," *IEEE Trans. Signal Process.*, vol. 63, no. 4, pp. 1001–1016, Feb. 2015.

[14] P. D. Lorenzo and G. Scutari, "Distributed nonconvex optimization over networks," in *Proc. IEEE Int. Workshop Comput. Adv. Multi-Sensor Adaptive Process. (CAMSAP)*, Cancun, Mexico, Dec. 2015, pp. 229–232.

[15] F. Facchinei, G. Scutari, and S. Sagratella, "Parallel selective algorithms for nonconvex big data optimization," *IEEE Trans. Signal Process.*, vol. 63, no. 7, pp. 1874–1889, Apr. 2015.

[16] G. Scutari, F. Facchinei, L. Lampariello, and P. Song, "Parallel and distributed methods for nonconvex optimization-part I: Theory," *IEEE Trans. Signal Process.*, vol. 65, no. 8, pp. 1929–2944, Apr. 2017.

[17] Y. Yang, G. Scutari, D. P. Palomar, and M. Pesavento, "A parallel decomposition method for nonconvex stochastic multi-agent optimization problems," *IEEE Trans. Signal Process.*, vol. 64, no. 11, pp. 2949–2964, Jun. 2016.

[18] J.-C. Pesquet and A. Repetti, "A class of randomized primal-dual algorithms for distributed optimization," *Journal of nonlinear and convex analysis*, vol. 16, no. 12, pp. 2453–2490, Nov. 2015.

[19] P. L. Combettes and J. Eckstein, "Asynchronous block-iterative primal-dual decomposition methods for monotone inclusions," *Math. Program., Ser. B*, pp. 1–28, 2016.

[20] Z. Peng, Y. Xu, M. Yan, and W. Yin, "Arock: an algorithmic framework for asynchronous parallel coordinate updates," *SIAM J. Sci. Comput.*, vol. 38, no. 5, pp. 2851–2879, Sep. 2016.

[21] D. Davis, "The asynchronous PALM algorithm for nonsmooth nonconvex problems," Apr. 2016, submitted. [Online]. Available: https://arxiv.org/abs/1604.00526

[22] L. Cannelli, F. Facchinei, V. Kungurtsev, and G. Scutari, "Asynchronous parallel algorithms for nonconvex big-data optimization. Part I: Model and convergence," Jul. 2016, arXiv preprint. [Online]. Available: https://arxiv.org/abs/1607.04818

[23] P. Bianchi, W. Hachem, and F. Iutzeler, "A coordinate descent primal-dual algorithm and application to distributed asynchronous optimization," *IEEE Trans. Autom. Control*, vol. 61, no. 10, p. 2947, Oct. 2016.

[24] T.-H. Chang, M. Hong, W.-C. Liao, and X. Wang, "Asynchronous distributed ADMM for large-scale optimization–part I: Algorithm and convergence analysis," *IEEE Trans. Signal Process.*, vol. 62, no. 12, pp. 3118–3130, Jun. 2016.

[25] X. Lian, Y. Huang, Y. Li, and J. Liu, "Asynchronous parallel stochastic gradient for nonconvex optimization," in *Adv. in Neural Information Processing Systems*, Montréal, Canada, Dec. 2015, pp. 2719–2727.

[26] S. J. Wright, "Coordinate descent algorithms," *Math. Program., Ser. B*, vol. 151, no. 1, pp. 3–34, Jun. 2015.

[27] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Machine Learning*, vol. 3, no. 1, pp. 1–122, 2010.

[28] Y. Wang, W. Yin, and J. Zeng, "Global convergence of admm in nonconvex nonsmooth optimization," *UCLA CAM Report 15-62*, Nov. 2015. [Online]. Available: https://arxiv.org/abs/1511.06324

[29] P. Frankel, G. Garrigos, and J. Peypouquet, "Splitting methods with variable metric for Kurdyka-Łojasiewicz functions and general convergence rates," *J. Optim. Theory Appl.*, vol. 165, no. 1, pp. 874–900, 2015.

[30] S. Henrot, J. Chanussot, and C. Jutten, "Dynamical spectral unmixing of multitemporal hyperspectral images," *IEEE Trans. Image Process.*, vol. 25, no. 7, pp. 3219–3232, Jul. 2016.

[31] P.-A. Thouvenin, N. Dobigeon, and J.-Y. Tourneret, "Online unmixing of multitemporal hyperspectral images accounting for spectral variability," *IEEE Trans. Image Process.*, vol. 25, no. 9, pp. 3979–3990, Sep. 2016.

[32] N. Yokoya, X. X. Zhu, and A. Plaza, "Multisensor Coupled Spectral Unmixing for Time-Series Analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 5, pp. 2842–2857, May 2017.

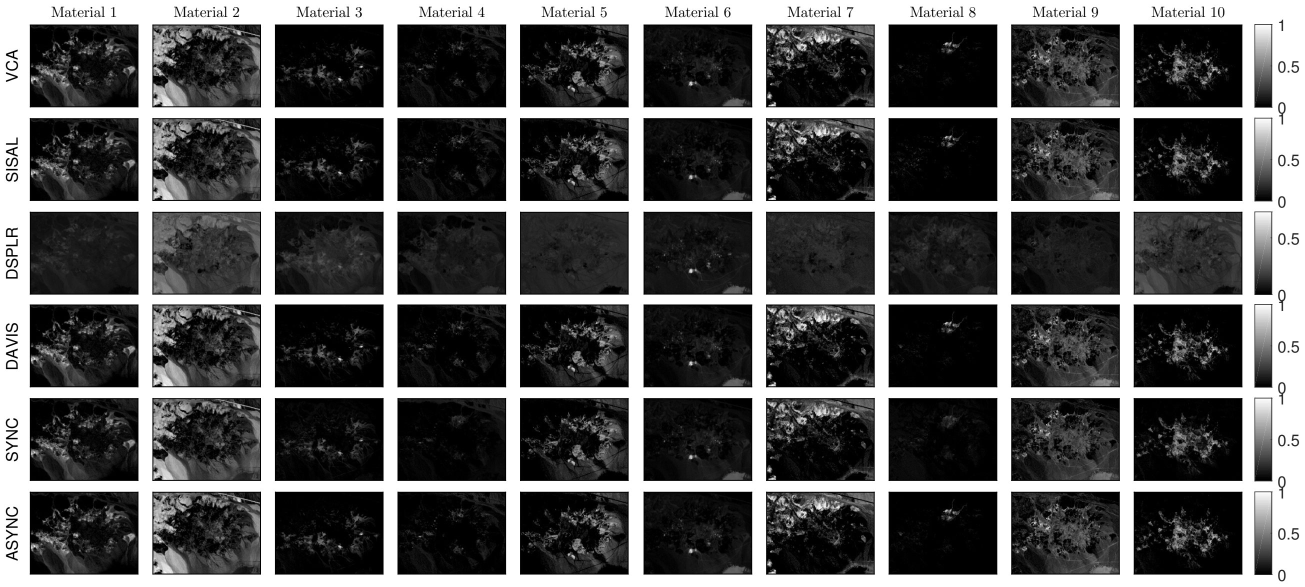


Fig. 6. Abundance maps recovered by the different methods (in each row) for the Cuprite dataset.

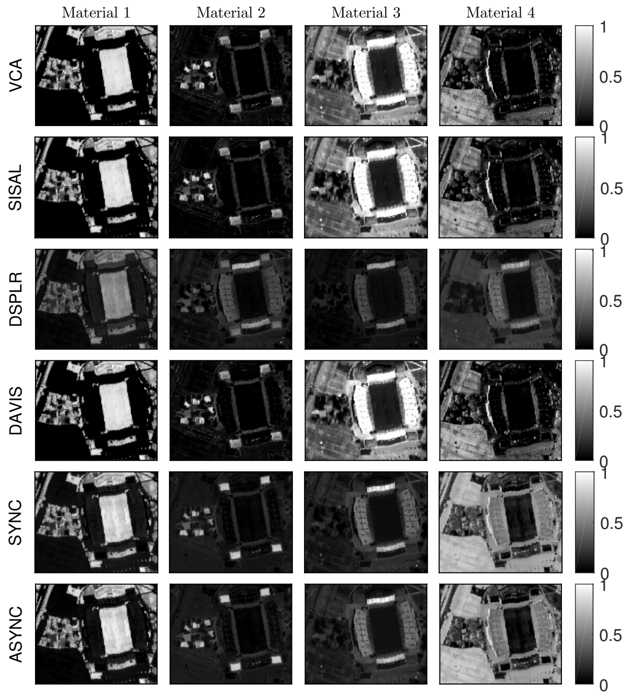


Fig. 7. Abundance maps recovered by the different methods (in each row) for the Houston dataset.

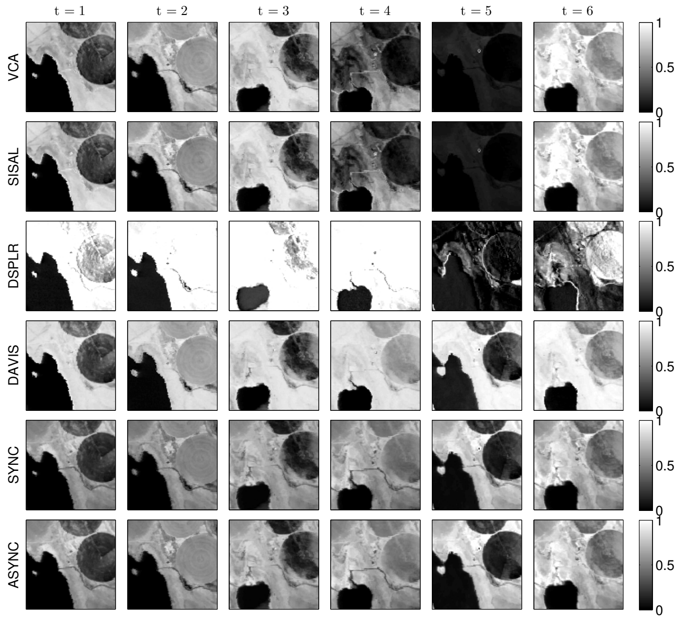


Fig. 8. Soil abundance map recovered by the different methods (in each row) at each time instant (given in column) for the experiment on the Mud lake dataset [the different rows correspond to VCA/FCLS, SISAL/FCLS, DSPLR [5], ASYNC [21], SYNC and ASYNC methods].

[33] C. Févotte, "Bayesian audio source separation," in *Blind speech separation*, S. Makino, T.-W. Lee, and H. Sawada, Eds. Dordrecht, The Netherlands: Springer, 2007, pp. 305–335.

[34] V. Y. F. Tan and C. Févotte, "Automatic relevance determination in nonnegative matrix factorization with the beta-divergence," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 7, pp. 1592–1605, 2013.

[35] B. Gao, W. L. Woo, and B. W.-K. Ling, "Machine learning source separation using maximum a posteriori nonnegative matrix factorization," *IEEE Trans. Comput.*, vol. 44, no. 7, pp. 1169–1179, Aug. 2014.

[36] P. L. Combettes and J.-C. Pesquet, "Proximal splitting methods in signal processing," *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, vol. 49, pp. 185–212, May 2011.

[37] J. Duchi, S. Shalev-Schwartz, Y. Singer, and T. Chandra, "Efficient projection onto the ℓ_1-ball for learning in high dimensions," in *Proc. Int. Conf. Machine Learning (ICML)*, Helsinki, Finland, 2008.

[38] L. Condat, "Fast projection onto the simplex and the ℓ_1 ball," *Math. Program., Ser. A*, pp. 1–11, Sep. 2015. [Online]. Available: http://dx.doi.org/10.1007/s10107-015-0946-6

[39] M. Berman, H. Kiiveri, R. Lagerstrom, A. Ernst, R. Dunne, and J. F. Huntington, "ICE: A statistical approach to identifying endmembers in hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 10, pp. 2085–2095, Oct. 2004.

[40] P.-A. Thouvenin, N. Dobigeon, and J.-Y. Tourneret, "A hierarchical Bayesian model accounting for endmember variability and abrupt spectral changes to unmix multitemporal hyperspectral images," *IEEE Trans.*

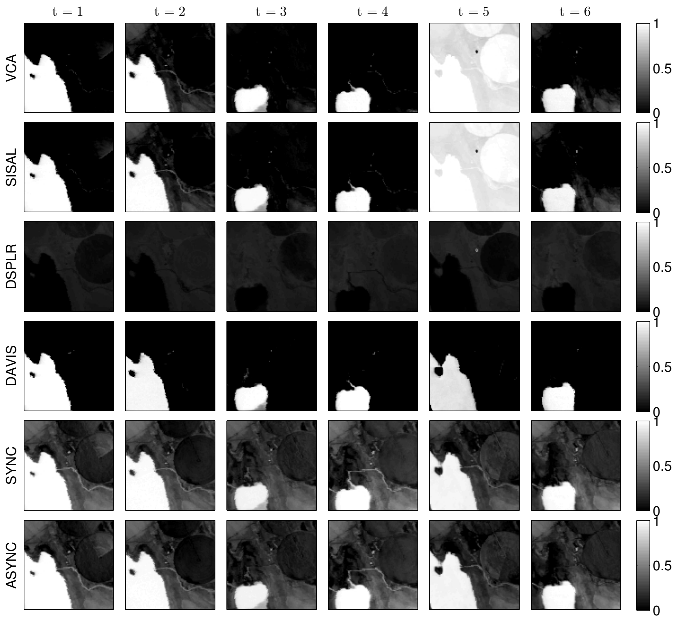


Fig. 9. Water abundance map recovered by the different methods (in each row) at each time instant (given in column) for the experiment on the Mud lake dataset [the different rows correspond to VCA/FCLS, SISAL/FCLS, DSPLR [5], ASYNC [21], SYNC and ASYNC methods].

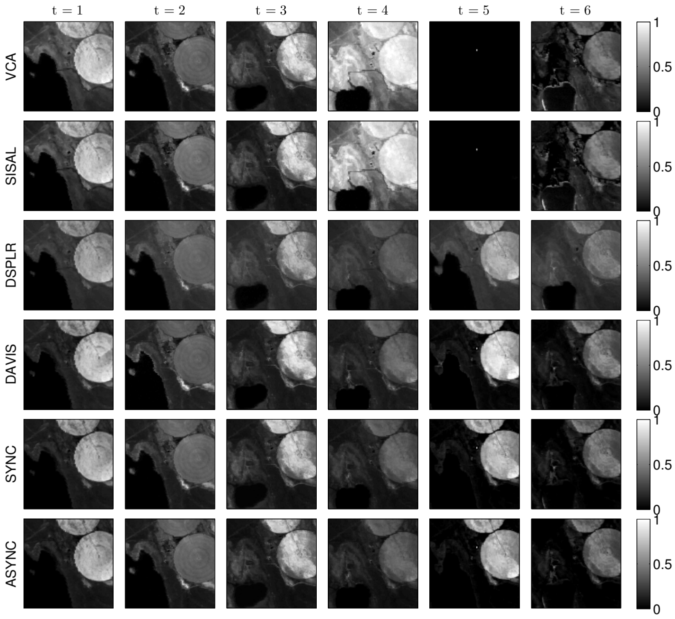


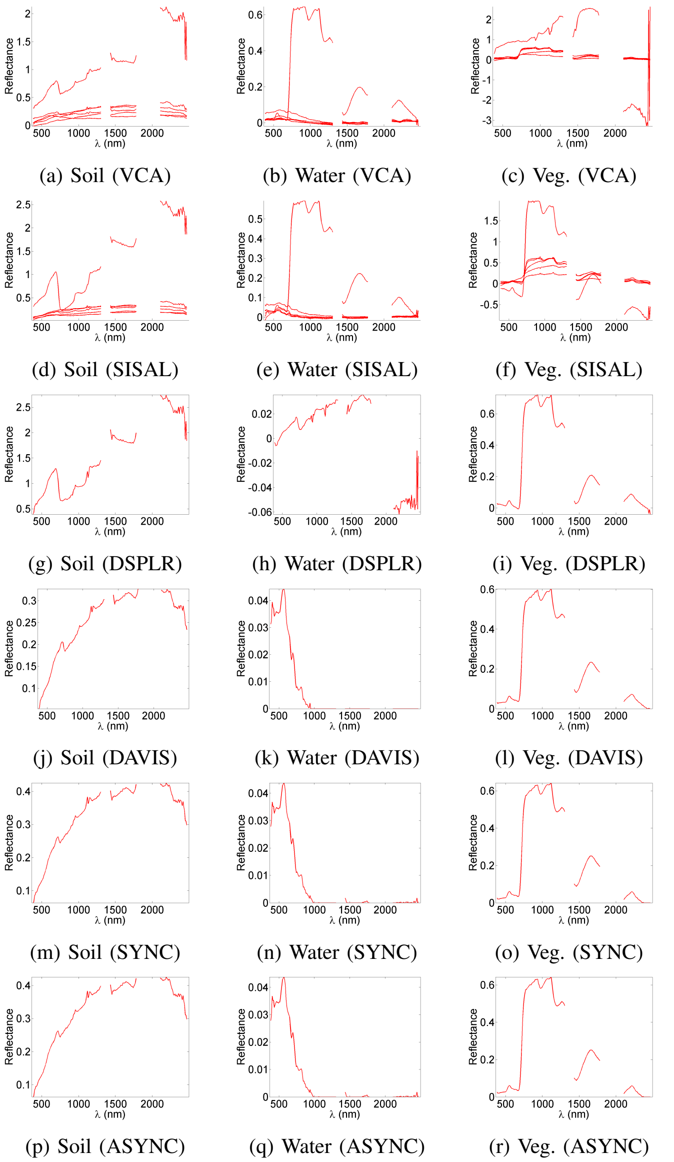
Fig. 10. Vegetation abundance map recovered by the different methods (in each row) at each time instant (given in column) for the experiment on the Mud lake dataset [the different rows correspond to VCA/FCLS, SISAL/FCLS, DSPLR [5], ASYNC [21], SYNC and ASYNC methods].

Comput. Imag., vol. 4, no. 1, pp. 32–45, Mar. 2018.

[41] N. Dobigeon, S. Moussaoui, M. Coulon, J.-Y. Tourneret, and A. O. Hero, "Joint Bayesian endmember extraction and linear unmixing for hyperspectral imagery," *IEEE Trans. Signal Process.*, vol. 57, no. 11, pp. 4355–4368, Nov. 2009.

[42] J. Plaza, E. M. Hendrix, I. Garca, G. Martn, and A. Plaza, "On endmember identification in hyperspectral images without pure pixels: A comparison of algorithms," *J. Math. Imag. Vision*, vol. 42, pp. 163–175, Mar. 2011.

[43] J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah, "Julia: A fresh approach to numerical computing," *SIAM Review*, vol. 59, no. 1, pp. 65–98, 2017.



(a) Soil (VCA) (b) Water (VCA) (c) Veg. (VCA)

(d) Soil (SISAL) (e) Water (SISAL) (f) Veg. (SISAL)

(g) Soil (DSPLR) (h) Water (DSPLR) (i) Veg. (DSPLR)

(j) Soil (DAVIS) (k) Water (DAVIS) (l) Veg. (DAVIS)

(m) Soil (SYNC) (n) Water (SYNC) (o) Veg. (SYNC)

(p) Soil (ASYNC) (q) Water (ASYNC) (r) Veg. (ASYNC)

Fig. 11. Endmembers ($\mathbf{m}_r$, red lines) recovered by the different methods from the real dataset depicted in Fig. 5. Endmembers extracted by VCA, SISAL and DSPLR show a notable sensitivity to the presence of outliers in these data,

[44] J. M. Nascimento and J. M. Bioucas-Dias, "Vertex component analysis: a fast algorithm to unmix hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 4, pp. 898–910, Apr. 2005.

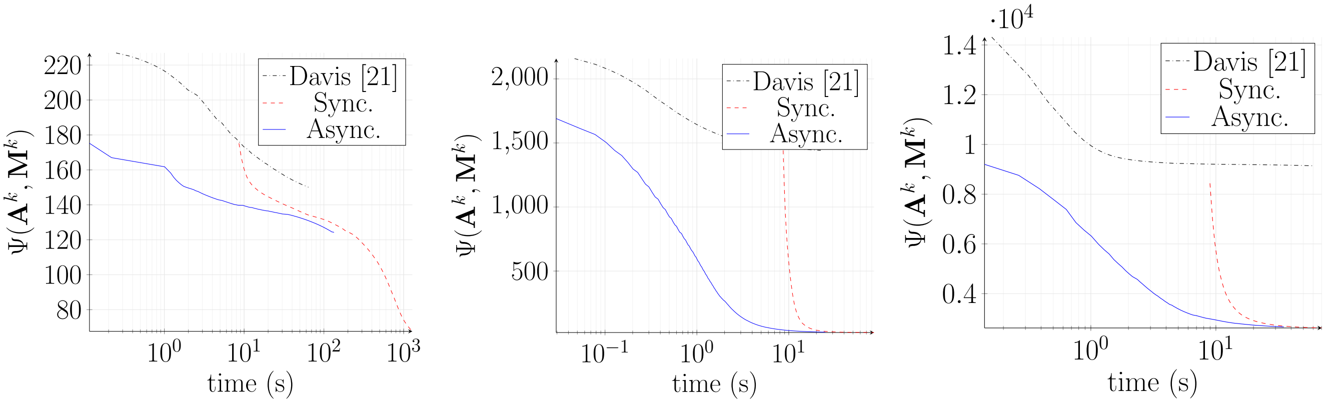
[45] J. M. Bioucas-Dias and M. A. T. Figueiredo, "Alternating direction algorithms for constrained sparse regression: Application to hyperspectral unmixing," in *Proc. IEEE GRSS Workshop Hyperspectral Image Signal Process.: Evolution in Remote Sens. (WHISPERS)*, Reykjavik, Iceland, Jun. 2010, pp. 1–4.

[46] J. M. Bioucas-Dias, "A variable splitting augmented Lagrangian approach to linear spectral unmixing," in *Proc. IEEE GRSS Workshop Hyperspectral Image Signal Process.: Evolution in Remote Sens. (WHISPERS)*, Grenoble, France, Aug. 2009, pp. 1–4.

[47] L. Miao and H. Qi, "Endmember extraction from highly mixed data using minimum volume constrained nonnegative matrix factorization," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 3, pp. 765–776, Mar. 2007.

[48] C. Debes, A. Merentitis, R. Heremans, J. Hahn, N. Frangiadakis, T. van Kasteren, W. Liao, R. Bellens, A. Piurica, S. Gautama, W. Philips, S. Prasad, Q. Du, and F. Pacifici, "Hyperspectral and lidar data fusion: Outcome of the 2013 grss data fusion contest," *IEEE J. Sel. Topics Appl. Earth Observ. in Remote Sens.*, vol. 7, no. 6, pp. 2405–2418, Jun. 2014.

[49] L. Drumetz, M.-A. Veganzones, S. Henrot, R. Phlypo, J. Chanussot, and C. Jutten, "Blind hyperspectral unmixing using an extended linear mix-



(a) Cuprite

(b) Houston

(c) Mud lake

Fig. 12. Evolution of the objective function for the synthetic datasets, obtained for DAVIS [21], Algo. 1 and its synchronous version until convergence.

ing model to address spectral variability," *IEEE Trans. Image Process.*, vol. 25, no. 8, pp. 3890–3905, Aug. 2016.

[50] P.-A. Thouvenin, N. Dobigeon, and J.-Y. Tourneret, "Online unmixing of multitemporal hyperspectral images accounting for spectral variability - complementary results and supporting material," University of Toulouse, IRIT/INP-ENSEEIHT, Tech. Rep., Oct. 2015. [Online]. Available: http://thouvenin.perso.enseeiht.fr/papers/Thouvenin_TR_2015.pdf

[51] A. Halimi, P. Honeine, M. Kharouf, C. Richard, and J.-Y. Tourneret, "Estimating the intrinsic dimension of hyperspectral images using a noise-whitened eigen-gap approach," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 7, pp. 3811–3821, Jul. 2016.

[52] E. Chouzenoux, J.-C. Pesquet, and A. Repetti, "Variable metric forward–backward algorithm for minimizing the sum of a differentiable function and a convex function," *J. Optim. Theory Appl.*, vol. 162, no. 1, pp. 107–132, 2014.

[53] D. P. Bertsekas, *Nonlinear Programming*. Belmont, MA: Athena Scientific, 1999.

Pierre-Antoine Thouvenin (S'15–M'17) received the state engineering degree in electrical engineering from ENSEEIHT, Toulouse, France, and the M.Sc. degree in signal processing from the National Polytechnic Institute of Toulouse (INP Toulouse), both in 2014, as well as the PhD degree in Signal Processing from the INP Toulouse in 2017. Since September 2017, he has been working as a post-doctoral research associate within the Biomedical and Astronomical Signal Processing (BASP) group, Heriot-Watt University, Edinburgh, UK. His research interests include statistical modeling, optimization techniques and hyperspectral unmixing.

Nicolas Dobigeon (S'05–M'08–SM'13) received the state engineering degree in electrical engineering from ENSEEIHT, Toulouse, France, and the M.Sc. degree in signal processing from the National Polytechnic Institute of Toulouse (INP Toulouse), both in June 2004, as well as the Ph.D. degree and Habilitation à Diriger des Recherches in Signal Processing from the INP Toulouse in 2007 and 2012, respectively. He was a Post-Doctoral Research Associate with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI, USA, from 2007 to 2008.

Since 2008, he has been with the National Polytechnic Institute of Toulouse (INP-ENSEEIHT, University of Toulouse) where he is currently a Professor. He conducts his research within the Signal and Communications Group of the IRIT Laboratory and he is also an affiliated faculty member of the Telecommunications for Space and Aeronautics (TéSA) cooperative laboratory. His current research interests include statistical signal and image processing, with a particular interest in Bayesian inverse problems with applications to remote sensing, biomedical imaging and genomics.

Jean-Yves Tourneret (SM'08) received the ingénieur degree in electrical engineering from the Ecole Nationale Supérieure d'Electronique, d'Electrotechnique, d'Informatique, d'Hydraulique et des Télécommunications (ENSEEIHT) de Toulouse in 1989 and the Ph.D. degree from the National Polytechnic Institute from Toulouse in 1992. He is currently a professor in the university of Toulouse (ENSEEIHT) and a member of the IRIT laboratory (UMR 5505 of the CNRS). His research activities are centered around statistical signal and image processing with a particular interest to Bayesian and Markov chain Monte-Carlo (MCMC) methods. He has been involved in the organization of several conferences including the European conference on signal processing EUSIPCO'02 (program chair), the international conference ICASSP'06 (plenaries), the statistical signal processing workshop SSP'12 (international liaisons), the International Workshop on Computational Advances in Multi-Sensor Adaptive Processing CAMSAP 2013 (local arrangements), the statistical signal processing workshop SSP'2014 (special sessions), the workshop on machine learning for signal processing MLSP'2014 (special sessions). He has been the general chair of the CIMI workshop on optimization and statistics in image processing hold in Toulouse in 2013 (with F. Malgouyres and D. Kouamé) and of the International Workshop on Computational Advances in Multi-Sensor Adaptive Processing CAMSAP 2015 (with P. Djuric). He has been a member of different technical committees including the Signal Processing Theory and Methods (SPTM) committee of the IEEE Signal Processing Society (2001-2007, 2010-present). He has been serving as an associate editor for the IEEE Transactions on Signal Processing (2008-2011, 2015-present) and for the EURASIP journal on Signal Processing (2013-present).