Unsupervised Post-Nonlinear Unmixing of Hyperspectral Images Using a Hamiltonian Monte Carlo Algorithm

Yoann Altmann, Nicolas Dobigeon, Jean-Yves Tourneret

Unsupervised Post-Nonlinear Unmixing of Hyperspectral Images Using a Hamiltonian Monte Carlo Algorithm

Yoann Altmann, *Member, IEEE*, Nicolas Dobigeon, *Senior Member, IEEE*, and Jean-Yves Tourneret

***Abstract*— This paper presents a nonlinear mixing model for hyperspectral image unmixing. The proposed model assumes that the pixel reflectances are post-nonlinear functions of unknown pure spectral components contaminated by an additive white Gaussian noise. These nonlinear functions are approximated using second-order polynomials leading to a polynomial post-nonlinear mixing model. A Bayesian algorithm is proposed to estimate the parameters involved in the model yielding an unsupervised nonlinear unmixing algorithm. Due to the large number of parameters to be estimated, an efficient Hamiltonian Monte Carlo algorithm is investigated. The classical leapfrog steps of this algorithm are modified to handle the parameter constraints. The performance of the unmixing strategy, including convergence and parameter tuning, is first evaluated on synthetic data. Simulations conducted with real data finally show the accuracy of the proposed unmixing strategy for the analysis of hyperspectral images.**

***Index Terms*— Hyperspectral imagery, unsupervised spectral unmixing, Hamiltonian Monte Carlo, post-nonlinear model.**

I. Introduction

IDENTIFYING macroscopic materials and quantifying the proportions of these materials are major issues when analyzing hyperspectral images. This blind source separation problem, also referred to as unsupervised spectral unmixing (SU), has been widely studied for the applications where the pixel reflectances are linear combinations of pure component spectra [1]–[5]. However, as explained in [6] and [7], the linear mixing model (LMM) can be inappropriate for some hyperspectral images, such as those containing sand, trees or vegetation. Nonlinear mixing models (NLMMs) provide an interesting alternative for overcoming the inherent limitations of the LMM. They have been proposed in the hyperspectral image literature and cfan be divided into two main classes.

The first class of NLMMs consists of physical models based on the nature of the environment. These models include the bidirectional reflectance based model proposed in [8] for intimate mixtures associated with sand-like materials, combinations of LMM and intimate mixture models [9] and the bilinear/polynomial models studied in [10]–[17] to account for scattering effects mainly observed in vegetation and urban areas. The second class of NLMMs contains more flexible models allowing different kinds of nonlinearities to be approximated. Precisely, these analytical models are not explicitly based on the physical phenomena involved in the mixing process but are able to model different deviations from the classical LMM. Such deviations can result, for instance, from the presence of relief or multi-layered materials, from illumination heterogeneity, or from the spectral variability of the scene components. These flexible models can be constructed from neural networks [18], [19], kernels [20]–[22], or post-nonlinear transformations [23]–[25] (The reader is invited to consult [26] for a recent review). In particular, a polynomial post-nonlinear mixing model (PPNMM) has recently shown interesting properties for the SU of hyperspectral images [27]. This model assumes that the observed pixels result from nonlinear transformations applied to linear combinations of endmembers. The nonlinearities are approximated by second-order polynomials. It has been shown in [27] that the PPNMM is a flexible model (in terms of pixel reconstruction) that can provide accurate abundance estimates for nonlinear unmixing.

Most nonlinear unmixing strategies available in the literature are supervised, i.e., the endmembers contained in the image are assumed to be known (chosen from a spectral library or extracted from the data by an endmember extraction algorithm (EEA)). Moreover, most existing EEAs rely on the LMM [28]–[30] and thus can be inaccurate for nonlinear mixtures. Recently, a nonlinear EEA based on the approximation of geodesic distances has been proposed in [31] to extract endmembers from the data. However, this algorithm can suffer from the absence of pure pixels in the image (as most linear EEAs). This paper presents a new fully unsupervised Bayesian unmixing algorithm based on the PPNMM studied in [27]. The proposed method allows a joint estimation of the endmembers and abundances (mixing coefficients) and does not assume the presence of pure pixels in the observed image. In the Bayesian framework, appropriate prior distributions are chosen for the unknown PPNMM parameters, i.e., the endmembers, the abundances, the nonlinearity parameters and the noise variances. The joint posterior distribution of these parameters is then derived. Since the classical Bayesian estimators cannot

This work was supported in part by the Direction Générale de l'armement, French Ministry of Defence, in part by the HYPANEMA ANR Project under Grant ANR-12-BS03-003, and in part by the Thematic Trimester on Image Processing of the CIMI Labex, Toulouse, France, under Grant ANR-11-LABX-0040-CIMI within the Program ANR-11-IDEX-0002-02. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Jong Chul Ye.

The authors are with the University of Toulouse, Toulouse Cedex 7 31071, France (e-mail: yoann.altmann@enseeiht.fr; nicolas.dobigeon@enseeiht.fr; jean-yves.tourneret@enseeiht.fr).

Digital Object Identifier 10.1109/TIP.2014.2314022

be easily computed from this joint posterior we investigate a Markov chain Monte Carlo (MCMC) method to generate samples according to this posterior. More precisely, following the principles of the Gibbs sampler, samples are generated according to the conditional distributions of the posterior. Due to the large number of parameters to be estimated we propose to use a Hamiltonian Monte Carlo (HMC) [32] method to sample according to some of the conditional distributions. HMCs are powerful simulation strategies based on Hamiltonian dynamics which can improve the convergence and mixing properties of classical MCMC methods (such as the Gibbs sampler and the Metropolis-Hastings algorithm) [33], [34]. These methods have received growing interest in many applications, especially when the number of parameters to be estimated is large [35], [36]. The classical HMC can only be used for unconstrained variables. However, new HMC methods have been recently proposed to handle constrained variables [33, Chap. 5] [37], [38] which allow HMCs to sample according to the posterior of the Bayesian model proposed for SU. Finally, as in any MCMC method, the generated samples are used to compute Bayesian estimators as well as measures of uncertainties such as confidence intervals.

The problem addressed in this paper is the unsupervised nonlinear unmixing of hyperspectral images. The main contribution of this paper is a Bayesian approach which consists of estimating jointly the endmembers and the abundances using the PPNMM. Appropriate prior distributions are assigned to the unknown parameters. In particular, sparsity promoting priors are considered for the nonlinearity parameters. To handle the large number of parameters to be sampled, an efficient constrained HMC method is used, leading to an efficient sampling procedure.

The paper is organized as follows. Section II introduces the PPNMM for hyperspectral image analysis. Section III presents the hierarchical Bayesian model associated with the proposed PPNMM and its posterior distribution. The constrained HMC (CHMC) algorithm used to sample some parameters of this posterior is described in Section IV. The CHMC is coupled with a standard Gibbs sampler presented in Section V. Some simulation results conducted on synthetic and real data are shown and discussed in Sections VI and VII. Conclusions are finally reported in Section VIII.

II. Problem Formulation

A. Polynomial Post-Nonlinear Mixing Model

This section recalls the nonlinear mixing model used in [27] for supervised SU of hyperspectral images. We consider a set of N observed spectra $\mathbf{y}_n = [y_{n,1}, \ldots, y_{n,L}]^T$, $n \in \{1, \ldots, N\}$ where L is the number of spectral bands. Each of these spectra is defined as a nonlinear transformation $\boldsymbol{g}_n$ of a linear mixture of R spectra $\mathbf{m}_r$ contaminated by additive noise

$$\mathbf{y}_n = \boldsymbol{g}_n\left(\sum_{r=1}^{R} a_{r,n}\mathbf{m}_r\right) + \mathbf{e}_n = \boldsymbol{g}_n\left(\mathbf{M}\boldsymbol{a}_n\right) + \mathbf{e}_n \qquad (1)$$

where $\mathbf{m}_r = [m_{r,1}, \ldots, m_{r,L}]^T$ is the spectrum of the rth material present in the scene, $a_{r,n}$ is its corresponding proportion in the nth pixel, R is the number of endmembers contained in the image and $\boldsymbol{g}_n$ is a nonlinear function associated with the nth pixel. Moreover, $\mathbf{e}_n$ is an additive independently distributed zero-mean Gaussian noise sequence with diagonal covariance matrix $\boldsymbol{\Sigma} = \text{diag}\left(\boldsymbol{\sigma}^2\right)$, denoted as $\mathbf{e}_n \sim \mathcal{N}\left(\mathbf{0}_L, \boldsymbol{\Sigma}\right)$, where $\boldsymbol{\sigma}^2 = [\sigma_1^2, \ldots, \sigma_L^2]^T$ is the vector of the L noise variances and $\text{diag}\left(\boldsymbol{\sigma}^2\right)$ is an $L \times L$ diagonal matrix containing the elements of the vector $\boldsymbol{\sigma}^2$. Note that the usual matrix and vector notations $\mathbf{M} = [\mathbf{m}_1, \ldots, \mathbf{m}_R]$ and $\boldsymbol{a}_n = [a_{1,n}, \ldots, a_{R,n}]^T$ have been used in the right hand side of (1). As in [27], the N nonlinear functions $\boldsymbol{g}_n$ are defined as second order polynomial nonlinearities defined by

$$\begin{aligned} \boldsymbol{g}_n : \quad [0,1]^L &\to \mathbb{R}^L \\ \mathbf{s} &\mapsto \left[s_1 + b_n s_1^2, \ldots, s_L + b_n s_L^2\right]^T \end{aligned} \qquad (2)$$

with $\mathbf{s} = [s_1, \ldots, s_L]^T$ and b_n is a real parameter. Motivations for considering polynomial nonlinearities have been discussed in [27]. In particular, it has been shown that the PPNMM involves bilinear and quadratic terms (with respect to the endmembers) which have been considered to handle multiple scattering effects [14], [15]. Thus, it is very flexible and can approximate many different nonlinearities. Because the nonlinearity is characterized by a single nonlinearity parameter per pixel, it is difficult to infer the sources of nonlinearities that can occur in the image pixels using the PPNMM. However, it allows linearly/nonlinearly mixed regions in the image to be identified (as will be shown in Section VII). Straightforward computations allow the PPNMM observation matrix to be expressed as follows

$$\mathbf{Y} = \mathbf{MA} + \left[(\mathbf{MA}) \odot (\mathbf{MA})\right] \text{diag}\left(\boldsymbol{b}\right) + \mathbf{E} \qquad (3)$$

where $\mathbf{A} = [\boldsymbol{a}_1, \ldots, \boldsymbol{a}_N]$ is an $R \times N$ matrix, $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_N]$ and $\mathbf{E} = [\mathbf{e}_1, \ldots, \mathbf{e}_N]$ are $L \times N$ matrices, $\boldsymbol{b} = [b_1, \ldots, b_N]^T$ is an $N \times 1$ vector containing the nonlinearity parameters and $\odot$ denotes the Hadamard (termwise) product.

B. Abundance Reparametrization

Due to physical considerations, the abundance vectors $\boldsymbol{a}_n$ satisfy the following positivity and sum-to-one constraints

$$\sum_{r=1}^{R} a_{r,n} = 1, \quad a_{r,n} > 0, \forall r \in \{1, \ldots, R\}. \qquad (4)$$

To handle these constraints, we propose to reparameterize the abundance vectors belonging to the following set

$$\mathcal{S} = \left\{ \boldsymbol{a} = [a_1, \ldots, a_R]^T \,\middle|\, a_r > 0, \sum_{r=1}^{R} a_r = 1 \right\} \qquad (5)$$

using the following transformation

$$a_{r,n} = \left(\prod_{k=1}^{r-1} z_{k,n}\right) \times \begin{cases} 1 - z_{r,n} & \text{if } r < R \\ 1 & \text{if } r = R. \end{cases} \qquad (6)$$

This transformation has been recently suggested in [39]. One motivation for using the latent variables $z_{r,n}$ instead of $a_{r,n}$ is

the fact that the constraints (4) for the nth abundance vector $\boldsymbol{a}_n$ express as

$$0 < z_{r,n} < 1, \quad \forall r \in \{1, \ldots, R-1\} \tag{7}$$

for the nth coefficient vector $\boldsymbol{z}_n = [z_{1,n}, \ldots, z_{R-1,n}]^T$. As a consequence, the constraints (7) are much easier to handle for the sampling procedure than (4) (as will be shown in Sections IV and V). It is interesting to note that the abundance reparametrization considered in this paper depends on the endmember order. This point will be discussed in Section V-A. The next section presents the Bayesian model associated with the PPNMM (1) for SU.

III. Bayesian Model

This section generalizes the hierarchical Bayesian model introduced in [27] in order to jointly estimate the abundances and endmembers, leading to a fully unsupervised hyperspectral unmixing algorithm. The unknown parameter vector associated with the PPNMM contains the reparameterized abundances $\mathbf{Z} = [\boldsymbol{z}_1, \ldots, \boldsymbol{z}_N]$ (satisfying the constraints (7)), the endmember matrix $\mathbf{M}$, the nonlinearity parameter vector $\boldsymbol{b}$ and the additive noise variance $\boldsymbol{\sigma}^2$. This section summarizes the likelihood and the parameter priors (associated with the proposed hierarchical Bayesian PPNMM) introduced to perform nonlinear unsupervised hyperspectral unmixing.

A. Likelihood

Equation (3) shows that $\mathbf{y}_n|\mathbf{M}, \boldsymbol{z}_n, b_n, \boldsymbol{\sigma}^2$ is distributed according to a Gaussian distribution with mean $\boldsymbol{g}_n(\mathbf{M}\boldsymbol{a}_n)$ and covariance matrix $\boldsymbol{\Sigma}$, denoted as $\mathbf{y}_n|\mathbf{M}, \boldsymbol{z}_n, b_n, \boldsymbol{\sigma}^2 \sim \mathcal{N}\left(\boldsymbol{g}_n(\mathbf{M}\boldsymbol{a}_n), \boldsymbol{\Sigma}\right)$. Note that the abundance vector $\boldsymbol{a}_n$ should be denoted as $\boldsymbol{a}_n(\boldsymbol{z}_n)$. However, the argument $\boldsymbol{z}_n$ has been omitted for brevity. Assuming independence between the observed pixels, the joint likelihood of the observation matrix $\mathbf{Y}$ can be expressed as

$$f(\mathbf{Y}|\mathbf{M}, \mathbf{Z}, \boldsymbol{b}, \boldsymbol{\sigma}^2) \propto |\boldsymbol{\Sigma}|^{-N/2} \text{etr}\left[-\frac{(\mathbf{Y}-\boldsymbol{X})^T\boldsymbol{\Sigma}^{-1}(\mathbf{Y}-\boldsymbol{X})}{2}\right] \tag{8}$$

where $\propto$ means "proportional to", $\text{etr}(\cdot)$ denotes the exponential trace and $\boldsymbol{X} = \mathbf{M}\mathbf{A} + [(\mathbf{M}\mathbf{A}) \odot (\mathbf{M}\mathbf{A})]\,\text{diag}(\boldsymbol{b})$ is an $L \times N$ matrix.

B. Parameter Priors

1) Coefficient Matrix $\mathbf{Z}$*:* To reflect the lack of prior knowledge about the abundances, we propose to assign prior distributions for the coefficient vector $\boldsymbol{z}_n$ that correspond to noninformative prior distributions for $\boldsymbol{a}_n$. More precisely, assigning the following beta priors

$$z_{n,r} \sim \mathcal{B}e(R-r, 1) \quad r \in \{1, \ldots, R-1\} \tag{9}$$

and assuming prior independence between the elements of $\boldsymbol{z}_n$ yield an abundance vector $\boldsymbol{a}_n$ uniformly distributed in the set defined in (5) (see [39] for details). Assuming prior independence between the coefficient vectors $\{\boldsymbol{z}_n\}_{n=1,\ldots,N}$ leads to

$$f(\mathbf{Z}) = \prod_{r=1}^{R-1}\left\{\frac{1}{B(R-r,1)^N}\prod_{n=1}^{N} z_{n,r}^{R-r-1}\right\} \tag{10}$$

where $B(\cdot,\cdot)$ is the Beta function.

2) Endmembers: Each endmember $\mathbf{m}_r = [m_{r,1}, \ldots, m_{r,L}]^T$ is a reflectance vector satisfying the following constraints

$$0 \leq m_{r,\ell} \leq 1, \forall r \in \{1, \ldots, R\}, \forall \ell \in \{1, \ldots, L\}. \tag{11}$$

For each endmember $\mathbf{m}_r$, we propose to use a Gaussian prior

$$\mathbf{m}_r \sim \mathcal{N}_{[0,1]^L}(\bar{\mathbf{m}}_r, s^2\mathbf{I}_L), \tag{12}$$

truncated on $[0,1]^L$ to satisfy the constraints (11). In this paper, we propose to select the mean vectors $\bar{\mathbf{m}}_r$ as the pure components previously identified by the nonlinear EEA studied in [31] and referred to as "Heylen". The variance s^2 reflects the degree of confidence given to this prior information. When no additional knowledge is available, this variance is fixed to a large value ($s^2 = 0.5$ in our simulations). Note that any EEA could be used to define the $L \times R$ matrix $\overline{\mathbf{M}} = [\bar{\mathbf{m}}_1, \ldots, \bar{\mathbf{m}}_R]$.

3) Nonlinearity Parameters: The PPNMM reduces to the LMM for $b_n = 0$. Since the LMM is relevant for most observed pixels, it makes sense to assign prior distributions to the nonlinearity parameters that enforce sparsity for the vector $\boldsymbol{b}$. To detect linear and nonlinear mixtures of the pure spectral signatures in the image, the following conjugate Bernoulli-Gaussian prior is assigned to b_n

$$f(b_n|w, \sigma_b^2) = (1-w)\delta(b_n) + w\mathcal{N}\left(0, \sigma_b^2\right) \tag{13}$$

where $\delta(\cdot)$ denotes the Dirac delta function. Note that the prior distributions for the nonlinearity parameters $\{b_n\}_{n=1,\ldots,N}$ share the same hyperparameters $w \in [0,1]$ and $\sigma_b^2 \in \mathbb{R}^+$. More precisely, the weight w is the prior probability of having a nonlinearly mixed pixel in the image. Assuming prior independence between the nonlinearity parameters $\{b_n\}_{n=1,\ldots,N}$, the joint prior distribution of the nonlinearity parameter vector $\boldsymbol{b}$ can be expressed as follows

$$f(\boldsymbol{b}|w, \sigma_b^2) = \prod_{n=1}^{N} f(b_n|w, \sigma_b^2) \tag{14}$$

4) Noise Variances: A Jeffreys' prior is chosen for the noise variance of each spectral band σ_ℓ^2

$$f(\sigma_\ell^2) \propto \frac{1}{\sigma_\ell^2}\mathbf{1}_{\mathbb{R}^+}\left(\sigma_\ell^2\right) \tag{15}$$

which reflects the absence of knowledge for this parameter (see [40] for motivations). Assuming prior independence between the noise variances, we obtain

$$f(\boldsymbol{\sigma}^2) = \prod_{\ell=1}^{L} f\left(\sigma_\ell^2\right). \tag{16}$$

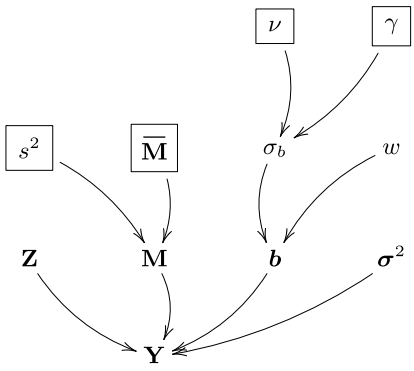


Fig. 1. DAG for the parameter and hyperparameter priors (the fixed parameters appear in boxes).

C. Hyperparameter Priors

The performance of the proposed Bayesian model for spectral unmixing depends on the values of the hyperparameters σ_b^2 and w. When the hyperparameters are difficult to adjust, it is classical to include them in the unknown parameter vector, resulting in a hierarchical Bayesian model [27], [41]. This strategy requires to define prior distributions for the hyperparameters.

A conjugate inverse-Gamma prior is assigned to σ_b^2

$$\sigma_b^2 \sim \mathcal{IG}(\gamma, \nu) \tag{17}$$

where (γ, ν) are real parameters fixed to obtain a flat prior, reflecting the absence of knowledge about the variance σ_b^2 ((γ, ν) will be set to $(10^{-1}, 10^{-1})$ in the simulation section). A uniform prior distribution is assigned to w

$$w \sim \mathcal{U}_{[0,1]}(w) \tag{18}$$

since there is no a priori information regarding the proportions of linearly and nonlinearly mixed pixels in the image. The resulting directed acyclic graph (DAG) associated with the proposed Bayesian model is depicted in Fig. 1.

As mentioned in Section II, the PPNMM is an analytical model which could not systematically provide physically meaningful endmember estimates. Even when assuming the classical LMM, geometric considerations (such as volume constraints) and positivity constraints are used to favor reasonable endmember estimates. The problem becomes more difficult when considering possible nonlinearities. In this paper, range constraints are first considered for the endmembers to ensure that the estimated spectral signatures belong to the reflectance domain $[0, 1]$. Second, the prior distribution assigned to the nonlinearity parameters favors weak/null nonlinearities for pixels that can be accurately modeled by the LMM. Consequently, the estimated endmembers are implicitly enforced to be close to those that would be estimated using LMM-based endmember extraction algorithms. Moreover, similarly to the Bayesian algorithm studied in [41] for linear unsupervised unmixing, the prior distribution assigned to the abundance vectors promotes small volumes for the simplex defined by the endmembers, i.e, promotes endmembers close to the data and thus physically meaningful (as illustrated in Sections VI and VII).

D. Joint Posterior Distribution

The joint posterior distribution of the unknown parameter/hyperparameter vector $\{\boldsymbol{\theta}, \boldsymbol{\Phi}\}$ where $\boldsymbol{\theta} = \left\{\mathbf{Z}, \mathbf{M}, \boldsymbol{b}, \boldsymbol{\sigma}^2\right\}$ and $\boldsymbol{\Phi} = \left\{\sigma_b^2, w\right\}$ can be computed using the following hierarchical structure

$$f(\boldsymbol{\theta}, \boldsymbol{\Phi}|\mathbf{Y}) \propto f(\mathbf{Y}|\boldsymbol{\theta}, \boldsymbol{\Phi})\, f(\boldsymbol{\theta}, \boldsymbol{\Phi}) \tag{19}$$

where $f(\mathbf{Y}|\boldsymbol{\theta})$ has been defined in (8). By assuming *a priori* independence between the parameters $\mathbf{Z}$, $\mathbf{M}$, $\boldsymbol{b}$ and $\boldsymbol{\sigma}^2$ and between the hyperparameters σ_b and w, the joint prior distribution of the unknown parameter vector can be expressed as

$$\begin{aligned} f(\boldsymbol{\theta}, \boldsymbol{\Phi}) &= f(\boldsymbol{\theta}|\boldsymbol{\Phi})\, f(\boldsymbol{\Phi}) \\ &= f(\mathbf{Z})\, f(\mathbf{M})\, f(\boldsymbol{\sigma}^2)\, f(\boldsymbol{b}|\sigma_b^2, w)\, f(\sigma_b^2)\, f(w). \end{aligned} \tag{20}$$

The joint posterior distribution $f(\boldsymbol{\theta}, \boldsymbol{\Phi}|\mathbf{Y})$ can then be computed up to a multiplicative constant after replacing (20) and (8) in (19). Unfortunately, it is difficult to obtain closed form expressions for the standard Bayesian estimators (including the maximum a posteriori (MAP) and the minimum mean square error (MMSE) estimators) associated with (19). In this paper, we propose to use efficient Markov Chain Monte Carlo (MCMC) methods to generate samples asymptotically distributed according to (19). Due to the large number of parameters to be sampled, we use an HMC algorithm which allows the number of sampling steps to be reduced and which improves the mixing properties of the sampler. The generated samples are then used to compute the MMSE estimator of the unknown parameter vector $(\boldsymbol{\theta}, \boldsymbol{\Phi})$. The next section summarizes the basic principles of the HMC methods that will be used to sample asymptotically from (19).

IV. Constrained Hamiltonian Monte Carlo Method

HMCs are powerful methods for sampling from many continuous distributions by introducing fictitious momentum variables. Let $\boldsymbol{q} \in \mathbb{R}^D$ be the parameter of interest and $\pi(\boldsymbol{q})$ its corresponding distribution to be sampled from. From statistical mechanics, the distribution $\pi(\boldsymbol{q})$ can be related to a potential energy function $U(\boldsymbol{q}) = -\log\left[\pi(\boldsymbol{q})\right] + c$ where c is a positive constant such that $\int \exp(-U(\boldsymbol{q}) + c)\, \mathrm{d}\boldsymbol{q} = 1$. The Hamiltonian of $\pi(\boldsymbol{q})$ is a function of the energy $U(\boldsymbol{q})$ and of an additional momentum vector $\boldsymbol{p} \in \mathbb{R}^D$ defined as

$$H(\boldsymbol{q}, \boldsymbol{p}) = U(\boldsymbol{q}) + K(\boldsymbol{p}) \tag{21}$$

where $K(\boldsymbol{p})$ is an arbitrary kinetic energy function. Usually, a quadratic kinetic energy is chosen and we propose to use $K(\boldsymbol{p}) = \boldsymbol{p}^T\boldsymbol{p}/2$ in this paper (for reasons explained later). The Hamiltonian (21) defines the following distribution

$$\begin{aligned} f(\boldsymbol{q}, \boldsymbol{p}) &\propto \exp\left[-H(\boldsymbol{q}, \boldsymbol{p})\right] \\ &\propto \pi(\boldsymbol{q}) \exp\left(-\frac{1}{2}\boldsymbol{p}^T\boldsymbol{p}\right) \end{aligned} \tag{22}$$

for $(\boldsymbol{q}, \boldsymbol{p})$ which shows that $\boldsymbol{q}$ and $\boldsymbol{p}$ are independent and that the marginal distribution of $\boldsymbol{p}$ is a $\mathcal{N}(\mathbf{0}_D, \mathbf{I}_D)$ distribution.

The HMC algorithm allows samples to be asymptotically generated according to (22). The ith HMC iteration starts with an initial pair of vectors $(\boldsymbol{q}^{(i)}, \boldsymbol{p}^{(i)})$ and consists of two steps. The first step resamples the initial momentum $\tilde{\boldsymbol{p}}^{(i)}$ according to the standard multivariate Gaussian distribution. The new notation $\tilde{\boldsymbol{p}}^{(i)}$ is introduced here to highlight the fact that initial momentum used in the ith frepfrog scheme differs from the final momentum of the $(i-1)$th iteration, as shown in Algo. 1. The second step uses Hamiltonian dynamics to propose a candidate $(\boldsymbol{q}^*, \boldsymbol{p}^*)$ which is accepted with the following probability

$$\rho = \min\left\{\exp\left[-H(\boldsymbol{q}^*, \boldsymbol{p}^*) + H(\boldsymbol{q}^{(i)}, \tilde{\boldsymbol{p}}^{(i)})\right], 1\right\}. \quad (23)$$

A. Generation of the Candidate $(\boldsymbol{q}^*, \boldsymbol{p}^*)$

Hamiltonian dynamics are usually simulated by discretization methods such as Euler or leapfrog methods. The classical leapfrog method is a discretization scheme composed of N_{LF} steps with a discretization stepsize ϵ. The nth leapfrog step can be expressed as

$$\boldsymbol{p}^{(i,n\epsilon+\epsilon/2)} = \boldsymbol{p}^{(i,n\epsilon)} - \frac{\epsilon}{2}\frac{\partial U}{\partial \boldsymbol{q}^T}\left(\boldsymbol{q}^{(i,n\epsilon)}\right) \quad (24a)$$

$$\boldsymbol{q}^{(i,(n+1)\epsilon)} = \boldsymbol{q}^{(i,n\epsilon)} + \epsilon\, \boldsymbol{p}^{(i,n\epsilon+\epsilon/2)} \quad (24b)$$

$$\boldsymbol{p}^{(i,(n+1)\epsilon)} = \boldsymbol{p}^{(i,n\epsilon+\epsilon/2)} - \frac{\epsilon}{2}\frac{\partial U}{\partial \boldsymbol{q}^T}\left[\boldsymbol{q}^{(i,(n+1)\epsilon)}\right]. \quad (24c)$$

The leapfrog method starts with $(\boldsymbol{q}^{(i,0)}, \tilde{\boldsymbol{p}}^{(i)}) = (\boldsymbol{q}^{(i)}, \tilde{\boldsymbol{p}}^{(i)})$ and the candidate is set after N_{LF} steps to $(\boldsymbol{q}^*, \boldsymbol{p}^*) = (\boldsymbol{q}^{(i,\epsilon N_{\text{LF}})}, \tilde{\boldsymbol{p}}^{(i,\epsilon N_{\text{LF}})})$.

However, if $\boldsymbol{q}$ is subject to constraints, more sophisticated discretization methods must be used. Assume that the vector of interest $\boldsymbol{q} = [q_1, \ldots, q_D]^T$ satisfies the following constraints

$$q_l < q_d < q_u, \quad d \in \{1, \ldots, D\} \quad (25)$$

where q_l (resp. q_u) is the lower (resp. upper) bound for q_d (such kind of constraints need to be satisfied by the elements of $\mathbf{Z}$ and the endmembers in $\mathbf{M}$). In this paper we propose to use the constrained leapfrog scheme studied in [33, Chap. 5], consisting of N_{LF} steps, with a discretization stepsize ϵ_q. Each CHMC iteration starts in a similar way to the classical leapfrog method, with the sequential sampling of the momentum $\boldsymbol{p}$ (24a) and the vector $\boldsymbol{q}$ (24b). However, if the generated vector $\boldsymbol{q}$ violates the constraints (25), it is modified depending on the violated constraints and the momentum is negated (see [33, Chap. 5] for more details). This step is repeated until each component of the generated $\boldsymbol{q}$ satisfies the contraints. The CHMC ends with the update of the momentum $\boldsymbol{p}$ (24c). One iteration of the resulting constrained HMC algorithm (CHMC) is summarized in Algo. 1. As mentioned above, one might think of using a more sophisticated kinetic energy for $\boldsymbol{p}$ to improve the performance of the HMC algorithm. However, the kinetic energy $K(\boldsymbol{p}) = \boldsymbol{p}^T\boldsymbol{p}/2$ allows the discretization method handling the constraints to be simple and will provide good performance for our application (as will be shown in Section VI). The performance of the HMC mainly relies on the values of the parameters N_{LF} and ϵ_q. Fortunately, the choice of ϵ_q is almost independent of N_{LF} such that these two parameters can be tuned sequentially. The procedures used in this paper to adjust N_{LF} and ϵ_q are detailed in the next paragraphs.

Algorithm 1 Constrained Hamiltonian Monte Carlo Iteration

%Initialization of the ith iteration($n = 0$)
- $\boldsymbol{q}^{(i,0)} = \boldsymbol{q}^{(i)}$ satisfying the constraints (25)
- Sample $\boldsymbol{p}^{(i,0)} = \tilde{\boldsymbol{p}}^{(i)} \sim \mathcal{N}(\mathbf{0}_D, \mathbf{I}_D)$

%Modified leapfrog steps
for $n = 0 : N_{\text{LF}} - 1$ **do**
  %Standard leapfrog steps
  - Compute $\boldsymbol{p}^{(i,n\epsilon+\epsilon/2)} = \boldsymbol{p}^{(i,n\epsilon)} - \frac{\epsilon}{2}\frac{\partial U}{\partial \boldsymbol{q}^T}\left(\boldsymbol{q}^{(i,n\epsilon)}\right)$
  - Compute $\boldsymbol{q}^{(i,(n+1)\epsilon)} = \boldsymbol{q}^{(i,n\epsilon)} + \epsilon\boldsymbol{p}^{(i,n\epsilon+\epsilon/2)}$

  %Steps required to ensure $\boldsymbol{q}^{(i,(n+1)\epsilon)}$ satisfies the constraints (25)
  while $\boldsymbol{q}^{(i,(n+1)\epsilon)}$ does not satisfy (25) **do**
    for $d = 1 : D$ **do**
      if $q_d^{(i,(n+1)\epsilon)} < q_l$ **then**
        Set $q_d^{(i,(n+1)\epsilon)} = 2q_l - q_d^{(i,(n+1)\epsilon)}$
        (replace $q_d^{(i,(n+1)\epsilon)}$ by its symmetric with respect to q_l)
        Set $p_d^{(i,n\epsilon+\epsilon/2)} = -p_d^{(i,n\epsilon+\epsilon/2)}$
      end if
      if $q_d^{(t+\epsilon)} > q_u$ **then**
        Set $q_d^{(i,(n+1)\epsilon)} = 2q_u - q_d^{(i,(n+1)\epsilon)}$
        (replace $q_d^{(i,(n+1)\epsilon)}$ by its symmetric with respect to q_u)
        Set $p_d^{(i,n\epsilon+\epsilon/2)} = -p_d^{(i,n\epsilon+\epsilon/2)}$
      end if
    end for
  end while
  %Standard leapfrog step
  Compute $\boldsymbol{p}^{(i,(n+1)\epsilon)} = \boldsymbol{p}^{(i,n\epsilon+\epsilon/2)} - \frac{\epsilon}{2}\frac{\partial U}{\partial \boldsymbol{q}^T}\left[\boldsymbol{q}^{(i,(n+1)\epsilon)}\right]$
end for
%Accept-reject procedure
Set $\boldsymbol{p}^* = \boldsymbol{p}^{(i,\epsilon N_{\text{LF}})}$ and $\boldsymbol{q}^* = \boldsymbol{q}^{(i,\epsilon N_{\text{LF}})}$
Compute ρ using (23)
Set $(\boldsymbol{q}^{(i+1)}, \boldsymbol{p}^{(i+1)}) = (\boldsymbol{q}^*, \boldsymbol{p}^*)$ with probability ρ
Else set $(\boldsymbol{q}^{(i+1)}, \boldsymbol{p}^{(i+1)}) = (\boldsymbol{q}^{(i)}, \tilde{\boldsymbol{p}}^{(i)})$.

B. Tuning the Stepsize ϵ_q

The step size ϵ_q is related to the accuracy of the leapfrog method to approximate the Hamiltonian dynamics. When ϵ_q is "small", the approximation of the Hamiltonian dynamic is accurate and the acceptance rate (23) is high. However, the exploration of the distribution support is slow (for a given N_{LF}). In this paper, we propose to tune the stepsize during the burn-in period of the sampler. More precisely, the stepsize is decreased (resp. increased) by 25% if the average acceptance rate over the last 50 iterations is smaller than 0.5 (resp. higher than 0.8). Note that the stepsize update only happens during the burn-in period to ensure the Markov chain is homogeneous after the burn-in period.

C. Tuning the Number of Leapfrog Steps N_{LF}

Assume ϵ_q has been correctly adjusted. Too small values of N_{LF} lead to a slow exploration of the distribution (random

walk behavior) whereas too high values of N_{LF} require high computational time. Similarly to the stepsize ϵ_q, the optimal choice of N_{LF} depends on the distribution to be sampled. The sampling procedure proposed in this paper consists of several HMC updates included in a Gibbs sampler (as will be shown in the next section). The number of leapfrog steps required for each of these CHMC updates has been adjusted by cross-validation. From preliminary runs, we have observed that setting the number of leapfrog steps for each HMC update close to $N_{\mathrm{LF}} = 50$ provides a reasonable tradeoff ensuring a good exploration of the target distribution and a reasonable computational complexity. To avoid possible periodic trajectories, it is recommended to let N_{LF} random [33, Chap. 5]. In this paper, we have assumed that N_{LF} is uniformly drawn in the interval [45], [55] at each iteration of the Gibbs sampler.

Convergence isues associated with HMCs have been discussed in details in [33, Chap. 5]. In particular, it has been shown that a sampling scheme combining HMC updates within a classical Gibbs sampler (as will be used in the next section) converges to the target distribution. Although the performance improvement that can be obtained by replacing random walk procedures by HMC updates can be evaluated in closed form for simple problems, the gain obtained when using HMCs within a Gibbs sampler is more difficult to evaluate. The reader in invited to consult [42] for additional simulations illustrating the convergence of the proposed HMC-based sampler. The next section presents the Gibbs sampler (including CHMC steps) which is proposed to sample according to (19).

V. Gibbs Sampler

The principle of the Gibbs sampler is to sample according to the conditional distributions of the posterior of interest [34, Chap. 10]. Due to the large number of parameters to be estimated, it makes sense to use a block Gibbs sampler to improve the convergence of the sampling procedure. More precisely, we propose to sample sequentially $\mathbf{M}$, $\mathbf{Z}$, $\boldsymbol{b}$, $\boldsymbol{\sigma}^2$, σ_b^2 and w using six moves that are detailed in the next sections.

A. Sampling the Coefficient Matrix $\mathbf{Z}$

Sampling from $f(\mathbf{Z}|\mathbf{Y}, \mathbf{M}, \boldsymbol{b}, \boldsymbol{\sigma}^2, \sigma_b^2, w)$ is difficult due to the complexity of this distribution. In this case, it is classical to use an accept/reject procedure to update the coefficient matrix $\mathbf{Z}$ (leading to a hybrid Metropolis-Within-Gibbs sampler). Since the elements of $\mathbf{Z}$ satisfy the constraints (7), the CHMC studied in Section IV could be used to sample according to the conditional distribution $f(\mathbf{Z}|\mathbf{Y}, \mathbf{M}, \boldsymbol{b}, \boldsymbol{\sigma}^2, \sigma_b, w)$. However, as for Metropolis-Hastings updates, the convergence of HMCs generally slows down when the dimensionality of the vector to be sampled increases. Consequently, sampling an $N(R-1)$-dimensional vector using the proposed CHMC can be inefficient when the number of pixels is very large. However, it can be shown that

$$f(\mathbf{Z}|\mathbf{Y}, \mathbf{M}, \boldsymbol{b}, \boldsymbol{\sigma}^2, \sigma_b, w) = \prod_{n=1}^{N} f(\boldsymbol{z}_n|\mathbf{y}_n, \mathbf{M}, b_n, \boldsymbol{\sigma}^2) \quad (26)$$

i.e., the N coefficients vectors $\{\boldsymbol{z}_n\}_{n=1,\ldots,N}$ are a posteriori independent and can be sampled independently in a parallel manner. Straightforward computations lead to

$$f(\boldsymbol{z}_n|\mathbf{y}_n, \mathbf{M}, b_n, \boldsymbol{\sigma}^2) \propto \exp\left(-\frac{(\mathbf{y}_n - \boldsymbol{x}_n)^T \boldsymbol{\Sigma}^{-1}(\mathbf{y}_n - \boldsymbol{x}_n)}{2}\right) \times \mathbf{1}_{(0,1)^{R-1}}(\boldsymbol{z}_n) \prod_{r}^{R-1} z_{n,r}^{R-r-1} \quad (27)$$

where $\boldsymbol{x}_n = \boldsymbol{g}_n(\mathbf{M}\boldsymbol{a}_n)$, $\mathbf{1}_{(0,1)^{R-1}}(\cdot)$ denotes the indicator function over $(0,1)^{R-1}$. The distribution (27) is related to the following potential energy

$$U(\boldsymbol{z}_n) = \frac{(\mathbf{y}_n - \boldsymbol{x}_n)^T \boldsymbol{\Sigma}^{-1}(\mathbf{y}_n - \boldsymbol{x}_n)}{2} - \sum_{r=1}^{R-1} \log\left(z_{n,r}^{R-r-1}\right) \quad (28)$$

where we note that $f(\boldsymbol{z}_n|\mathbf{y}_n, \mathbf{M}, b_n, \boldsymbol{\sigma}^2) \propto \exp[-U(\boldsymbol{z}_n)]$. N momentum vectors associated with a canonical kinetic energy are introduced. The CHMC of Section IV is then applied independently to the N vectors $\boldsymbol{z}_n$ whose dimension $(R-1)$ is relatively small. The partial derivatives of the potential function (28) required in Algo. 1 are derived in the Appendix. As mentioned in Section II-B, the latent variables depend on the endmember order. However, since the proposed CHMC uses (27) to build an appropriate proposal distribution, the impact of the initial endmember permutation on the generated samples is not significant (see [42] for additional simulations obtained with different permutations).

B. Sampling the Endmember Matrix $\mathbf{M}$

From (19) and (20), it can be seen that

$$f(\mathbf{M}|\mathbf{Y}, \mathbf{Z}, \boldsymbol{b}, \boldsymbol{\sigma}^2, s^2, \overline{\mathbf{M}}) = \prod_{\ell=1}^{L} f(\mathbf{m}_{\ell,:}|\mathbf{y}_{\ell,:}, \mathbf{Z}, \boldsymbol{b}, \sigma_\ell^2, s^2, \bar{\mathbf{m}}_{\ell,:})$$

where $\mathbf{m}_{\ell,:}$ (resp. $\bar{\mathbf{m}}_{\ell,:}$ and $\mathbf{y}_{\ell,:}$) is the ℓth row of $\mathbf{M}$ (resp. of $\overline{\mathbf{M}}$ and $\mathbf{Y}$) and

$$f(\mathbf{m}_{\ell,:}|\mathbf{y}_{\ell,:}, \mathbf{Z}, \boldsymbol{b}, \sigma_\ell^2, s^2, \bar{\mathbf{m}}_{\ell,:}) \propto \exp\left(-\frac{\|\mathbf{y}_{\ell,:} - \boldsymbol{t}_\ell\|^2}{2\sigma_\ell^2}\right) \times \exp\left(-\frac{\|\mathbf{m}_{\ell,:} - \bar{\mathbf{m}}_{\ell,:}\|^2}{2s^2}\right) \mathbf{1}_{(0,1)^R}(\mathbf{m}_{\ell,:}) \quad (29)$$

with $\boldsymbol{t}_\ell = \mathbf{A}^T\mathbf{m}_{\ell,:} + \mathrm{diag}(\boldsymbol{b})\left[(\mathbf{A}^T\mathbf{m}_{\ell,:}) \odot (\mathbf{A}^T\mathbf{m}_{\ell,:})\right]$. Consequently, the rows of the endmember matrix $\mathbf{M}$ can be sampled independently similarly to the procedure described in the previous section (to sample $\mathbf{Z}$). More precisely, we introduce a potential energy $V(\mathbf{m}_{\ell,:})$ associated with $\mathbf{m}_{\ell,:}$ defined by

$$V(\mathbf{m}_{\ell,:}) = \frac{\|\mathbf{y}_{\ell,:} - \boldsymbol{t}_\ell\|^2}{2\sigma_\ell^2} + \frac{\|\mathbf{m}_{\ell,:} - \bar{\mathbf{m}}_{\ell,:}\|^2}{2s^2} \quad (30)$$

and a momentum vector associated with a canonical kinetic energy. The partial derivatives of the potential function (30) required in Algo. 1 are derived in the Appendix.

C. Sampling the Nonlinearity Parameter Vector $\boldsymbol{b}$

Using (19) and (20), it can be easily shown that the conditional distribution of $b_n|\mathbf{y}_n, \mathbf{M}\boldsymbol{z}_n, \boldsymbol{\sigma}^2, w, \sigma_b^2$ is the following Bernoulli-Gaussian distribution

$$b_n|\mathbf{y}_n, \mathbf{M}, \boldsymbol{z}_n, \boldsymbol{\sigma}^2, w, \sigma_b^2 \sim (1-w_n^*)\delta(b_n)+w_n^*\mathcal{N}\left(\mu_n, s_n^2\right) \tag{31}$$

where

$$\mu_n = \frac{\sigma_b^2\,(\mathbf{y}_n - \mathbf{M}\boldsymbol{a}_n)^T\,\boldsymbol{\Sigma}^{-1}\boldsymbol{h}_n}{\sigma_b^2\boldsymbol{h}_n^T\boldsymbol{\Sigma}^{-1}\boldsymbol{h}_n + 1}, \quad s_n^2 = \frac{\sigma_b^2}{\sigma_b^2\boldsymbol{h}_n^T\boldsymbol{\Sigma}^{-1}\boldsymbol{h}_n + 1}$$

and $\boldsymbol{h}_n = (\mathbf{M}\boldsymbol{a}_n) \odot (\mathbf{M}\boldsymbol{a}_n)$. Moreover,

$$w_n^* = \frac{w}{\beta_n + w(1-\beta_n)}$$
$$\beta_n = \frac{\sigma_b}{s_n}\exp\left(-\frac{\mu_n^2}{2s_n^2}\right). \tag{32}$$

For each b_n, the conditional distribution (31) does not depend on $\{b_k\}_{k\neq n}$. Consequently, the nonlinearity parameters $\{b_n\}_{n=1,\ldots,N}$ can be sampled independently.

D. Sampling the Noise Variance Vector $\boldsymbol{\sigma}^2$

Using (19), it can be shown that

$$f(\boldsymbol{\sigma}^2|\mathbf{Y}, \mathbf{M}, \mathbf{Z}, \boldsymbol{b}) = \prod_{\ell=1}^{L} f(\sigma_\ell^2|\mathbf{y}_{\ell,:}, \mathbf{m}_{:,\ell}, \mathbf{Z}, \boldsymbol{b}) \tag{33}$$

and that $\sigma_\ell^2|\mathbf{y}_{\ell,:}, \mathbf{m}_{:,\ell}, \mathbf{Z}, \boldsymbol{b}$ is distributed according to the following inverse-gamma distribution

$$\sigma_\ell^2|\mathbf{y}_{\ell,:}, \mathbf{m}_{:,\ell}, \mathbf{Z}, \boldsymbol{b} \sim \mathcal{IG}\left(\frac{N}{2}, \frac{(\mathbf{y}_{\ell,:} - \boldsymbol{x}_{\ell,:})^T(\mathbf{y}_{\ell,:} - \boldsymbol{x}_{\ell,:})}{2}\right) \tag{34}$$

where $\boldsymbol{X} = [\boldsymbol{x}_{1,:}, \ldots, \boldsymbol{x}_{L,:}]^T$. Thus the noise variances can be sampled easily and independently.

E. Sampling the Hyperparameters σ_b^2 and w

Looking carefully at the posterior distribution (19), it can be seen that $\sigma_b^2|\boldsymbol{b}, \gamma, \nu$ is distributed according to the following inverse-gamma distribution

$$\sigma_b^2|\boldsymbol{b}, \gamma, \nu \sim \mathcal{IG}\left(\frac{n_1}{2} + \gamma, \sum_{n\in I_1}\frac{b_n^2}{2} + \nu\right) \tag{35}$$

with $I_1 = \{n|b_n \neq 0\}$, $n_0 = \|\boldsymbol{b}\|_0$ (where $\|\cdot\|_0$ is the ℓ_0 norm, i.e., the number of elements of $\boldsymbol{b}$ that are different from zero) and $n_1 = N - n_0$, from which it is easy to sample. Similarly

$$w|\boldsymbol{b} \sim \mathcal{B}e(n_1 + 1, n_0 + 1). \tag{36}$$

Finally, the Gibbs sampler (including HMC procedures) used to sample according to the posterior (19) consists of the six steps summarized in Algo. 2. The small number of sampling steps is due to the high parallelization properties of the proposed sampling procedure, i.e., the generation of the N coefficient vectors $\{\boldsymbol{z}_n\}_{n=1,\ldots,N}$, the N nonlinearity parameters $\{b_n\}_{n=1,\ldots,N}$ and the L reflectance vectors $\left\{\mathbf{m}_{\ell,:}\right\}_{\ell=1,\ldots,L}$. After generating N_{MC} samples using the procedures detailed above,

Algorithm 2 Gibbs Sampler

1: Initialization $t = 0$
   - $\mathbf{Z}^{(0)}, \mathbf{M}^{(0)}, \boldsymbol{b}^{(0)}, \boldsymbol{\sigma}^{2(0)}, w^{(0)}, \sigma_b^{2(0)}$.
2: Iterations
3: **for** $t = 1 : N_{\text{MC}}$ **do**
4: Parameter update
5: Sample $\mathbf{Z}^{(t)}$ from the pdfs (27) using a CHMC procedure.
6: Sample $\mathbf{M}^{(t)}$ from the pdfs (29) using a CHMC procedure.
7: Sample $\boldsymbol{b}^{(t)}$ from the pdfs (31).
8: Sample $\boldsymbol{\sigma}^{2(t)}$ from the pdfs (34).
9: Hyperparameter update
10: Sample $\sigma_b^{2(t)}$ from the pdf (35).
11: Sample $w^{(t)}$ from the pdf (36).
12: **end for**

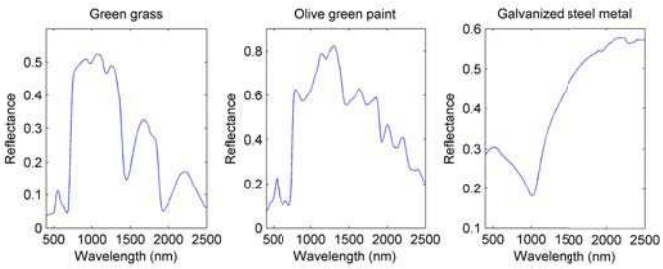


Fig. 2. The $R = 3$ endmembers extracted from the ENVI software [43] used to generated the synthetic images I_1 to I_3.

the MMSE estimator of the unknown parameters can be approximated by computing the empirical averages of these samples, after an appropriate burn-in period. In the simulations conducted in this paper, the number of iterations has been fixed to $N_{\text{MC}} = 15000$ including $N_{\text{bi}} = 14000$ burn-in iterations. The next section studies the performance of the proposed algorithm for synthetic hyperspectral images.

VI. Simulations on Synthetic Data

A. Simulation Scenario

The performance of the proposed unsupervised nonlinear SU algorithm is first evaluated by unmixing 3 synthetic images of size 50×50 pixels. The $R = 3$ endmembers contained in these images (and depicted in Fig. 2) have been extracted from the spectral libraries provided with the ENVI software [43] (i.e., green grass, olive green paint and galvanized steel metal). They consist of $L = 207$ different spectral bands ranging from 400nm to 2500nm with a spectral resolution of 4nm from 400nm to 800nm and of 6nm between 800nm and 2500nm. The main motivation for using these signatures is that these materials have been considered in previous papers [14], [21], [27], [41], allowing better comparisons. The first synthetic image I_1 has been generated using the standard linear mixing model (LMM). A second image I_2 has been generated according to the PPNMM and a third image I_3 has been generated according to the generalized bilinear mixing model (GBM) presented in [14]. Note that the PPNMM does not generalize the GBM but can be used to approximate it (as will be shown in this section). For each image, the abundance vectors a_n have been generated according to a

uniform distribution in the admissible set defined by

$$\mathcal{S}_{\mathrm{t}} = \left\{ \boldsymbol{a} \,\middle|\, 0 < a_r < 0.9, \sum_{r=1}^{R} a_r = 1 \right\}. \tag{37}$$

Note that the conditions $a_r < 0.9$ ensure that there is no pure pixel in the images. All images have been corrupted by an additive independent and identically distributed (i.i.d) Gaussian noise of variance $\sigma^2 = 10^{-4}$, corresponding to an average signal-to-noise ratio SNR $\simeq$ 31dB for the three images. The noise is assumed to be i.i.d. to fairly compare unmixing performance with SU algorithms assuming i.i.d. Gaussian noise. The nonlinearity coefficients are uniformly drawn in the set $[0, 1]$ for the GBM. The parameters b_n have been generated uniformly in the set $[-0.3, 0.3]$ for the PPNMM.

B. Comparison With Other SU Procedures

Different estimation procedures have been considered for the three mixing models. More precisely,

- Two unmixing algorithms have been considered for the LMM. The first strategy extracts the endmembers from the whole image using the N-FINDR algorithm [28] and estimates the abundances using the FCLS algorithm [2] (it is referred to as "SLMM" for supervised LMM). The second strategy is a Bayesian algorithm which jointly estimates the endmembers and the abundance matrix [41] (it is referred to as "ULMM" for unsupervised LMM).
- Two approaches have also been considered for the PPNMM. The first strategy uses the nonlinear EEA studied in [31] and the gradient-based approach based on the PPNMM studied in [27] for estimating the abundances and the nonlinearity parameter. This strategy is referred to as "SPPNMM" (supervised PPNMM). The second strategy is the proposed unmixing procedure referred to as "UPPNMM" (unsupervised PPNMM).
- The unmixing strategy used for the GBM is the nonlinear EEA studied in [31] and the gradient-based algorithm presented in [44] for abundance estimation.

The quality of the unmixing procedures can be measured by comparing the estimated and actual abundance vector using the root normalized mean square error (RNMSE) defined by

$$\mathrm{RNMSE} = \sqrt{\frac{1}{NR} \sum_{n=1}^{N} \left\| \hat{\boldsymbol{a}}_n - \boldsymbol{a}_n \right\|^2} \tag{38}$$

where $\boldsymbol{a}_n$ and $\hat{\boldsymbol{a}}_n$ are the actual and estimated abundance vectors for the nth pixel of the image and N is the number of image pixels. Table I shows the RNMSEs associated with the images $I_1, \ldots, I_3$ for the different estimation procedures. These results show that the proposed UPPNMM performs better (in terms of RNMSE) than the other considered unmixing methods for the three images. Moreover, the proposed method provides similar results when compared with the ULMM for the linearly mixed image I_1.

Fig. 3 compares the endmember simplexes estimated by Heylen's method [31] (black) (used to build the endmember prior) and by the proposed method (red) to the actual endmembers (green stars). For visualization, the observed pixels and the actual and estimated endmembers have been projected onto the three first axes provided by the principal component analysis. These figures show that the proposed unmixing procedure provides accurate estimated endmembers for the three images I_1 to I_3. Due to the absence of pure pixels in the image, the manifold generated by the observed pixels $\mathbf{Y}$ is difficult to estimate. This explains the limited performance obtained with Heylen's method. Conversely, the use of the prior (12) allows the endmembers $\mathbf{m}_r$ to depart from the prior estimations $\bar{\mathbf{m}}_r$ leading to improved performance.

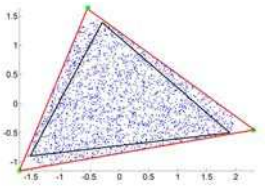
The quality of endmember estimation is also evaluated by the spectral angle mapper (SAM) defined as

$$\mathrm{SAM} = \arccos\left(\frac{\langle \hat{\mathbf{m}}_r, \mathbf{m}_r \rangle}{\|\hat{\mathbf{m}}_r\| \, \|\mathbf{m}_r\|} \right) \tag{39}$$

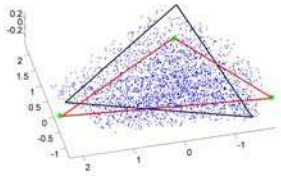
where $\mathbf{m}_r$ is the rth actual endmember and $\hat{\mathbf{m}}_r$ its estimate. The smaller |SAM|, the closer the estimated endmembers to their actual values. Table II compares the performance of the different endmember estimation algorithms. This table shows that the proposed UPPNMM generally provides more accurate endmember estimates than the others methods. Moreover, these results illustrate the robustness of the PPNMM regarding model mis-specification. Note that the ULMM and

TABLE I
ABUNDANCE RNMSEs ($\times 10^{-2}$): SYNTHETIC IMAGES

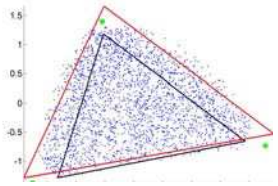
		I_1 (LMM)	I_2 (PPNMM)	I_3 (GBM)
LMM	SLMM	3.78	13.21	6.83
	ULMM	0.66	10.87	4.21
PPNMM	SPPNMM	4.18	6.04	4.13
	UPPNMM	**0.37**	**0.81**	**1.38**
GBM		4.18	11.15	5.02



(a)



(b)



(c)

Fig. 3. Visualization of the $N = 2500$ pixels (blue dots) of (a) I_1, (b) I_2 and (c) I_3 using the first principal components provided by the standard PCA. The green stars correspond to the actual endmembers and the triangles are the simplexes defined by the endmembers estimated by the Heylen's method (black) and the proposed method (red).

TABLE II
SAMs ($\times 10^{-2}$): SYNTHETIC IMAGES

		N-Findr	ULMM	Heylen	UPPNMM
I_1	$\mathbf{m}_1$	5.68	0.95	6.42	**0.27**
	$\mathbf{m}_2$	5.85	**0.32**	7.46	0.36
	$\mathbf{m}_3$	3.31	0.30	5.26	**0.27**
I_2	$\mathbf{m}_1$	9.27	9.68	6.71	**0.59**
	$\mathbf{m}_2$	8.58	8.67	11.80	**0.38**
	$\mathbf{m}_3$	4.47	6.34	4.98	**0.26**
I_3	$\mathbf{m}_1$	7.35	3.42	6.48	**1.50**
	$\mathbf{m}_2$	10.68	**3.13**	11.88	3.22
	$\mathbf{m}_3$	4.34	7.44	3.20	**0.85**

TABLE III
REs ($\times 10^{-2}$): SYNTHETIC IMAGES

		I_1 (LMM)	I_2 (PPNMM)	I_3 (GBM)
LMM	SLMM	1.04	1.74	15.16
	ULMM	**0.99**	1.43	1.07
PPNMM	SPPNMM	1.26	1.27	1.31
	UPPNMM	**0.99**	**0.99**	**0.99**
GBM		1.27	1.64	1.33

the UPPNMM provide similar results (in terms of SAMs) for the image I_1 generated according to the LMM.

Finally, the unmixing quality can be evaluated by the reconstruction error (RE) defined as

$$\text{RE} = \sqrt{\frac{1}{NL}\sum_{n=1}^{N}\left\|\hat{\mathbf{y}}_n - \mathbf{y}_n\right\|^2} \tag{40}$$

where $\mathbf{y}_n$ is the nth observation vector and $\hat{\mathbf{y}}_n$ its estimate. Table III compares the REs obtained for the different synthetic images. These results show that the REs are close for the different unmixing algorithms even if the estimated abundances can vary more significantly (see Table I). Again, the proposed PPNMM seems to be more robust than the other mixing models to deviations from the actual model in terms of RE.

C. Analysis of the Estimated Nonlinearity Parameters

As mentioned above, one of the major properties of the PPNMM is its ability to characterize the linearity/nonlinearity of the underlying mixing model for each pixel of the image via the nonlinearity parameter b_n. Fig. 4 shows the nonlinearity parameter distribution estimated for the three images I_1 to I_3 using the UPPNMM. This figure shows that the UPPNMM clearly identifies the linear mixtures of the image I_1 whereas more nonlinearly mixed pixels can be identified in the images I_2 and I_3. The analysis of Fig. 4 also shows that the nonlinearities contained in the image I_3 (GBM) are generally less significant than the nonlinearities affecting I_2 (PPNMM) for a same signal-to-noise ratio (SNR $\simeq$ 31dB).

D. Performance for Different Numbers of Endmembers

The next set of simulations analyzes the performance of the proposed UPPNMM algorithm for different numbers of

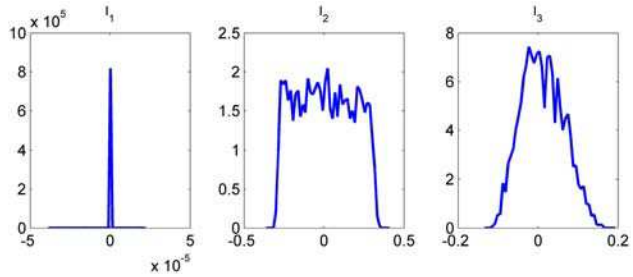


Fig. 4. Distributions of the nonlinearity parameters b_n for the images I_1 (left), I_2 (middle) and I_3 (right).

TABLE IV
UNMIXING PERFORMANCE: SYNTHETIC IMAGES

		$R=4$	$R=5$	$R=6$
Average SAMs ($\times 10^{-2}$)	SPPNMM	7.76	10.78	18.53
	UPPNMM	**0.47**	**0.81**	**1.09**
RNMSEs ($\times 10^{-2}$)	SPPNMM	7.58	10.95	16.52
	UPPNMM	**0.78**	**1.23**	**1.47**
REs ($\times 10^{-2}$)	SPPNMM	1.36	1.46	1.64
	UPPNMM	**0.99**	**0.99**	**0.99**

endmembers ($R \in \{4, 5, 6\}$) by unmixing three synthetic images of $N = 2500$ pixels distributed according to the PPNMM. The endmembers contained in these images have been extracted from the spectral libraries provided with the ENVI software [43]. For each image, the abundance vectors $\boldsymbol{a}_n, n = 1, \ldots, N$ have been randomly generated according to a uniform distribution over the admissible set (37). All images have been corrupted by an additive white Gaussian noise corresponding to $\sigma^2 = 10^{-4}$, corresponding to an average signal-to-noise ratio SNR $\simeq$ 31dB for the three images. The nonlinearity coefficients b_n are uniformly drawn in the set $[-0.3, 0.3]$. Tables IV compares the performance of the proposed method in terms of endmember estimation (average SAMs of the R endmembers), abundance estimation and reconstruction error. These results show a general degradation of the abundance and endmember estimations when R is increasing (this is intuitive since estimator variances usually increase with the number of parameters to be estimated). However, this degradation is reasonable when compared to Heylen's method. The proposed algorithm still provides accurate estimates, as illustrated in Fig. 5 which compares the actual and estimated endmembers associated with the image containing $R = 6$ endmembers.

VII. SIMULATIONS ON REAL DATA

A. Data Sets

The real image considered in this section was acquired in 2010 by the Hyspex hyperspectral scanner over Villelongue, France (00° 03'W and 42°57'N). $L = 160$ spectral bands ranging from about 408nm to 985nm were recorded, with a spectral resolution of 3.6nm and a spatial resolution of 0.5m. This dataset has already been studied in [21] and [45] and is mainly composed of forested and urban areas. More details about the data acquisition and pre-processing steps are available in [45]. Two sub-images denoted as scene #1 and

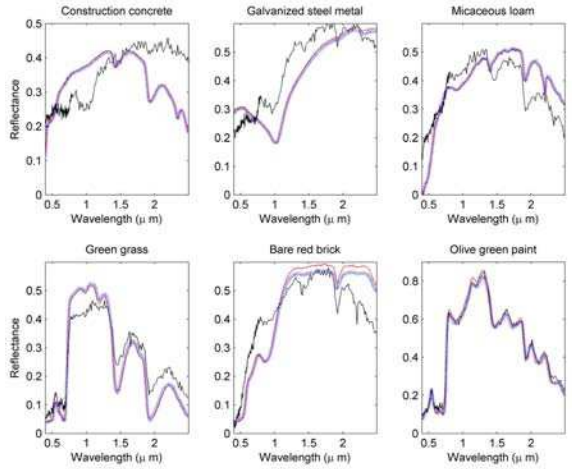


Fig. 5. Actual endmembers (blue dots) and the endmembers estimated by Heylen's method (black lines) and the UPPNMM (red lines) for the synthetic image containing $R = 6$ endmembers.

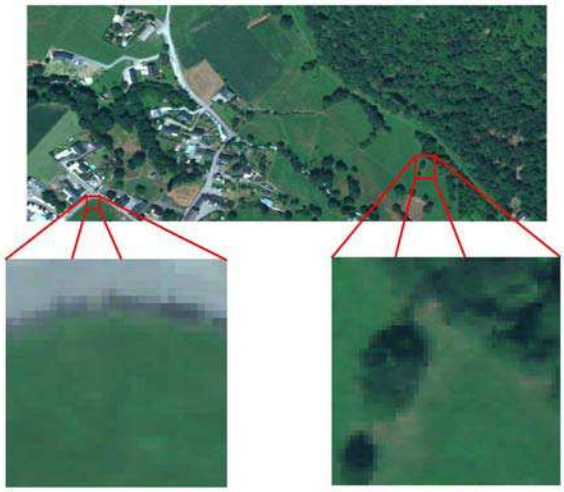


Fig. 6. Top: real hyperspectral Madonna data acquired by the Hyspex hyperspectral scanner over Villelongue, France. Bottom: Scene #1 (left) and Scene #2 (right) shown in true colors.

scene #2 (of size 31×30 and 50×50 pixels) are chosen here to evaluate the proposed unmixing procedure and are depicted in Fig. 6 (bottom images). The scene #1 is mainly composed of road, ditch and grass pixels. The scene #2 is more complex since it includes shadowed pixels. For this image, shadow is considered as an additional endmember, resulting in $R = 4$ endmembers, i.e., tree, grass, soil and shadow.

B. Endmember and Abundance Estimation

The endmembers extracted by N-FINDR, the ULMM algorithm [41] and Heylen's method [31] with $R = 3$ (resp. $R = 4$) for the scene #1 (resp. scene #2) are compared with the endmembers estimated by the UPPNMM in Fig. 7 (resp. Fig. 8). For the scene #1, the four algorithms provide similar endmember estimates whereas the estimated shadow spectra are different for the scene #2. The N-FINDR algorithm and Heylen's method estimate endmembers as the purest pixels of the observed image, which can be problematic when there

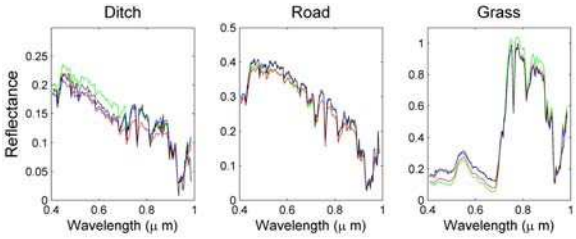


Fig. 7. The $R = 3$ endmembers estimated by N-Findr (blue lines), ULMM (green lines), Heylen's method (black lines) and the UPPNMM (red lines) for the scene #1.

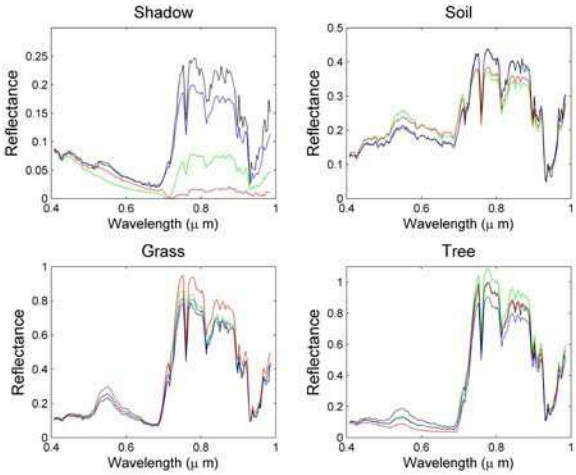


Fig. 8. The $R = 4$ endmembers estimated by N-Findr (blue lines), ULMM (green lines), Heylen's method (black lines) and the UPPNMM (red lines) for the scene #2.

is no pure pixel in the image (as it occurs with shadowed pixels in the scene #2). Conversely, the ULMM and UPPNMM methods, which jointly estimate the endmembers and the abundances seem to provide more relevant shadow spectra (of lower amplitude). Examples of abundance maps for the scene #1 (resp. scene #2), estimated by the ULMM and the UPPNMM algorithms are presented in Fig. 9 (resp. Fig. 10). The abundance maps obtained by the UPPNMM are similar to the abundance maps obtained with ULMM.

C. Analysis of Nonlinearities

Fig. 11 shows the estimated maps of b_n for the two considered images. Different nonlinear regions can be identified in the scene #1, mainly in the grass-planted region (probably due to endmember variability) and near the ditch (presence of relief). For the scene #2, nonlinear effects are mainly detected in shadowed pixels.

D. Estimation of Noise Variances

Fig. 12 compares the noise variance estimated by the UPPNMM for the two real images with the noise variance estimated by the HySime algorithm [46]. The HySime algorithm assumes additive noise and estimates the noise covariance matrix of the image using multiple regression. Fig. 12 shows that the two algorithms provide similar noise variance estimates. Moreover, these results motivate the consideration

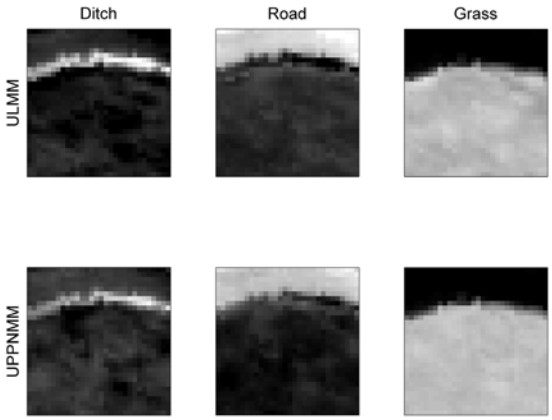


Fig. 9. Abundance maps estimated by the SLMM, the GBM and the UPPNMM algorithms for the scene #1.

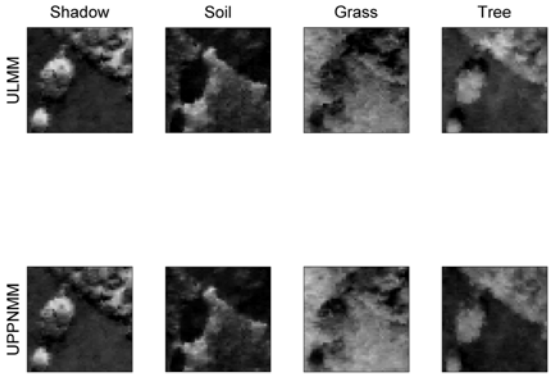


Fig. 10. Abundance maps estimated by the SLMM, the GBM and the UPPNMM algorithms for the scene #2.

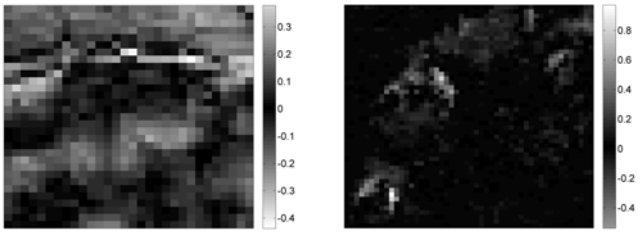


Fig. 11. Maps of the nonlinearity parameter b_n estimated by the UPPNMM for the real images. (a) Scene #1. (b) Scene #2.

of non i.i.d. noise for hyperspectral images since the noise variances increase for the higher wavelengths for the two images.

E. Image Reconstruction

The proposed algorithm is finally evaluated from the REs associated with the two real images. These REs are compared in Table V with those obtained by assuming other mixing models. The two unsupervised algorithms (ULMM and UPPNMM) provide smaller REs than the SU procedures decomposed into two steps. This observation motivates the use of joint abundance and endmember estimation algorithms.

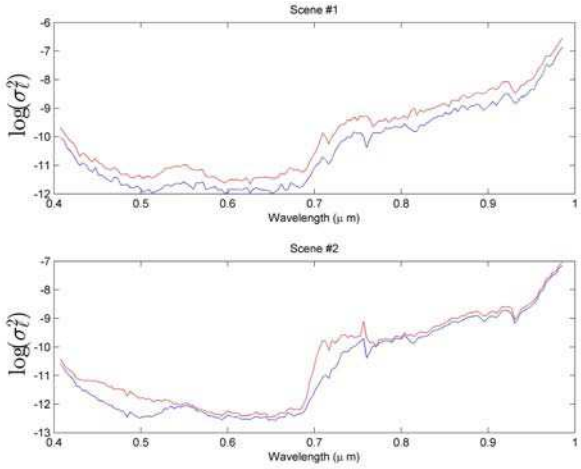


Fig. 12. Noise variances estimated by the UPPNMM (red) and the Hysime algorithm (blue) for the scene #1 (top) and the scene #2 (bottom).

TABLE V
REs ($\times 10^{-2}$): REAL IMAGE

		Scene #1	Scene #2
LMM	SLMM	1.53	1.04
	ULMM	1.11	**0.88**
PPNMM	SPPNMM	1.50	1.17
	UPPNMM	**1.08**	0.89
GBM		1.72	1.25

VIII. CONCLUSION AND FUTURE WORK

We proposed a new hierarchical Bayesian algorithm for unsupervised nonlinear spectral unmixing of hyperspectral images. This algorithm assumed that each pixel of the image is a post-nonlinear mixture of the endmembers contaminated by additive Gaussian noise. The physical constraints for the abundances and endmembers were included in the Bayesian framework through appropriate prior distributions. Due to the complexity of the resulting joint posterior distribution, a Markov chain Monte Carlo method was used to approximate the MMSE estimator of the unknown model parameters. Because of the large number of parameters to be estimated, Hamiltonian Monte Carlo methods were used to reduce the sampling procedure complexity and to improve the mixing properties of the proposed sampler. Simulations conducted on synthetic data illustrated the performance of the proposed algorithm for linear and nonlinear spectral unmixing. An important advantage of the proposed algorithm is its flexibility regarding the absence of pure pixels in the image. Another interesting property resulting from the post-nonlinear mixing model is the possibility of detecting nonlinearly from linearly mixed pixels. This detection can identify the image regions affected by nonlinearities in order to characterize the nonlinear effects more deeply. The number of endmembers contained in the hyperspectral image was assumed to be known in this work. Even if LMM-based methods could be used to estimate the number of components in a scene [46], [47], estimating the number of components present in image containing nonlinear mixtures is an important issue that should be considered in future work. A full Bayesian approach

was used in this paper. However, it could be interesting to consider other strategies (e.g., nonlinear optimization methods) for nonlinear unmixing with reduced computational complexity. Finally, considering endmember variability in linear mixtures has received much attention in the literature [3], and [48]–[50]. Extending these results to nonlinear mixtures is clearly an interesting prospect.

APPENDIX
DERIVATION OF THE POTENTIAL FUNCTIONS

The potential energy (28) can be rewritten

$$U(\boldsymbol{z}_n) = U_1(\boldsymbol{a}_n) + U_2(\boldsymbol{z}_n) \tag{41}$$

where

$$U_1(\boldsymbol{a}_n) = \frac{1}{2}\left[\mathbf{y}_n - \boldsymbol{g}_n\left(\mathbf{M}\boldsymbol{a}_n\right)\right]^T \boldsymbol{\Sigma}^{-1}\left[\mathbf{y}_n - \boldsymbol{g}_n\left(\mathbf{M}\boldsymbol{a}_n\right)\right],$$
$$U_2(\boldsymbol{z}_n) = -\sum_{r=1}^{R-1} \log\left(z_{r,n}^{R-r-1}\right).$$

Partial derivatives of $U(\boldsymbol{z}_n)$ with respect to $\boldsymbol{z}_n$ is obtained using the classical chain rule

$$\frac{\partial U(\boldsymbol{z}_n)}{\partial \boldsymbol{z}_n} = \frac{\partial U_1(\boldsymbol{a}_n)}{\partial \boldsymbol{a}_n}\frac{\partial \boldsymbol{a}_n}{\partial \boldsymbol{z}_n} + \frac{\partial U_2(\boldsymbol{z}_n)}{\partial \boldsymbol{z}_n}$$

Straightforward computations lead to

$$\frac{\partial U_1(\boldsymbol{a}_n)}{\partial \boldsymbol{a}_n} = -\left[\mathbf{y}_n - \boldsymbol{g}_n\left(\mathbf{M}\boldsymbol{a}_n\right)\right]^T \boldsymbol{\Sigma}^{-1}\left[\mathbf{M} + 2b_n\left(\mathbf{M}\boldsymbol{a}_n\mathbf{1}_R^T\right)\odot\mathbf{M}\right]$$
$$\frac{\partial a_{r,n}}{\partial z_{i,n}} = \begin{cases} 0 & \text{if} \quad i > r \\ \dfrac{a_{r,n}}{z_{i,n}-1} & \text{if} \quad i = r \\ \dfrac{a_{r,n}}{z_{i,n}} & \text{if} \quad i < r \end{cases}$$
$$\frac{\partial U_2(\boldsymbol{z}_n)}{\partial z_{i,n}} = -\frac{R-i-1}{z_{i,n}}. \tag{42}$$

Similarly, the potential energy (30) can be rewritten

$$V(\mathbf{m}_{\ell,:}) = V_1(\boldsymbol{t}_\ell) + V_2(\boldsymbol{z}_n) \tag{43}$$

with $\boldsymbol{t}_\ell = \mathbf{A}^T\mathbf{m}_{\ell,:} + \text{diag}\left(\boldsymbol{b}\right)\left[\left(\mathbf{A}^T\mathbf{m}_{\ell,:}\right)\odot\left(\mathbf{A}^T\mathbf{m}_{\ell,:}\right)\right]$ and

$$V_1(\boldsymbol{t}_\ell) = \frac{\|\mathbf{y}_{\ell,:} - \boldsymbol{t}_\ell\|^2}{2\sigma_\ell^2}$$
$$V_2(\mathbf{m}_{\ell,:}) = \frac{\|\mathbf{m}_{\ell,:} - \bar{\mathbf{m}}_{\ell,:}\|^2}{2s^2}.$$

The partial derivatives of the potential energy (30) can be obtained using the chain rule

$$\frac{\partial V(\mathbf{m}_{\ell,:})}{\partial \mathbf{m}_{\ell,:}} = \frac{\partial V_1(\boldsymbol{t}_\ell)}{\partial \boldsymbol{t}_\ell}\frac{\partial \boldsymbol{t}_\ell}{\partial \mathbf{m}_{\ell,:}} + \frac{\partial V_2(\mathbf{m}_{\ell,:})}{\partial \mathbf{m}_{\ell,:}}$$

and

$$\frac{\partial V_1(\boldsymbol{t}_\ell)}{\partial \boldsymbol{t}_\ell} = -\frac{(\mathbf{y}_{\ell,:} - \boldsymbol{t}_\ell)^T}{\sigma_\ell^2}$$
$$\frac{\partial \boldsymbol{t}_\ell}{\partial \mathbf{m}_{\ell,:}} = \mathbf{A}^T + 2\text{diag}\left(\boldsymbol{b}\right)\left[\left(\mathbf{A}^T\mathbf{m}_{\ell,:}\mathbf{1}_R^T\right)\odot\mathbf{A}^T\right]$$
$$\frac{\partial V_2(\mathbf{m}_{\ell,:})}{\partial \mathbf{m}_{\ell,:}} = \frac{(\mathbf{m}_{\ell,:} - \bar{\mathbf{m}}_{\ell,:})^T}{s^2}.$$

REFERENCES

[1] M. Craig, "Minimum volume transforms for remotely sensed data," *IEEE Trans. Geosci. Remote Sensing*, vol. 32, no. 3, pp. 542–552, May 1994.

[2] D. C. Heinz and C.-I. Chang, "Fully constrained least-squares linear spectral mixture analysis method for material quantification in hyperspectral imagery," *IEEE Trans. Geosci. Remote Sensing*, vol. 29, no. 3, pp. 529–545, Mar. 2001.

[3] O. Eches, N. Dobigeon, C. Mailhes, and J.-Y. Tourneret, "Bayesian estimation of linear mixtures using the normal compositional model," *IEEE Trans. Image Process.*, vol. 19, no. 6, pp. 1403–1413, Jun. 2010.

[4] L. Miao, H. Qi, and H. Szu, "A maximum entropy approach to unsupervised mixed-pixel decomposition," *IEEE Trans. Image Process.*, vol. 16, no. 4, pp. 1008–1021, Apr. 2007.

[5] Z. Yang, G. Zhou, S. Xie, S. Ding, J.-M. Yang, and J. Zhang, "Blind spectral unmixing based on sparse nonnegative matrix factorization," *IEEE Trans. Image Process.*, vol. 20, no. 4, pp. 1112–1125, Apr. 2011.

[6] N. Keshava and J. F. Mustard, "Spectral unmixing," *IEEE Signal Process. Mag.*, vol. 19, no. 1, pp. 44–57, Jan. 2002.

[7] J. M. Bioucas-Dias *et al.*, "Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches," *IEEE J. Sel. Topics Appl. Earth Observat. Remote Sensing*, vol. 5, no. 2, pp. 354–379, Apr. 2012.

[8] B. W. Hapke, "Bidirectional reflectance spectroscopy. I. Theory," *J. Geophys. Res.*, vol. 86, no. B4, pp. 3039–3054, 1981.

[9] R. Close, P. Gader, J. Wilson, and A. Zare, "Using physics-based macroscopic and microscopic mixture models for hyperspectral pixel unmixing," *Proc. SPIE*, vol. 8390, pp. 83901L-1–83901L-13, May 2012.

[10] A. Banerjee, H. Ren, and J. O. Jensen, "Extended linear hyperspectral mixing models," *Proc. SPIE*, vol. 5268, pp. 288–301, Feb. 2004.

[11] B. Somers *et al.*, "Nonlinear hyperspectral mixture analysis for tree cover estimates in orchards," *Remote Sensing Environ.*, vol. 113, no. 6, pp. 1183–1193, 2009.

[12] J. M. P. Nascimento and J. M. Bioucas-Dias, "Nonlinear mixture model for hyperspectral unmixing," *Proc. SPIE*, vol. 7477, no. 1, p. 74770I, 2009.

[13] W. Fan, B. Hu, J. Miller, and M. Li, "Comparative study between a new nonlinear model and common linear model for analysing laboratory simulated-forest hyperspectral data," *Remote Sensing Environ.*, vol. 30, no. 11, pp. 2951–2962, Jun. 2009.

[14] A. Halimi, Y. Altmann, N. Dobigeon, and J.-Y. Tourneret, "Nonlinear unmixing of hyperspectral images using a generalized bilinear model," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 11, pp. 4153–4162, Nov. 2011.

[15] I. Meganem, P. Deliot, X. Briottet, Y. Deville, and S. Hosseini, "Linear-quadratic mixing model for reflectances in urban environments," *IEEE Trans. Geosci. Remote Sensing*, vol. 52, no. 1, pp. 544–558, Jan. 2014.

[16] N. Yokoya, J. Chanussot, and A. Iwasaki, "Nonlinear unmixing of hyperspectral data using semi-nonnegative matrix factorization," *IEEE Trans. Geosci. Remote Sensing*, vol. 52, no. 2, pp. 1430–1437, Feb. 2014.

[17] Q. Qu, X. Sun, N. Nasrabadi, and T. Tran, "Low rank representation for bilinear abundance estimation problem," in *Proc. IEEE WHISPERS*, Gainesville, FL, USA, Jul. 2013, pp. 1–4.

[18] K. J. Guilfoyle, M. L. Althouse, and C.-I. Chang, "A quantitative and comparative analysis of linear and nonlinear spectral mixture models using radial basis function neural networks," *IEEE Geosci. Remote Sensing Lett.*, vol. 39, no. 8, pp. 2314–2318, Aug. 2001.

[19] Y. Altmann, N. Dobigeon, S. McLaughlin, and J.-Y. Tourneret, "Nonlinear unmixing of hyperspectral images using radial basis functions and orthogonal least squares," in *Proc. IEEE Int. Conf. Geosci. Remote Sensing*, Jul. 2011, pp. 1151–1154.

[20] J. Chen, C. Richard, and P. Honeine, "Nonlinear unmixing of hyperspectral data based on a linear-mixture/nonlinear-fluctuation model," *IEEE Trans. Signal Process.*, vol. 61, no. 2, pp. 480–492, Jan. 2013.

[21] Y. Altmann, N. Dobigeon, S. McLaughlin, and J. Tourneret, "Nonlinear spectral unmixing of hyperspectral images using Gaussian processes," *IEEE Trans. Signal Process.*, vol. 61, no. 10, pp. 2442–2453, May 2013.

[22] R. S. Rand, A. Banerjee, and J. Broadwater, "Automated endmember determination and adaptive spectral mixture analysis using kernel methods," *Proc. SPIE*, vol. 8870, pp. 88700Q-1–88700Q-17, Sep. 2013.

[23] C. Jutten and J. Karhunen, "Advances in nonlinear blind source separation," in *Proc. 4th Int. Symp. Independ. Compon. Anal. Blind Signal Separat.*, Nara, Japan, Apr. 2003, pp. 245–256.

[24] M. Babaie-Zadeh, C. Jutten, and K. Nayebi, "Separating convolutive post non-linear mixtures," in *Proc. 3rd Workshop Independ. Compon. Anal. Signal Separat.*, San Diego, CA, USA, 2001, pp. 138–143.

[25] J. Chen, C. Richard, and P. Honeine, "Estimating abundance fractions of materials in hyperspectral images by fitting a post-nonlinear mixing model," in *Proc. IEEE WHISPERS*, Gainesville, FL, USA, Jun. 2013, pp. 1–4.

[26] N. Dobigeon, J.-Y. Tourneret, C. Richard, J. Bermudez, S. McLaughlin, and A. Hero, "Nonlinear unmixing of hyperspectral images: Models and algorithms," *IEEE Signal Process. Mag.*, vol. 31, no. 1, pp. 82–94, Jan. 2014.

[27] Y. Altmann, A. Halimi, N. Dobigeon, and J. Tourneret, "Supervised nonlinear spectral unmixing using a postnonlinear mixing model for hyperspectral imagery," *IEEE Trans. Image Process.*, vol. 21, no. 6, pp. 3017–3025, Jun. 2012.

[28] M. Winter, "Fast autonomous spectral end-member determination in hyperspectral data," in *Proc. 13th Int. Conf. Appl. Geologic Remote Sensing*, vol. 2. Apr. 1999, pp. 337–344.

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

[30] F. Chaudhry, C.-C. Wu, W. Liu, C.-I. Chang, and A. Plaza, "Pixel purity index-based algorithms for endmember extraction from hyperspectral imagery," in *Recent Advances in Hyperspectral Signal and Image Processing*, C.-I Chang, Ed. Trivandrum, Kerala, India: Res. Signpost, 2006, ch. 2.

[31] R. Heylen, D. Burazerovic, and P. Scheunders, "Non-linear spectral unmixing by geodesic simplex volume maximization," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 3, pp. 534–542, Jun. 2011.

[32] S. Duane, A. D. Kennedy, B. J. Pendleton, and D. Roweth, "Hybrid Monte Carlo," *Phys. Lett. B*, vol. 195, pp. 216–222, Sep. 1987.

[33] S. Brooks, *Handbook of Markov Chain Monte Carlo*. New York, NY, USA: Taylor & Francis, 2011.

[34] C. P. Robert and G. Casella, *Monte Carlo Statistical Methods*, 2nd ed. New York, NY, USA: Springer-Verlag, 2004.

[35] R. M. Neal, *Bayesian Learning for Neural Networks* (Lecture Notes in Statistics), 1st ed. Secaucus, NJ, USA: Springer-Verlag, 1996.

[36] M. N. Schmidt, "Function factorization using warped Gaussian processes," in *Proc. Int. Conf. Mach. Learn.*, 2009, pp. 921–928.

[37] C. Hartmann and C. Schuette, "A constrained hybrid Monte-Carlo algorithm and the problem of calculating the free energy in several variables," *ZAMM - J. Appl. Math. Mech. / Zeitschrift Angew. Math. Mech.*, vol. 85, no. 10, pp. 700–710, 2005.

[38] M. A. Brubaker, M. Salzmann, and R. Urtasun, "A family of MCMC methods on implicitly defined manifolds," *J. Mach. Learn. Res.*, vol. 22, pp. 161–172, 2012.

[39] M. J. Betancourt, "Cruising the simplex: Hamiltonian Monte Carlo and the Dirichlet distribution," Oct. 2010.

[40] J. M. Bernardo and A. F. M. Smith, *Bayesian Theory*. New York, NY, USA: Wiley, 1994.

[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. 2657–2669, Nov. 2009.

[42] Y. Altmann, N. Dobigeon, and J.-Y. Tourneret, "Unsupervised post-nonlinear unmixing of hyperspectral images using a hamiltonian Monte Carlo algorithm," Signal and Communications Group, IRIT/ENSEEIHT, Toulouse, France, Tech. Rep. 1, Sep. 2013.

[43] *ENVI User's guide Version 4.0*, Research Systems Inc., Boulder, CO, USA, Sep. 2003.

[44] A. Halimi, Y. Altmann, N. Dobigeon, and J.-Y. Tourneret, "Unmixing hyperspectral images using a generalized bilinear model," in *Proc. IEEE Int. Conf. Geosci. Remote Sensing*, Jul. 2011, pp. 1886–1889.

[45] D. Sheeren, M. Fauvel, S. Ladet, A. Jacquin, G. Bertoni, and A. Gibon, "Mapping ash tree colonization in an agricultural mountain landscape: Investigating the potential of hyperspectral imagery," in *Proc. IEEE Int. Conf. Geosci. Remote Sensing*, Jul. 2011, pp. 3672–3675.

[46] J. M. Bioucas-Dias and J. M. P. Nascimento, "Hyperspectral subspace identification," *IEEE Trans. Geosci. Remote Sensing*, vol. 46, no. 8, pp. 2435–2445, Aug. 2008.

[47] A. Ambikapathi, T.-H. Chan, C.-Y. Chi, and K. Keizer, "Hyperspectral data geometry-based estimation of number of endmembers using p-norm-based pure pixel identification algorithm," *IEEE Trans. Geosci. Remote Sensing*, vol. 51, no. 5, pp. 2753–2769, May 2013.

[48] B. Somers, G. P. Asner, L. Tits, and P. Coppin, "Endmember variability in spectral mixture analysis: A review," *Remote Sensing Environ.*, vol. 115, no. 7, pp. 1603–1616, 2011.

[49] A. Zare, P. Gader, T. Allgire, D. Drashnikov, and R. Close, "Bootstrapping for piece-wise convex endmember distribution detection," in *Proc. IEEE GRSS WHISPERS*, May 2012, pp. 1–4.

[50] A. Zare and K. C. Ho, "Endmember variability in hyperspectral analysis," *IEEE Signal Process. Mag.*, vol. 31, no. 1, pp. 95–104, Jan. 2014.

Yoann Altmann (S'10–M'14) was born in Toulouse, France, in 1987. He received the Engineering degree in electrical engineering from Ecole Nationale Supérieure d'Electronique, d'Electrotechnique, d'Informatique, d'Hydraulique et des Télécommunications, Toulouse, and the M.Sc. degree in signal processing from the National Polytechnic Institute of Toulouse (INP Toulouse), Toulouse, both in 2010, and the Ph.D. degree from INP Toulouse in 2013. Since 2014, he has been with the Heriot-Watt University, Edinburgh, as a Post-Doctoral Researcher. He conducts his research within the Institute of Sensors, Signals and Systems, School of Engineering and Physical Sciences. His current research activities focus on statistical signal and image processing, with a particular interest in Bayesian inverse problems with applications to remote sensing and biomedical imaging.

Nicolas Dobigeon (S'05–M'08–SM'13) was born in Angoulême, France, in 1981. He received the Engineering degree in electrical engineering from Ecole Nationale Supérieure d'Electronique, d'Electrotechnique, d'Informatique, d'Hydraulique et des Télécommunications (ENSEEIHT), Toulouse, France, and the M.Sc. degree in signal processing from the National Polytechnic Institute of Toulouse (INP Toulouse), both in 2004, and the Ph.D. degree and Habilitation Diriger des Recherches degree in signal processing from INP Toulouse in 2007 and 2012, respectively. From 2007 to 2008, he was a Post-Doctoral Research Associate with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor. Since 2008, He has been at INP Toulouse, University of Toulouse, where he is currently an Associate Professor. He conducts his research within the Signal and Communications Group, IRIT Laboratory, and he is also an Affiliated Faculty Member of the Telecommunications for Space and Aeronautics Co-Operative Laboratory. His recent research activities have been focused on 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 received the Ingénieur degree in electrical engineering from Ecole Nationale Supérieure d'Electronique, d'Electrotechnique, d'Informatique, d'Hydraulique et des Télécommunications (ENSEEIHT), Toulouse, and the Ph.D. degree from the National Polytechnic Institute, Toulouse, in 1989 and 1992, respectively. He is currently a Professor with at ENSEEIHT, University of Toulouse, 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 methods. He has been involved in the organization of several conferences, including the 2002 European Conference on Signal Processing (Program Chair), the 2006 International Conference on Acoustics, Speech and Signal Processing (plenaries), the 2012 Statistical Signal Processing (SSP) Workshop (international liaisons), the 2013 International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (local arrangement), and the 2014 SSP Workshop (special sessions). He has been the Co-General Chair of the CIMI Workshop on Optimization and Statistics in Image Processing held in Toulouse in 2013. He has been a member of different technical committees, including the Signal Processing Theory and Methods Committee of the IEEE Signal Processing Society, since 2010 as well as from 2001 to 2007. He was an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING from 2008 to 2011 and has been serving as an Associate Editor for the *EURASIP Journal on Signal Processing* since 2013.