Int. J. Appl. Math. Comput. Sci., 2014, Vol. 24, No. 2, 233–247
DOI: 10.2478/amcs-2014-0017

REGULARIZED NONNEGATIVE MATRIX FACTORIZATION: GEOMETRICAL INTERPRETATION AND APPLICATION TO SPECTRAL UNMIXING

RAFAŁ ZDUNEK

Department of Electronics
Wrocław University of Technology, ul. Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland
e-mail: rafal.zdunek@pwr.wroc.pl

Nonnegative Matrix Factorization (NMF) is an important tool in data spectral analysis. However, when a mixing matrix or sources are not sufficiently sparse, NMF of an observation matrix is not unique. Many numerical optimization algorithms, which assure fast convergence for specific problems, may easily get stuck into unfavorable local minima of an objective function, resulting in very low performance. In this paper, we discuss the Tikhonov regularized version of the Fast Combinatorial NonNegative Least Squares (FC-NNLS) algorithm (proposed by Benthem and Keenan in 2004), where the regularization parameter starts from a large value and decreases gradually with iterations. A geometrical analysis and justification of this approach are presented. The numerical experiments, carried out for various benchmarks of spectral signals, demonstrate that this kind of regularization, when applied to the FC-NNLS algorithm, is essential to obtain good performance.

Keywords: blind source separation, nonnegative matrix factorization, active-set algorithm, regularized NMF, polytope approximation.

1. Introduction

Nonnegative Matrix Factorization (NMF) decomposes an input matrix into lower-rank factors that have nonnegative values and usually some physical meaning or interpretation. Hence, it has already found diverse applications in data spectral analysis, mostly as a tool for blind unmixing or extraction of pure spectra (endmembers) from observed noisy mixtures. Examples include Raman scattering (e.g., Sajda *et al.*, 2003; Li *et al.*, 2007; Miron *et al.*, 2011), hyperspectra unmixing (e.g., Miao and Qi, 2007; Zymnis *et al.*, 2007; Zhang *et al.*, 2008; Jia and Qian, 2009; Guo *et al.*, 2009; Huck *et al.*, 2010; Chan *et al.*, 2011; Heylen *et al.*, 2011; Qian *et al.*, 2011; Iordache *et al.*, 2011; 2012; Plaza *et al.*, 2012; Bioucas-Dias *et al.*, 2012; Zdunek, 2012), spectral unmixing in microscopy (e.g., Pengo *et al.*, 2010), chemical shift imaging (e.g., Sajda *et al.*, 2004), reflectance spectroscopy (e.g., Pauca *et al.*, 2006; Hamza and Brady, 2006), fluorescence spectroscopy (e.g., Gobinet *et al.*, 2004), two-photon spectroscopic analysis (e.g., Hancewicz and Wang, 2005), astrophysical ice spectra unmixing (e.g., Igual *et al.*, 2006; Igual and Llinares, 2008; Llinares *et al.*, 2010), and gas chromatography-mass spectrometry (e.g., Likic, 2009).

Various numerical optimization algorithms have been successfully applied for NMF. Probably the most popular ones are based on multiplicative updates (Lee and Seung, 1999) that assure a monotonic convergence but only with a linear rate. To tackle the slow convergence problem, several algorithms with additive updates have been proposed, including Projected Gradient (PG) descent (Lin, 2007), Alternating Least Squares (ALS) (e.g., Berry *et al.*, 2007) and active-set algorithms (Kim *et al.*, 2007; Kim and Park, 2008; 2011). A survey of PG and ALS algorithms used for NMF is presented by Cichocki *et al.* (2009). Since the pure spectra or their abundance/concentration profiles are expected to be large and sparse, a good choice seems to be active-set algorithms.

In this paper, we discuss the selected active-set algorithms that are inspired by the NonNegative Least Squares (NNLS) algorithm, proposed by Lawson and Hanson (1974). The solution estimated by the NNLS algorithm is proved to be optimal according to the Karush–Kuhn–Tucker (KKT) conditions. Many research works have reported its usefulness in diverse research areas (e.g., Chen and Plemmons, 2009; Garda and Galias, 2012). Bro and Jong (1997) considerably accelerated the NNLS algorithm by rearranging computations for cross-product matrices.

Kim and Park (2008) applied its modified version to the l_1- and l_2-norm regularized Least Squares (LS) problems in NMF, and showed that such algorithms work very efficiently for gene expression microarrays. Their approach assumes constant regularization parameters to enforce the desired degree of sparsity.

Unfortunately, the basic NNLS algorithms are not very efficient for solving nonnegatively constrained linear systems with multiple Right-Hand Side (RHS) vectors since they compute a complete pseudoinverse once for each RHS. To tackle this problem, Benthem and Keenan (2004) devised the Fast Combinatorial NNLS (FC-NNLS) algorithm and experimentally demonstrated it works efficiently for energy-dispersive X-ray spectroscopy data.

Zdunek (2011) noticed that the regularization parameter in the l_2 regularized version of the FC-NNLS algorithm should decrease gradually with iterations to enforce a given character of iterative updates. This observation and the efficiency of using the FC-NNLS algorithm in hyperspectral imaging (Zdunek, 2012) motivate the study presented here. In this approach, we give the geometrical interpretation and justification of the proposed method, and extend it to spectral signal unmixing. NMF updates obtained with the regularized FC-NNLS algorithm are analyzed from a geometric point of view. The updates projected onto a probabilistic simplex form convex polytopes. The proposed method attempts to estimate probable positions of vertices of the convex polytope generated by observations, even for moderately noisy data. We have also improved the original implementation of the FC-NNLS algorithm.

The paper is organized in the following way. The next section discusses the concept of NMF for spectra unmixing. Section 3 is concerned with NNLS algorithms. The geometrical analysis is included in Section 4. The experiments are presented in Section 5. Finally, the conclusions are given in Section 6.

2. NMF for spectra recovering

The aim of NMF is to find lower-rank nonnegative matrices $\boldsymbol{A} = [a_{ij}] \in \mathbb{R}_+^{I\times J}$ and $\boldsymbol{X} = [x_{jt}] \in \mathbb{R}_+^{J\times T}$ such that $\boldsymbol{Y} = [y_{it}] \cong \boldsymbol{AX} \in \mathbb{R}_+^{I\times T}$, given the matrix $\boldsymbol{Y}$, the lower rank J (the number of pure spectra), and possibly *a priori* knowledge on the matrices $\boldsymbol{A}$ and $\boldsymbol{X}$. The set of nonnegative real numbers is denoted by $\mathbb{R}_+$. Assuming each row vector of $\boldsymbol{Y}$ represents an observed mixed spectrum, and J is an *a priori* known number of pure spectra, we can interpret each row vector of $\boldsymbol{X}$ as an unknown constituent pure spectrum (endmember), and the corresponding column vector of $\boldsymbol{A} = [\boldsymbol{a}_1, \ldots, \boldsymbol{a}_J]$ as abundance/concentration of the constituent material.

To estimate the matrices $\boldsymbol{A}$ and $\boldsymbol{X}$ from $\boldsymbol{Y}$, we assume the Tikhonov regularized Euclidean function:

$$D(\boldsymbol{Y}||\boldsymbol{AX}) = \frac{1}{2}||\boldsymbol{Y} - \boldsymbol{AX}||_F^2 + \frac{\lambda}{2}||\boldsymbol{A}||_F^2, \tag{1}$$

where λ is a regularization parameter and $||\cdot||_F$ denotes the Frobenious norm. The function (1) is convex with respect to only one set of arguments, i.e., the matrix $\boldsymbol{A}$ or $\boldsymbol{X}$, but it is not jointly convex. Thus, the solution to the nonconvex optimization problem:

$$\{\boldsymbol{A}^*, \boldsymbol{X}^*\} = \arg\min_{\mathbf{A},\mathbf{X}} D(\boldsymbol{Y}||\boldsymbol{AX}), \quad \text{subject to } \boldsymbol{A}, \boldsymbol{X} \geq 0, \tag{2}$$

is approximated by solving alternately the following nonnegatively constrained convex subproblems:

$$\min_{x_{jt}\geq 0} D(\boldsymbol{Y}||\boldsymbol{AX}), \quad \min_{a_{ij}\geq 0} D(\boldsymbol{Y}||\boldsymbol{AX}). \tag{3}$$

The Lagrangian functional associated with (3) can be written as

$$\mathcal{L}(\boldsymbol{A}, \boldsymbol{X}, \boldsymbol{\Lambda}_A, \boldsymbol{\Lambda}_X) = \frac{1}{2}||\boldsymbol{Y} - \boldsymbol{AX}||_F^2 + \frac{\lambda}{2}||\boldsymbol{A}||_F^2 - \operatorname{tr}\left(\boldsymbol{\Lambda}_A^T\boldsymbol{A}\right) - \operatorname{tr}\left(\boldsymbol{\Lambda}_X^T\boldsymbol{X}\right),$$

where $\boldsymbol{\Lambda}_A = [\lambda_{ij}^{(A)}] \in \mathbb{R}^{I\times J}$ and $\boldsymbol{\Lambda}_X = [\lambda_{jt}^{(X)}] \in \mathbb{R}^{J\times T}$ are the matrices of Lagrangian multipliers. The optimal solution $(\boldsymbol{A}^*, \boldsymbol{X}^*)$ satisfies the KKT first-order optimality conditions:

$$\nabla_{\boldsymbol{A}}\mathcal{L}(\boldsymbol{A}^*, \boldsymbol{X}^*, \boldsymbol{\Lambda}_A^*, \boldsymbol{\Lambda}_X^*) = \mathbf{0}, \tag{4}$$
$$\nabla_{\boldsymbol{X}}\mathcal{L}(\boldsymbol{A}^*, \boldsymbol{X}^*, \boldsymbol{\Lambda}_A^*, \boldsymbol{\Lambda}_X^*) = \mathbf{0}, \tag{5}$$

$$a_{ij}^* > 0 \quad \text{and} \quad (\lambda_{ij}^*)^{(A)} = 0, \tag{6}$$
$$a_{ij}^* = 0 \quad \text{and} \quad (\lambda_{ij}^*)^{(A)} \geq 0, \tag{7}$$
$$x_{jt}^* > 0 \quad \text{and} \quad (\lambda_{jt}^*)^{(X)} = 0, \tag{8}$$
$$x_{jt}^* = 0 \quad \text{and} \quad (\lambda_{jt}^*)^{(X)} \geq 0, \tag{9}$$

and the complementarity slackness conditions

$$a_{ij}^*(\lambda_{ij}^*)^{(A)} = 0, \qquad x_{jt}^*(\lambda_{jt}^*)^{(X)} = 0. \tag{10}$$

For the objective function (1), we have

$$\boldsymbol{\Lambda}_A = \nabla_{\boldsymbol{A}} D(\boldsymbol{Y}||\boldsymbol{AX}) = (\boldsymbol{AX} - \boldsymbol{Y})\boldsymbol{X}^T + \lambda\boldsymbol{A} \in \mathbb{R}^{I\times J}$$

and

$$\boldsymbol{\Lambda}_X = \nabla_{\boldsymbol{X}} D(\boldsymbol{Y}||\boldsymbol{AX}) = \boldsymbol{A}^T(\boldsymbol{AX} - \boldsymbol{Y}) \in \mathbb{R}^{J\times T}.$$

In the remainder, we will discuss the methods for solving the first problem in (3), i.e., the minimization with respect to $\boldsymbol{X}$, assuming that both problems are symmetric in arguments, and $\boldsymbol{A}$ can be estimated by solving the transposed system $\boldsymbol{X}^T\boldsymbol{A}^T = \boldsymbol{Y}^T$ with the same or other suitable method.

3. NNLS algorithms

The NNLS algorithm, which was originally proposed by Lawson and Hanson (1974), is given by Algorithm 1. It iteratively partitions the unknown variables into the basic variables that are strictly positive and the nonbasic ones that should satisfy the active constraints (zero-values).

Let $P = \{j : x_{jt} > 0\}$ be a passive set that contains indices of the basic variables, and $R = \{1, \ldots, J\} \backslash P$ be an active set with indices of the remaining (not necessary active) variables. In consequence, we assume the following partitions: $\forall t : \boldsymbol{x}_t = [\boldsymbol{x}_t^{(P)}; \boldsymbol{x}_t^{(R)}]^T \in \mathbb{R}^J$ and $\boldsymbol{g}_t = [g_{jt}] = \nabla_{\boldsymbol{x}_t} D(\boldsymbol{y}_t || \boldsymbol{A}\boldsymbol{x}_t) = [\boldsymbol{g}_t^{(P)}; \boldsymbol{g}_t^{(R)}]^T \in \mathbb{R}^J$, where $\boldsymbol{x}_t$ is the t-th column of $\boldsymbol{X}$. The columns of $\boldsymbol{A}$ can be also partitioned in a similar way: $\boldsymbol{A} = [\boldsymbol{A}_P \ \boldsymbol{A}_R]$, where $\boldsymbol{A}_P = [a_{*,P}]$ and $\boldsymbol{A}_R = [a_{*,R}]$. Algorithm 1 starts from $\boldsymbol{x}_t = \boldsymbol{0}$ and recursively updates the basic variables according to the KKT conditions. Note that, if $\exists m : g_{mt} < -\tau$ for any threshold $\tau > 0$, then x_{mt} cannot be a nonbasic variable. Hence, the index m is moved from the set R to P (Line 6), and the basic variables are then updated by solving the unconstrained LS problem (line 9):

$$\bar{\boldsymbol{x}}_t^{(P)} = \arg\min_{\mathbf{x}_t^{(P)}} \left\{ ||\boldsymbol{y}_t - \boldsymbol{A}_P \boldsymbol{x}_t^{(P)}||_2 \right\}, \tag{11}$$

where $\boldsymbol{A}_P$ has full column rank. Note that the update $\bar{\boldsymbol{x}}_t^{(P)}$ may cross the border of the feasible region, if the step length along the gradient $\boldsymbol{g}_t^{(P)}$ is too long. If so, the maximum step length α_t along the search direction $\boldsymbol{p}_t = \alpha_t(\bar{\boldsymbol{x}}_t^{(P)} - \boldsymbol{x}_t^{(P)})$ is determined. The update $\boldsymbol{x}_t^{(P)} \longleftarrow \boldsymbol{x}_t^{(P)} + \boldsymbol{p}_t$ moves the estimate to the border of the feasible region. Thus some variables become nonbasic, which involves the update of the sets P and R (Line 14), and the problem (11) is recomputed. All the variables whose indices belong to the set R are set to a zero-value.

As mentioned by Lawson and Hanson (1974), the inner loop of Algorithm 1 requires no more than $|P| - 1$ iterations. The number of iterations in the main loop depends on the sparsity of the solution, and it should not be greater than $|P|$ for $\boldsymbol{x}_t^*$.

Bro and Jong (1997) considerably speed up this algorithm for $I \gg J$ by precomputing the normal matrix $\boldsymbol{A}^T\boldsymbol{A}$ and the vector $\boldsymbol{A}^T\boldsymbol{y}_t$, and then solving the problem (11) as follows:

$$\bar{\boldsymbol{x}}_t^{(P)} = \left((\boldsymbol{A}^T\boldsymbol{A})_{P,P} \right)^{-1} (\boldsymbol{A}^T\boldsymbol{y}_t)_P. \tag{12}$$

Unfortunately, the inverse of $(\boldsymbol{A}^T\boldsymbol{A})_{P,P}$ must be computed for each t, which is still very expensive if the number of RHSs is very large.

Benthem and Keenan (2004) proposed the FC-NNLS algorithm to tackle the problem of a high computational cost of Algorithm 1 for multiple RHSs. They noticed that, for a sparse solution with multiple column vectors, a probability of finding column vectors that have the same layout of zero-entries (active constraints) is high. Hence, after detecting basic variables for each $\boldsymbol{x}_t$, the passive entries in the vectors $\{\boldsymbol{x}_t\}$ that have an identical sparsity profile (a common passive set) are updated by computing the inverse of $(\boldsymbol{A}^T\boldsymbol{A})_{P,P}$ only once.

Kim and Park (2008) applied the NNLS algorithm to Tikhonov regularized NMF problems, where the penalty terms are formed using the l_1- and/or l_2-norm constraints. They showed that the following problem:

$$\min_{\mathbf{A},\mathbf{X}} \left\{ \frac{1}{2}||\boldsymbol{Y} - \boldsymbol{A}\boldsymbol{X}||_F^2 + \frac{\lambda}{2}||\boldsymbol{A}||_F^2 + \frac{\beta}{2}\sum_{t=1}^{T}||\boldsymbol{x}_t||_1^2 \right\}, \tag{13}$$

Algorithm 1: LH-NNLS.

Input : $\boldsymbol{A} \in \mathbb{R}^{I \times J}$, $\boldsymbol{y}_t \in \mathbb{R}^I$
Output: $\boldsymbol{x}_t^* \geq 0$ such that $\boldsymbol{x}_t^* = \arg\min_{\mathbf{x}_t} ||\boldsymbol{y}_t - \boldsymbol{A}\boldsymbol{x}_t||_2$

1 **Initialization:** $P = \emptyset$, $R = \{1, \ldots, J\}$, $\boldsymbol{x}_t = \boldsymbol{0}$, $\boldsymbol{g}_t = -\boldsymbol{A}^T\boldsymbol{y}_t$, $k = 0$;
2 **repeat**
3 $k \longleftarrow k + 1$;
4 $m = \arg\min_{j \in R}\{g_{jt}\}$; `// the constraint to add`
5 **if** $R \neq \emptyset$ and $g_{mt} < -\tau$ **then**
6 $P \longleftarrow P \cup m$, and $R \longleftarrow R \backslash m$; `// updates of the passive and active sets`
7 **else**
8 **stop** with $\boldsymbol{x}_t$ as an optimal solution
9 $\bar{\boldsymbol{x}}_t^{(P)} = \left((\boldsymbol{A}_P)^T\boldsymbol{A}_P\right)^{-1}(\boldsymbol{A}_P)^T\boldsymbol{y}_t$ where $\boldsymbol{A}_P = [a_{*,P}] \in \mathbb{R}^{I \times |P|}$;
10 **while** $\min\{\bar{\boldsymbol{x}}_t^{(P)}\} \leq 0$ **do**
11 $\alpha_t = \min_{\substack{j \in P \\ \bar{x}_{jt}^{(P)} \leq 0}} \left\{ \frac{x_{jt}^{(P)}}{x_{jt}^{(P)} - \bar{x}_{jt}^{(P)}} \right\}$; `// steplength`
12 $\boldsymbol{x}_t^{(P)} \longleftarrow \boldsymbol{x}_t^{(P)} + \alpha_t(\bar{\boldsymbol{x}}_t^{(P)} - \boldsymbol{x}_t^{(P)})$;
13 $N = \left\{ j : x_{jt}^{(P)} = 0 \right\}$; `// the constraints to drop`
14 $P \longleftarrow P \backslash N$, and $R \longleftarrow R \cup N$;
15 $\bar{\boldsymbol{x}}_t^{(P)} = \left((\boldsymbol{A}_P)^T\boldsymbol{A}_P\right)^{-1}(\boldsymbol{A}_P)^T\boldsymbol{y}_t$ where $\boldsymbol{A}_P = [a_{*,P}] \in \mathbb{R}^{I \times |P|}$;
16 $\boldsymbol{x}_t \longleftarrow \left[\boldsymbol{x}_t^{(P)};\ \boldsymbol{x}_t^{(R)}\right]^T \in \mathbb{R}_+^J$ where $\boldsymbol{x}_t^{(P)} = \bar{\boldsymbol{x}}_t^{(P)}$ and $\boldsymbol{x}_t^{(R)} = \boldsymbol{0}$;
17 $\boldsymbol{g}_t = \boldsymbol{A}^T(\boldsymbol{A}_P\boldsymbol{x}_t^{(P)} - \boldsymbol{y}_t)$; `// gradient`
18 **until** $k > k_{\max}$;

Algorithm 2: RFC-NNLS.

Input : $\boldsymbol{A} \in \mathbb{R}^{I\times J}$, $\boldsymbol{Y} \in \mathbb{R}^{I\times T}$, $\lambda \geq 0$
Output: $\boldsymbol{X}^* \geq \mathbf{0}$ such that $\boldsymbol{X}^* = \arg\min_{\mathbf{X}} \left\{\frac{1}{2}||\boldsymbol{Y}-\boldsymbol{A}\boldsymbol{X}||_F^2 + \frac{\lambda}{2}||\boldsymbol{X}||_F^2\right\}$

Initialization: $M = \{1,\ldots,T\}$, $N = \{1,\ldots,J\}$;
Precompute: $\boldsymbol{B} = [b_{ij}] = \boldsymbol{A}^T\boldsymbol{A} + \lambda \boldsymbol{I}_J$ and $\boldsymbol{C} = [c_{it}] = \boldsymbol{A}^T\boldsymbol{Y}$;
$\boldsymbol{X} = \boldsymbol{B}^{-1}\boldsymbol{C}$; `// unconstrained min.`
$\boldsymbol{P} = [p_{jt}]$, where $p_{jt} = \begin{cases} 1 & \text{if } x_{jt} > 0, \\ 0 & \text{otherwise} \end{cases}$; `// passive set`
$F = \{t \in M : \sum_j p_{jt} < I\}$; `// columns to be optimized or verified`
$x_{jt} \longleftarrow \begin{cases} x_{jt} & \text{if } p_{jt} = 1, \\ 0 & \text{otherwise} \end{cases}$;
while $F \neq \emptyset$ **do**
  $\boldsymbol{P}_F = [p_{*,F}] \in \mathbb{R}^{J\times|F|}$,
  $\boldsymbol{C}_F = [c_{*,F}] \in \mathbb{R}^{J\times|F|}$;
  $[\bar{x}_{*,F}] = \texttt{cssls}(\boldsymbol{B}, \boldsymbol{C}_F, \boldsymbol{P}_F)$; `// basic variable update with the CSSLS`
  $H = \{t \in F : \min_{j\in N}\{\bar{x}_{jt}\} < 0\}$;
  `// columns with neg. vars.`
  while $H \neq \emptyset$ and $k \leq k_{inner}$ **do**
    $S_t = \{j : \bar{x}_{jt} < 0 \text{ and } p_{jt} > 0, j \in N, t \in H\}$;
    $\alpha_{jt} = \dfrac{x_{jt}}{x_{jt} - \bar{x}_{jt}}$ for $j \in S_t$ and $t \in H$;
    $j_t = \arg\min_{j\in S_t}\{\alpha_{jt}\}$ for $t \in H$;
    `// constraint to drop`
    $x_{*,t} \longleftarrow x_{*,t} + \alpha_t^*(\bar{x}_{*,t} - x_{*,t})$, where $\alpha_t^* = \alpha_{j_t,t}$ for $t \in H$;
    $x_{j_t,t} = 0, p_{j_t,t} = 0$ for $t \in H$;
    `// active entries`
    $\boldsymbol{P}_H = [p_{*,H}] \in \mathbb{R}^{J\times|H|}$,
    $\boldsymbol{C}_H = [c_{*,H}] \in \mathbb{R}^{J\times|H|}$;
    $[\bar{x}_{*,H}] = \texttt{cssls}(\boldsymbol{B}, \boldsymbol{C}_H, \boldsymbol{P}_H)$;
    $H = \{t \in F : \min_{j\in N}\{\bar{x}_{jt}\} < 0\}$;
    `// columns with neg. vars.`
  $\boldsymbol{W}_F = [w_{*,F}] = \boldsymbol{C}_F - \boldsymbol{B}\bar{\boldsymbol{X}}_F$, where $\bar{\boldsymbol{X}}_F = [\bar{x}_{*,F}]$; `// negative gradient`
  $J = \{t \in F : w_{jt}(1-p_{jt}) \leq 0, \quad \forall j \in N\}$; `// optimized columns`
  $F \longleftarrow F\backslash J$; `// col. to be optim.`
  if $F \neq 0$ **then**
    $[x_{*,F}] = [\bar{x}_{*,F}],\ z_{jt} = w_{jt}(1-p_{jt})$;
    $p_{jt} = \begin{cases} 1 & \text{if } j = \arg\max_{j\in N}\{z_{jt},\ \forall t \in F\}, \\ p_{jt} & \text{otherwise} \end{cases}$

subject to $\boldsymbol{A}, \boldsymbol{X} \geq \mathbf{0}$, can be solved with any NNLS algorithm by applying it to the reformulated alternating minimization subproblems:

$$\min_{\mathbf{X}\geq\mathbf{0}} \left|\left| \begin{pmatrix} \boldsymbol{A} \\ \sqrt{\beta}\mathbf{1}_{1\times J} \end{pmatrix} \boldsymbol{X} - \begin{pmatrix} \boldsymbol{Y} \\ \mathbf{0}_{1\times T} \end{pmatrix} \right|\right|_F^2, \tag{14}$$

$$\min_{\mathbf{A}\geq\mathbf{0}} \left|\left| \begin{pmatrix} \boldsymbol{X}^T \\ \sqrt{\lambda}\boldsymbol{I}_J \end{pmatrix} \boldsymbol{A}^T - \begin{pmatrix} \boldsymbol{Y}^T \\ \mathbf{0}_{J\times I} \end{pmatrix} \right|\right|_F^2, \tag{15}$$

where $\boldsymbol{I}_J \in \mathbb{R}^{J\times J}$ is an identity matrix, $\mathbf{1}_{1\times J}$ is a row vector of all ones, and $\mathbf{0}_{J\times I}$ is a zero matrix of size $J \times I$. In their approach, the parameters λ and β are set to fixed small positive values. The parameter λ is used to suppress the growth of $||\boldsymbol{A}||_F$, and the parameter β controls the sparsity in $\boldsymbol{X}$. Moreover, the parameter λ with a small value assures full rank of the matrix $\boldsymbol{X}\boldsymbol{X}^T$ when applied for updating the matrix $\boldsymbol{A}$.

Contrary to Kim and Park (2008) and motivated by Zdunek and Cichocki (2007), we propose to gradually decrease the regularization parameter λ with alternating iterations for updating the matrix $\boldsymbol{A}$, starting from a large value $\lambda_0 > 0$. The parameter can be changed according to the following rule: $\lambda = \max\left\{\bar{\lambda}, 2^{-k}\lambda_0\right\}$, where $k = 0, 1, \ldots$ is the current alternating step, and $\bar{\lambda} > 0$ determines the lowest value of this parameter. The aim of using such regularization for updating $\boldsymbol{A}$ is to enforce a certain character of iterations, rather than to stabilize ill-posed problems since the matrix $\boldsymbol{X}$ is not expected to be severely ill-conditioned. When λ is large, the updates can be regarded as gradient descent ones with a small stepsize, and when the alternating iterations proceed, the update are determined by the Newton step. When $\lambda \cong \bar{\lambda}$, the proposed NMF algorithm is identical with SNMF/R given by Kim and Park (2008), and the parameter $\bar{\lambda}$ plays the role of the standard Tikhonov regularization. In the next sec-

Algorithm 3: CSSLS.

Input : $\boldsymbol{B} \in \mathbb{R}^{J\times J}$, $\boldsymbol{C} \in \mathbb{R}^{J\times K}$, $\boldsymbol{P} \in \mathbb{R}^{J\times K}$
Output: $\boldsymbol{X} \in \mathbb{R}^{J\times K}$

$M = \{1,\ldots,K\}, N = \{1,\ldots,J\}$, $\boldsymbol{P} = [\boldsymbol{p}_1,\ldots,\boldsymbol{p}_K]$;
Find the set of L unique columns in $\boldsymbol{P}$: $\boldsymbol{U} = [\boldsymbol{u}_1,\ldots,\boldsymbol{u}_L] = \text{unique}\{\boldsymbol{P}\}$;
for $j = 1,\ldots,L$ **do**
  $\boldsymbol{h}_j = \{t \in M : \boldsymbol{p}_t = \boldsymbol{u}_j\}$; `// indices of columns with identical passive sets`
  $x_{\boldsymbol{u}_j,\boldsymbol{h}_j} = \left([\boldsymbol{B}]_{\boldsymbol{u}_j,\boldsymbol{u}_j}\right)^{-1}[\boldsymbol{C}]_{\boldsymbol{u}_j,\boldsymbol{h}_j}$; `// back-substitution with submatrices`

Algorithm 4: RASLS-NMF.

Input : $\boldsymbol{Y} \in \mathbb{R}_+^{I \times T}$, J - lower rank, λ_0 - initial regularization parameter, $\bar{\lambda}$ - minimal value of regularization parameter,
Output: Factors $\boldsymbol{A}$ and $\boldsymbol{X}$

1 Initialize (randomly) $\boldsymbol{A}$ and $\boldsymbol{X}$, $k = 0$;
2 **repeat**
3 $k \longleftarrow k + 1$;
4 $\lambda = \max\left\{\bar{\lambda}, 2^{-k}\lambda_0\right\}$; `// Regular. param. schedule`
5 $\boldsymbol{X} \longleftarrow$ `RFCNNLS`$(\boldsymbol{A}, \boldsymbol{Y}, 10^{-12})$; `// Update for` $\boldsymbol{X}$
6 $d_j^{(X)} = \sum_{t=1}^{T} x_{jt}$;
7 $\boldsymbol{X} \longleftarrow \operatorname{diag}\left\{(d_j^{(X)})^{-1}\right\} \boldsymbol{X}$;
8 $\boldsymbol{A} \longleftarrow \boldsymbol{A} \operatorname{diag}\left\{d_j^{(X)}\right\}$;
9 $\bar{\boldsymbol{A}} \longleftarrow$ `RFCNNLS`$(\boldsymbol{X}^T, \boldsymbol{Y}^T, \lambda)$; `// Update for` $\boldsymbol{A}^T$
10 $\boldsymbol{A} = \bar{\boldsymbol{A}}^T$;
11 $d_j^{(A)} = \sum_{i=1}^{I} a_{ij}$;
12 $\boldsymbol{X} \longleftarrow \operatorname{diag}\left\{d_j^{(A)}\right\} \boldsymbol{X}$;
13 $\boldsymbol{A} \longleftarrow \boldsymbol{A} \operatorname{diag}\left\{(d_j^{(A)})^{-1}\right\}$
14 **until** `Stop criterion` is satisfied ;

tion, we give a geometrical interpretation and justification of the proposed approach.

Regularized Active-Set Least-Squares (RASLS) NMF that is based on the regularized FC-NNLS algorithm with $\beta = 0$ is given by Algorithm 4. Both factors $\boldsymbol{A}$ and $\boldsymbol{X}$ are updated with Algorithms 2 and 3, but only the estimate of the factor $\boldsymbol{A}$ is regularized with a varying regularization parameter. For updating the matrix $\boldsymbol{X}$, the regularization parameter is set to a small value, e.g., $\lambda = 10^{-12}$, to avoid numerical instabilities.

4. Geometrical interpretation

Exact nonnegative matrix factorization, i.e., $\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{X}$, means that each column vector in $\boldsymbol{Y}$ is a convex combination of the nonnegative column vectors in $\boldsymbol{A}$. The vectors $\{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_J\}$ form the simplicial cone (Donoho and Stodden, 2004) in $\mathbb{R}^I$ that lies inside the nonnegative orthant $\mathbb{R}_+^I$. If the column vectors of $\boldsymbol{A}$ are linearly independent, then the simplicial cone is generated by these vectors that are referred to as the extreme rays.

Definition 1. The $(I-1)$-dimensional *probability simplex* $\mathcal{S}^{(I-1)} = \{\boldsymbol{y} = [y_i] \in \mathbb{R}_+^I : y_i \geq 0, \mathbf{1}_I^T \boldsymbol{y} = 1\}$ contains all the points of $\mathbb{R}_+^I$ located onto the hyperplane $\Pi : ||\boldsymbol{y}||_1 = 1$. Its vertices are determined by the versors (unit vectors) of the Cartesian coordinate system.

Definition 2. The matrix $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T] \in \mathbb{R}_+^{J \times T}$ is *sufficiently sparse* if there exists a square diagonal full-rank submatrix $\tilde{\boldsymbol{X}} \in \mathbb{R}_+^{J \times J}$ created from a subset of its column vectors.

The intersection of the directional rays determined by a columns of $\boldsymbol{Y}$ and $\mathcal{S}^{(I-1)}$ can be expressed as

$$\mathcal{P}_{\mathcal{S}^{(I-1)}}(\boldsymbol{Y}) = \bar{\boldsymbol{Y}} = \left\{\frac{\boldsymbol{y}_1}{||\boldsymbol{y}_1||_1}, \ldots, \frac{\boldsymbol{y}_T}{||\boldsymbol{y}_T||_1}\right\}. \quad (16)$$

The intersection can be also regarded as the special projection of the nonzero columns in $\boldsymbol{Y}$ onto $\mathcal{S}^{(I-1)}$ along the directional rays. The points in $\mathcal{P}_{\mathcal{S}^{(I-1)}}(\boldsymbol{Y})$ form the convex polytope $\mathcal{C}(\boldsymbol{Y})$ (Chu and Lin, 2008). If the matrix $\boldsymbol{X}$ is sufficiently sparse (see Definition 2), the vertices of $\mathcal{C}(\boldsymbol{Y})$ correspond to those column vectors of $\boldsymbol{A}$ that intersect with $\mathcal{S}^{(I-1)}$. Any column vector $\bar{\boldsymbol{y}}_t$ whose the corresponding vector $\boldsymbol{x}_t$ contains at most two positive entries lies on the edge of the convex polytope $\mathcal{C}(\boldsymbol{Y})$.

The aim of NMF is to find these vertices or possible locations of the vertices if the observations are corrupted with noise. This task is not easy to achieve unless the vertices of $\mathcal{C}(\boldsymbol{Y})$ lie on the border of $\mathbb{R}_+^I$. If $\forall i, j : a_{ij} > 0$, all the vertices of $\mathcal{C}(\boldsymbol{Y})$ are located strictly inside $\mathbb{R}_+^I$. Assuming strict positivity for the entries of $\boldsymbol{A}$, NMF of $\boldsymbol{Y}$ is not unique although the matrix $\boldsymbol{X}$ is sufficiently sparse. In other words, there might exist a convex polytope $\mathcal{C}(\tilde{\boldsymbol{A}})$ such that $\mathcal{C}(\boldsymbol{Y}) \subset \mathcal{C}(\tilde{\boldsymbol{A}}) \subset \mathbb{R}_+^I$ whose vertices are not determined by the column vectors of $\boldsymbol{A}$, i.e., $\tilde{\boldsymbol{A}} \neq \boldsymbol{A}$ and $\tilde{\boldsymbol{A}} \in \mathbb{R}_+^{I \times J}$. Note that such $\mathcal{C}(\tilde{\boldsymbol{A}})$ does not come from the scale or permutation indeterminacies that are intrinsic ambiguities of NMF. To minimize the risk of convergence to such $\mathcal{C}(\tilde{\boldsymbol{A}})$, we assume that the updates for $\boldsymbol{A}$ should be Tikhonov regularized with a decreasing regularization parameter.

Let $\boldsymbol{A}_k$ denote the update of $\boldsymbol{A}$ in the k-th alternating step, and let $\boldsymbol{Y}^{(Z_k)} = [y_{*,Z_k}] \in \mathbb{R}_+^{I \times |Z_k|}$, where

$$Z_k = \left\{t : \sum_{j=1}^{J} p_{jt}^{(k)} = J\right\}$$

and $p_{jt}^{(k)}$ are the entries of the matrix $\boldsymbol{P}$ in Algorithm 2 used for updating the matrix $\boldsymbol{X}$. The RASLS-NMF algorithm for computing $\boldsymbol{X}$ at $k = 1$ is initialized with any $\boldsymbol{A}_0 \in \mathbb{R}_+^{I \times J}$. If $\mathcal{C}(\boldsymbol{A}_0) \cap \mathcal{C}(\boldsymbol{Y}) \neq \emptyset$, we have $\boldsymbol{Y}^{(Z_1)} = \{\boldsymbol{y} : \boldsymbol{y} \in \mathcal{C}(\boldsymbol{Y}) \wedge \boldsymbol{y} \in \mathcal{C}(\boldsymbol{A}_0)\}$ after computing $\boldsymbol{X}_1$. Then, the rows of $\boldsymbol{X}_1$ are scaled to the unit l_1-norm (see line 7 in Algorithm 4).

If the initial regularization parameter λ_0 is large enough, i.e., when the condition $\lambda_0 \gg \sigma_{\max}(\boldsymbol{X}_1)$ is satisfied, where $\sigma_{\max}(\boldsymbol{X}_1)$ is the largest singular value of $\boldsymbol{X}_1$, we have

$$\boldsymbol{A}_1 = \boldsymbol{Y}\boldsymbol{X}_1^T(\boldsymbol{X}_1\boldsymbol{X}_1^T + \lambda \boldsymbol{I}_J)^{-1} \cong \frac{1}{\lambda}\boldsymbol{Y}\boldsymbol{X}_1^T. \quad (17)$$

Since $\lambda > 0$, $\forall i,t : y_{it} \geq 0$ and $\forall j,t : x_{jt}^{(1)} \geq 0$, where $\boldsymbol{X}_1 = [x_{jt}^{(1)}]$, then $\forall i,j : a_{ij}^{(1)} \geq 0$ from (17). If so, $F = \emptyset$ in Algorithm 2, and this algorithm terminates after the first iteration. Moreover, due to the normalization $\forall j : \sum_{t=1}^{T} x_{jt}^{(1)} = 1$, the columns in the matrix $\boldsymbol{A}_1$ are a convex combination of the columns in $\boldsymbol{Y}$. Thus after the normalization in line 13 of Algorithm 4, the columns of the matrix $\boldsymbol{A}_1$ have the unit l_1-norm, and $\mathcal{C}(\boldsymbol{A}_1) \subseteq \mathcal{C}(\boldsymbol{Y})$. This means that the first iteration of Algorithm 4 ensures that the columns of $\boldsymbol{A}_1$ are located inside the polytope $\mathcal{C}(\boldsymbol{Y})$.

If $\lambda \gg \sigma_{\max}(\boldsymbol{X}_k)$ for $k > 1$, the consecutive iterations do not change the positions of the columns in the estimated matrix $\boldsymbol{A}_k$ noticeably. In consequence, we may observe a stagnation in the residual error versus iterations. When $\mathcal{C}(\boldsymbol{A}_k) \subseteq \mathcal{C}(\boldsymbol{Y})$, then each $\boldsymbol{y} \in \mathcal{C}(\boldsymbol{A}_k)$ also belongs to $\boldsymbol{Y}^{(Z_{k+1})}$, where

$$Z_{k+1} = \Big\{ t : \sum_{j=1}^{J} p_{jt}^{(k+1)} = J \Big\}.$$

All the column vectors of the matrix $\boldsymbol{X}^{(Z_{k+1})}$, where $\boldsymbol{X}^{(Z_{k+1})} = [x_{*,Z_{k+1}}] \in \mathbb{R}_+^{J \times |Z_{k+1}|}$, are computed in the first iteration of Algorithm 2 (line 3). The remaining columns of $\boldsymbol{X}_{k+1}$ are updated in the inner iterations (line 7). The columns that contain the active entries determine the border of $\mathcal{C}(\boldsymbol{A}_k)$

When $\lambda \approx \sigma_{\max}(\boldsymbol{X}_{k+1})$, the large singular values of $\boldsymbol{X}_{k+1}$ take part in updating $\boldsymbol{A}_{k+1}$, and the minimization of the objective function $D(\boldsymbol{Y}||\boldsymbol{A}\boldsymbol{X}_{k+1})$ with respect to $\boldsymbol{A}$ expands the volume of $\mathcal{C}(\boldsymbol{A}_{k+1})$. Algorithm 2 attempts to find a nonnegative minimizer of the objective function that has the lowest number of active entries. Note that Algorithm 2 does not find the sparsest solution but in the nested loop (lines 11–19) it tries to find the smallest set of the active entries to guarantee a nonnegative update. Hence, it is obvious that it tends to maximize the volume of $\mathcal{C}(\boldsymbol{A}_{k+1})$ because all the entries strictly inside $\mathcal{C}(\boldsymbol{A}_{k+1})$ belong to the passive set of the update.

According to Elden (1977) as well as Rojas and Steihaug (2002), the Tikhonov regularized LS problem with the objective function (1) is equivalent to the Trust-Region Subproblem (TRS):

$$\min_{\boldsymbol{A}} \frac{1}{2}||\boldsymbol{Y} - \boldsymbol{A}\boldsymbol{X}||_F^2, \tag{18}$$

subject to $||\boldsymbol{A}||_F^2 \leq \Delta$, where $\Delta > 0$ is the TR radius that is reversely related to the regularization parameter, e.g., when λ is large, Δ is small. Decreasing the regularization parameter λ, Δ increases, leading to the TR expansion. In consequence, more and more entries from $\boldsymbol{A}$ that were active in the previous alternating step move to the passive set. This increases the volume of $\mathcal{C}(\boldsymbol{A}_{k+1})$ until all the column vectors of $\boldsymbol{Y}$ belong to $\mathcal{C}(\boldsymbol{A})$. A gradual increase in the volume of $\mathcal{C}(\boldsymbol{A}_k)$ is needed to avoid convergence to such $\tilde{\boldsymbol{A}}$ for which $\mathcal{C}(\boldsymbol{Y}) \subset \mathcal{C}(\tilde{\boldsymbol{A}}) \subset \mathbb{R}_+^I$ and $\tilde{\boldsymbol{A}} \neq \boldsymbol{A}$. When the regularization parameter decreases, the nature of updates for $\boldsymbol{A}$ is changing from gradient descent steps to Newton steps that explore the local minimum deeply. Thus, the regularization parameter should diminish to a small value when the factorization is exact. For noisy data, a certain lower bound for λ should be assumed to exclude some outliers from $\mathcal{C}(\boldsymbol{A})$. In this case, the problem of selecting the right value of $\bar{\lambda}$ or the number of alternating steps can be considered in terms of the standard regularization approach to ill-posed problems (e.g., Hansen, 1998; Krawczyk-Stańdo and Rudnicki, 2007; Calvetti *et al.*, 2001).

The parameter λ_0 should satisfy the conditions

$$\lambda_0 \gg \sigma_{\max}(\boldsymbol{X}_1).$$

From Gelfand's formula, we have

$$\sigma_{\max}(\boldsymbol{Y}) \leq \sigma_{\max}(\boldsymbol{A})\sigma_{\max}(\boldsymbol{X}).$$

The normalization of the estimated factors leads to

$$\sigma_{\max}(\boldsymbol{A}) \geq 1$$

$$\sigma_{\max}(\boldsymbol{X}) \geq 1,$$

$$\sigma_{\max}(\boldsymbol{Y}) \geq \sigma_{\max}(\boldsymbol{X}).$$

Thus the assumption $\lambda_0 \gg \sigma_{\max}(\boldsymbol{Y})$ satisfies the condition $\lambda_0 \gg \sigma_{\max}(\boldsymbol{X})$. When the matrix $\boldsymbol{Y}$ is large, the direct computation of $\sigma_{\max}(\boldsymbol{Y})$ may involve a large computational cost. Since $\forall i,t : y_{it} \geq 0$ and $\boldsymbol{Y}$ is irreducible, from the Perron–Frobenius theorem we have

$$\lambda_{\max}(\boldsymbol{Y}\boldsymbol{Y}^T) \leq \gamma^{(1)}$$

$$\lambda_{\max}(\boldsymbol{Y}^T\boldsymbol{Y}) \leq \gamma^{(2)},$$

where

$$\gamma^{(1)} = \max_{1 \leq i \leq I} \sum_{m=1}^{I} [\boldsymbol{Y}\boldsymbol{Y}^T]_{im},$$

$$\gamma^{(2)} = \max_{1 \leq t \leq T} \sum_{n=1}^{T} [\boldsymbol{Y}^T\boldsymbol{Y}]_{tn},$$

and $\lambda_{\max}(\cdot)$ is the maximal eigenvalue of a matrix. Thus, in practice we may assume

$$\lambda_0 \geq \min\{\sqrt{\gamma^{(1)}}, \sqrt{\gamma^{(2)}}\}. \tag{19}$$

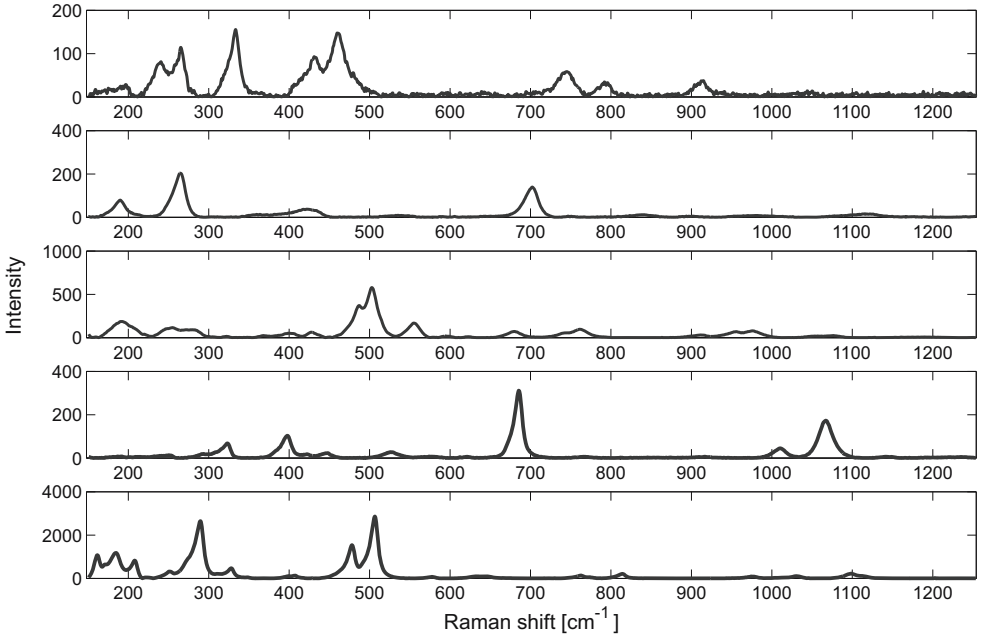

Fig. 1. Processed Raman spectra for the following minerals (from the top): Dickite, Muscovite, Anorthite, Beryl, Albite, measured at 780 nm.

5. Experiments

The numerical experiments are carried out for three different benchmarks of nonnegative signals. Benchmark A is created from five real Raman spectra taken from the $\mathrm{RRUFF}^{\mathrm{TM}}$ project[1]. We selected the spectra for the following minerals: Dickite from Stara Gora, Lower Silesia in Poland, Muscovite from California (USA), Anorthite from Miyakejima in Japan, Beryl from North Keystone in South Dakota (USA), and Albite from the Madawaska/Faraday mine, Bancroft, Ontario in Canada. Figure 1 illustrates these spectra in the following order from the top: Dickite, Muscovite, Anorthite, Beryl, Albite. All the signals in benchmark A are measured at 780 nm, and resampled to 2000 samples ($T = 2000$). Note that the signals differ in their magnitudes considerably, and the selected signals are highly correlated. Let $\rho(\boldsymbol{x}_i, \boldsymbol{x}_j)$ be the correlation coefficient between the i-th and the j-th signal. We have $\rho(\boldsymbol{x}_1, \boldsymbol{x}_2) = 0.26$ and $\rho(\boldsymbol{x}_3, \boldsymbol{x}_5) = 0.66$, and $|\rho(\boldsymbol{x}_i, \boldsymbol{x}_j)| < 0.01$ for the other pairs of signals. For such highly correlated source signals, Independent Component Analysis (ICA) techniques (e.g., Hyvrinen *et al.*, 2001; Dabrowski and Cetnarowicz, 2008; Siwek *et al.*, 2009; Makowski, 2003; Tong *et al.*, 2003) cannot be used. According to Definition 2, the matrix $\boldsymbol{X}$ created from the signals in benchmark A is sufficiently sparse.

Benchmark B contains 50 random positive spiky signals where the sparsity of each signal amounts to about 80%. The total number of samples is 1000, i.e., $T = 1000$.

Benchmark C is generated from 15 randomly selected spectral reflectance signals taken from the USGS library[2]. The spectra are measured with a 224-channel imaging AVIRIS spectrometer, covering the range $400 - -2500$ nm with the spectral resolution of about 10 nm. For the selected signals, the minimum angle between any two signals is larger than 10 degrees.

For benchmarks A and B the observed spectra are obtained by mixing the source spectra with a uniformly distributed random matrix $\boldsymbol{A}$, e.g., $\forall i, j : a_{ij} > 0$. In noisy scenarios, the observations are corrupted with an additive zero-mean Gaussian noise with the variance adopted to have a given Signal-to-Noise Ratio (SNR). We carry out a few numerical tests described below.

5.1. Test A. The aim of Test A is to experimentally verify the theoretical discussion, presented in Section 4. The mixed signals in $\boldsymbol{Y}$ are generated synthetically. The true sources are modeled by $\boldsymbol{X} = [x_{jt}] \in \mathbb{R}_+^{3\times 200}$, where $x_{jt} = \max\{0, \tilde{x}_{jt}\}$ and $\tilde{x}_{jt} \sim \mathcal{N}(0, 1)$. The true mixing matrix $\boldsymbol{A} \in \mathbb{R}_+^{3\times 3}$, generated from a uniform distribution and scaled to the unit l_1 norm columns, is given as follows:

$$\boldsymbol{A}^{(\mathrm{scaled})} = \begin{bmatrix} 0.6 & 0.03 & 0.09 \\ 0.15 & 0.8 & 0.32 \\ 0.25 & 0.17 & 0.59 \end{bmatrix}. \tag{20}$$

The column vectors $\boldsymbol{y}_t$ ($t = 1, \ldots, T$) create a geometrical object in the 3D observation space. The plane

[1] http://rruff.info.

[2] http://speclab.cr.usgs.gov/spectral.lib06/.

intersecting the points $(1,0,0)$, $(0,1,0)$ and $(0,0,1)$ of the Cartesian coordinates in the 3D space determines the probabilistic simplex $\mathcal{S}^{(2)}$ in $\mathbb{R}^3_+$, which is an equilateral triangle. Figures 2–4 show the analyzed vectors distributed on $\mathcal{S}^{(2)}$. The vectors $\{\boldsymbol{y}_t\}$ intersected with this place are marked as the small cross points onto $\mathcal{S}^{(2)}$. Note that these points form the triangle onto $\mathcal{S}^{(2)}$, but generally they will form a convex polytope $\mathcal{C}(\boldsymbol{Y})$ onto $\mathcal{S}^{(I-1)}$. Since the matrix $\boldsymbol{X}$ is sparse, many points $\boldsymbol{y}_t$ lie on the border of $\mathcal{C}(\boldsymbol{Y})$. The positions of the column vectors $\boldsymbol{a}_j^*$ $(j = 1, 2, 3)$ of the true mixing matrix $\boldsymbol{A}_{\rm true}$ are denoted by the filled pentagrams. The columns of the initial matrix $\boldsymbol{A}_0$, which form $\mathcal{C}(\boldsymbol{A}_0)$, are marked with the unfilled pentagrams. Note that $\mathcal{C}(\boldsymbol{A}_0) \cap \mathcal{C}(\boldsymbol{Y}) \neq \emptyset$, and all the points $\boldsymbol{y}_t$ that fall into the intersection of these sets are marked with the stars. After updating $\boldsymbol{X}_1$ with Algorithm 2, we found that all the points of $\boldsymbol{Y}^{(Z_1)}$ are the same as the points of $\mathcal{C}(\boldsymbol{A}_0) \cap \mathcal{C}(\boldsymbol{Y})$. The columns of the updated matrix $\boldsymbol{A}_1$ are denoted by the filled squares.

Note that all the columns of $\boldsymbol{A}_1$ are located inside $\mathcal{C}(\boldsymbol{Y})$ only for the regularized updates, i.e., in Figs. 2 and 4. For these cases, we set $\lambda_0 = 100$, satisfying the condition $\lambda_0 > \sigma_{\max}(\boldsymbol{X}_1)$, where $\sigma_{\max}(\boldsymbol{X}_1) \approx 41$. Figure 4 refers to the noisy case with $\mathrm{SNR} = 30$ [dB]. When $\lambda_0 = 0$ (see Fig. 3), the columns of $\boldsymbol{A}_1$ are outside $\mathcal{C}(\boldsymbol{Y})$ and quite far from $\mathcal{C}(\boldsymbol{A}_{\rm true})$ (true solution). In the second iteration, the points of $\boldsymbol{Y}^{(Z_2)}$ are marked with the unfilled circles which all belong to $\mathcal{C}(\boldsymbol{A}_1)$. Figure 3 illustrates that a majority of the vectors $\boldsymbol{y}_t$ belongs to $\mathcal{C}(\boldsymbol{A}_1)$, which suggests a very fast convergence—unfortunately, to a wrong local minimum. The columns of $\boldsymbol{A}_k$ in consecutive iterations are plotted with the unfilled squares. Note that the updates $\boldsymbol{A}_k$ are convergent to $\boldsymbol{A}_{\rm true}$ only in Fig. 2. For the noisy case (see Fig. 4), the updates $\boldsymbol{A}_k$ for $k > 1$ are located in the vicinity of $\boldsymbol{A}_{\rm true}$, although the extreme rays of $\mathcal{C}(\boldsymbol{Y})$ are very far from the extreme rays of $\mathcal{C}(\boldsymbol{A}_{\rm true})$.

The estimated mixing matrices are also evaluated in terms of the averaged Signal-to-Interference Ratio (SIR). This measure is defined by Cichocki *et al.* (2009, Chapter 3). For all the scenarios presented in Figs. 2–4, we obtained $\mathrm{SIR} = 77.1,\ 13.1,\ 26.1$ [dB] after 50 alternating steps, respectively.

The normalized residual error $||\boldsymbol{Y} - \boldsymbol{A}\boldsymbol{X}||_F / ||\boldsymbol{Y}||_F$ is plotted versus iterations in Fig. 5. Note that the fastest convergence and the lowest value of the residual error is obtained with the unregularized updates, i.e., for $\lambda_0 = 0$. Hence, one should be aware that this measure is inappropriate for estimating the quality of estimation when the underlying problem is not unique. Probably, due to the normalization of the updated factors, the local minimum for the estimates shown in Fig. 3 is deeper than for the estimates in Fig. 2. This suggests that any NMF algorithm that assures too fast convergence in early alternating steps may encounter similar problems.

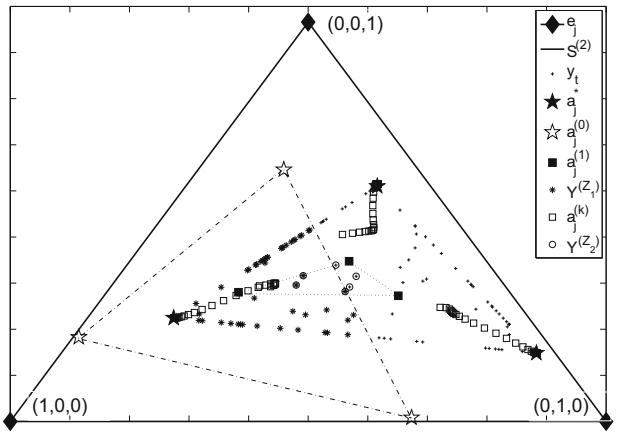

Fig. 2. Geometry of updates projected onto $\mathcal{S}^{(2)}$ for $\lambda_0 = 100$ and noise-free data. After 50 alternating steps, the mixing matrix is estimated with $\mathrm{SIR} = 77.1$ [dB].

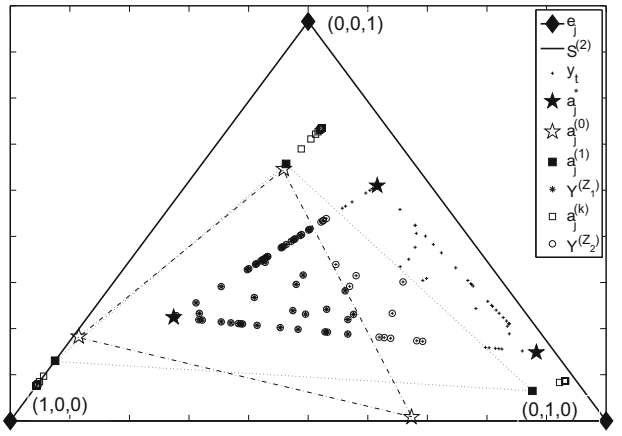

Fig. 3. Geometry of updates projected onto $\mathcal{S}^{(2)}$ for $\lambda_0 = 0$ and noise-free data. After 50 alternating steps, the mixing matrix is estimated with $\mathrm{SIR} = 13.1$ [dB].

5.2. Test B. In this test, we compare several algorithms using some mixtures of the spectral signals presented in Fig. 1. We analyze three scenarios for observations: (a) $I = J = 3$ (the first three signals from Fig. 1), (b) $I = J = 5$, (c) $I = 10$, $J = 5$. For each scenario, the noise-free and noisy mixtures with $\mathrm{SNR} = 30$ [dB] are obtained using the mixing matrix $\boldsymbol{A} \in \mathbb{R}_+^{I \times J}$ randomly generated from a uniform distribution.

We compare the following algorithms: a few versions of RASLS, FC-NNLS (Benthem and Keenan, 2004), projected ALS (Berry *et al.*, 2007; Cichocki *et al.*, 2009), Lin's Projected Gradient (LPG) (Lin, 2007), standard Lee–Seung NMF for the Euclidean distance (referred to as MUE) (Lee and Seung, 1999), Minimum Volume Constrained NMF (MVC-NMF) (Miao and Qi, 2007), Vertex Component Analysis (VCA) (Nascimento and

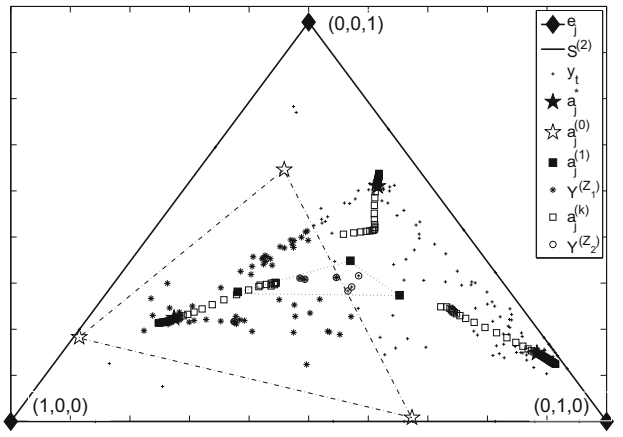

Fig. 4. Geometry of updates projected onto $\mathcal{S}^{(2)}$ for $\lambda_0 = 100$ and noisy data with $\mathrm{SNR} = 30$ [dB]. After 50 alternating steps, the mixing matrix is estimated with $\mathrm{SIR} = 26.1$ [dB].

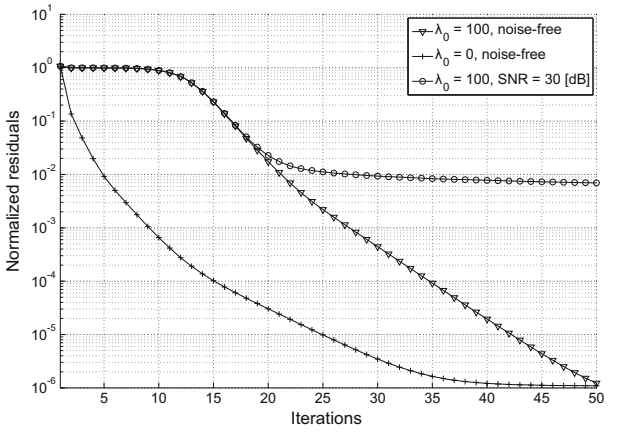

Fig. 5. Normalized residual error: $||\boldsymbol{Y} - \boldsymbol{AX}||_F / ||\boldsymbol{Y}||_F$ versus iterations for Test A.

Bioucas-Dias, 2005), Minimum Volume Simplex Analysis (MVSA) (Li and Bioucas-Dias, 2008), Simplex Identification via Split Augmented Lagrangian (SISAL) (Bioucas-Dias, 2009), and two versions of the SUNSAL algorithm (SUNSAL and SUNSAL-TV) (Bioucas-Dias and Figueiredo, 2010; Iordache *et al.*, 2012) combined with the FC-NNLS algorithm. The SUNSAL-based algorithm is used for updating the matrix $\boldsymbol{X}$, and the matrix $\boldsymbol{A}$ is updated with the FC-NNLS algorithm. The minimum number of augmented Lagrangian iterations in both SUNSAL-based algorithms is experimentally set to 50. The regularization parameter λ for the l_1-norm penalty term amounts to 0.01, and for the TV-based term is set to 0.001. For the LPG algorithm[3] we set $tol = 10^{-8}$ to avoid early termination of iterations.

[3] `http://www.csie.ntu.edu.tw/~cjlin/`.

Table 1. Notation for RASLS algorithms.

Algorithm	Description
RASLS-L1(d)	In the problem (14) the parameter β is decreasing according to the rule $\beta = \max\{\bar{\beta}, 2^{-k}\beta_0\}$, where $\beta_0 = 1000$ and $\bar{\beta} = 10^{-12}$. The parameter λ in the problem (15) is set to a constant value 10^{-12}.
RASLS-L2(d)	$\beta = 0$ and λ is decreased, starting from $\lambda_0 = 1000$ (Algorithm 4).
RASLS-L1(s)	$\beta = 10^4$ is fixed and $\lambda = 10^{-12}$ is fixed.
RASLS-L2(s)	$\beta = 0$ and $\lambda = 10^{-4}$ is fixed.
RASLS-L1-L2(d)	$\beta = 10^4$ is fixed and λ is decreased, starting from $\lambda_0 = 1000$.

The notation for a family of RASLS algorithms is given in Table 1. The parameters β and λ in RASLS-L1(s) and RASLS-L2(s) are set optimally according to the results obtained for the scenario (c), and presented in Fig. 6. It illustrates the SIR statistics (mean and std.) for RASLS-L1(s) and the RASLS-L2(s) versus the penalty parameter. Note that RASLS-L2(s) gives good results only for a very narrow range of the parameter λ.

All the NMF-based algorithms, except for MVC-NMF, are terminated after 50 alternating steps. The maximum number of alternating steps for MVC-NMF is set to 150 (as default).

Each NMF-based algorithm (except for MVC-NMF) is run for 100 Monte Carlo (MC) trials with a random initialization. The statistics of SIR samples for benchmark A and the scenario (c) are shown in Fig. 7(a) for noise-free mixtures and in Fig. 7(b) for noisy mixtures with $SNR = 30$ [dB]. The SIR results for the scenarios (a) and (b) are presented in Table 2, together with the runtime. The VCA algorithm is not randomly initialized, and the MVSA and SISAL algorithms are initialized with the VCA output. Hence, the MC analysis is not carried out for the VCA, MVSA and SISAL algorithms.

The algorithms are coded in Matlab 2008a, and executed on a computer with CPU X9650 (4 cores), 3 GHz, 64 bit, 8 GB RAM. The runtime considers evaluation of the stopping criteria for RASLS and LPG. In ALS and MUE, the matrices $\boldsymbol{A}$ and $\boldsymbol{X}$ are updated only once in each alternating step.

Figure 7 shows that the RASLS-L2(d) algorithm applied to the noise-free data gives the best mean-SIR results. For the noisy data, RASLS-L1-L2(d) seems to be the best choice, especially since it gives the most stable results (with the lowest variation in the SIR performance). Some outliers beyond the boxplots in Fig. 7 for RASLS-L2(d) are probably caused by a wrong initialization when the condition $\mathcal{C}(\boldsymbol{A}_0) \cap \mathcal{C}(\boldsymbol{Y}) \neq \emptyset$ is not satisfied.

The results presented in Table 2 also confirm that a

Table 2. Mean-SIR [dB], STD [dB], and elapsed time [seconds] obtained in Test B for the scenario (a) ($I = J = 3$) and scenario (b) ($I = J = 5$). The algorithms in the rows 1–11 (RASLS-L1(d)—NNLS-SUNSAL-TV) are initialized randomly in each MC run. The mean-SIR and STD (in parenthesis) are calculated for 100 MC runs.

Algorithm	$I = J = 3$, $\mathrm{cond}(\boldsymbol{A}) = 2.65$			$I = J = 5$, $\mathrm{cond}(\boldsymbol{A}) = 18.66$		
	SIR (STD) noise-free	SIR (STD) SNR = 30 [dB]	Time sec.	SIR (STD) noise-free	SIR (STD) SNR = 30 [dB]	Time sec.
RASLS-L1(d)	24.19 (8.61)	21.99 (5.56)	0.67	**21.53 (4.99)**	**19.15 (2.9)**	1.48
RASLS-L2(d)	31.13 (2.18)	**28.86 (0)**	0.48	15.12 (3.15)	11.29 (1.77)	2.43
RASLS-L1(c)	13.29 (2.19)	12.69 (1.81)	0.82	14.91 (1.45)	11.27 (2.33)	2.13
RASLS-L2(c)	28.71 (2.08)	26.09 (1.31)	0.44	14.01 (3.04)	14.06 (2.87)	2.16
RASLS-L1-L2(d)	13.18 (1.87)	13.16 (1.85)	0.85	16.92 (0)	16.91 (0)	0.88
FC-NNLS	24.95 (6.93)	19.87 (4.59)	0.35	9.88 (3.81)	6.59 (2.12)	1.34
ALS	21.07 (7.01)	19.11 (4.32)	0.12	11.54 (6.04)	6.25 (2.54)	0.2
LPG	17.15 (2.17)	16.64 (2.01)	2.51	7.23 (1.57)	6.38 (1.31)	2.96
MUE	13.53 (4.96)	12.95 (4.72)	0.12	7.63 (1.82)	7.21 (1.67)	0.19
NNLS-SUNSAL	11.87 (2.84)	11.4 (2.65)	3.54	13.23 (0)	12.29 (1.87)	6.77
NNLS-SUNSAL-TV	24.48 (6.03)	23.59 (5.94)	51.24	8.84 (3.07)	8.25 (2.91)	105.85
MVC-NMF	**36. 59**	13.61	14.89	-	-	-
VCA	20.46	14.83	0.16	12.95	6.42	0.16
MVSA	17.23	16.86	0.55	15.06	8.42	0.77
SISAL	17.76	16.98	0.24	14.52	13.73	0.37

gradual decrease in the regularization parameters is very important to obtain good results. RASLS-based algorithms are outperformed by the MVC-NMF algorithm but only for the noise-free data with $I = J = 3$. Unfortunately, MVC-NMF works very unstable for $J > 3$. Surprisingly, the SIR result obtained with RASLS-L2(d) is very good even for the noisy data with $I = J = 3$, which is very difficult for most NMF algorithms. Obviously, if the number of mixtures is the same as that of sources, the mixing matrix must be well-conditioned. In our case, $\mathrm{cond}(\boldsymbol{A}_{\mathrm{true}}) = 2.65$ for $I = J = 3$, and $\mathrm{cond}(\boldsymbol{A}_{\mathrm{true}}) = 18.66$ for $I = J = 5$.

The good SIR performance of RASLS-based algorithms does not considerably increase a computational complexity. According to Table 2, the computational time for these algorithms is from about three to twelve times longer than for ALS and MUE, and slightly shorter than for LPG. Interestingly, when the l_1-norm constraint is imposed, the runtime for the RASLS algorithm decreases. This is probably caused by an increase in the sparsity and consequently in the number of active entries.

5.3. Test C. Test C is carried out for benchmark B. The original 50 spiky signals are mixed using a random positive matrix $\boldsymbol{A} \in \mathbb{R}_{+}^{150\times 50}$. The MC simulations are carried out for the following algorithms: RASLS-L1(d), RASLS-L2(d), FC-NNLS, projected ALS, LPG, MUE, SUNSAL combined with FC-NNLS, and SUNSAL-TV combined with FC-NNLS. Each algorithm runs for 50 alternating steps. In RASLS-L1(d) and RASLS-L2(d), we set $\beta_0 = 10^3$ and $\lambda_0 = 10^4$, respectively.

The SIR results are shown in Fig. 8(a) for noise-free mixtures, and in Fig. 8(b) for noisy mixtures with $\mathrm{SNR} = 30$ [dB]. For VCA, MVSA and SISAL, the SIR performance is 8, 3.37, 12.7 [dB] for the noise-free data, and 6.58, 2.9, and 11.57 [dB] for the noisy data, respectively. The normalized residual error for the noisy case is shown in Fig. 9.

The averaged elapsed time for the RASLS-L1(d), RASLS-L2(d), FC-NNLS, ALS, LPG, MUE, NNLS-SUNSAL, and NNLS-SUNSAL-TV algorithms running for 50 alternating steps is 28.58, 92.84, 102.28, 1.66, 13.67, 1.56, 30.19, 650.36 seconds, respectively. We noticed that the elapsed time of RASLS-L2(d) increases exponentially with the number of sources.

This test demonstrates that RASLS-L2(d) gives the best SIR performance for benchmark B, both for noise-free and noisy data. The noise only slightly deteriorates the performance. The residual error informs us that, starting from $\lambda_0 = 10^4$, RASLS-L2(d) updates run for about the first 12 alternating steps in a stagnation point, and then go monotonically to another stagnation point after about 30 iterations for the noisy data. The latter stagnation reaches the lowest value of the residual error from all the tested algorithms. For the noise-free data, the behavior of the residual error is similar for the RASLS algorithms.

5.4. Test D. Test D is carried out for spectral unmixing of hyperspectral data. The linear mixtures are created using the spectral signals of the endmembers from benchmark C. The abundance maps are generated according to Miao and Qi (2007), using a 7×7 low pass filter. The pure pixels are removed. The noisy mixtures are obtained in a similar way as in the previous experiments.

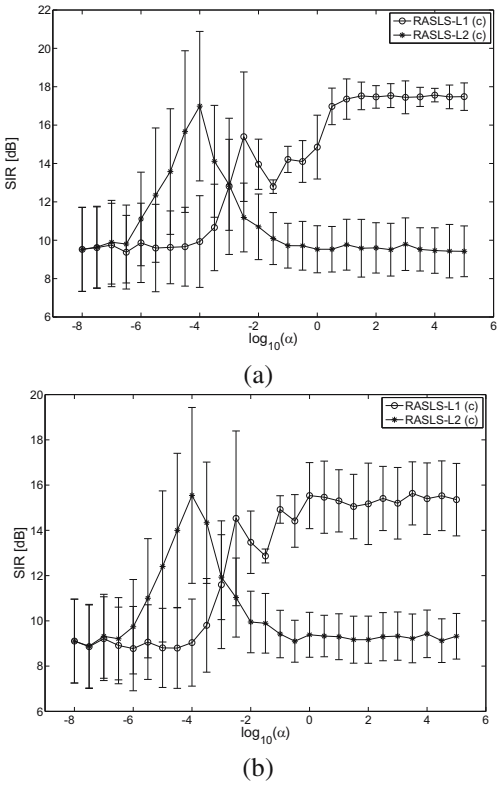

Fig. 6. SIR statistics for the estimation of the mixing matrix $\boldsymbol{A}$ in Test B using RASLS-L1(s) and RASLS-L2(s) versus fixed values of the penalty parameter α: noise-free data (a), noisy data with $\mathrm{SNR} = 30$ [dB] (b). For RASLS-L1(s): $\alpha = \beta$ and $\lambda = 10^{-12}$. For RASLS-L2(s): $\alpha = \lambda$ and $\beta = 0$.

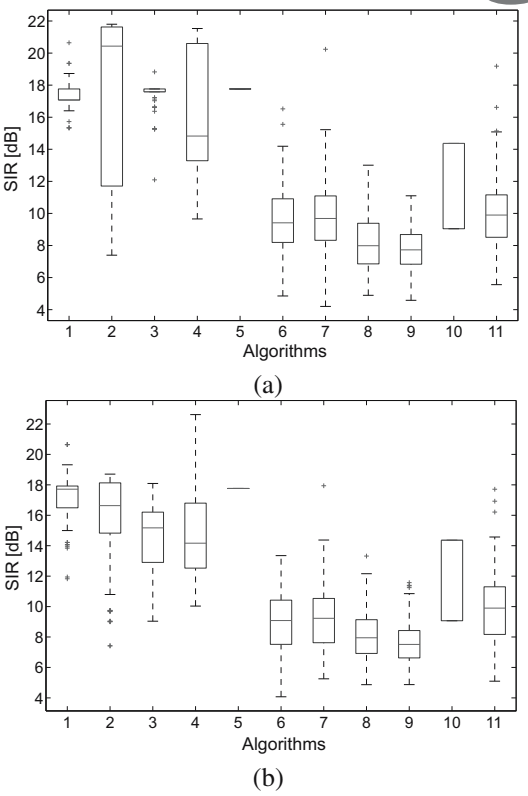

Fig. 7. SIR statistics for the estimation of the mixing matrix $\boldsymbol{A}$ using various NMF algorithms (1: RASLS-L1(d), 2: RASLS-L2(d), 3: RASLS-L1(s), 4: RASLS-L2(s), 5: RASLS-L1-L2(d), 6: FC-NNLS, 7: ALS, 8: LPG, 9: MUE, 10: NNLS-SUNSAL, 11: NNLS-SUNSAL-TV) in Test B ($I = 10$, $J = 5$): noise-free data (a), noisy data with $\mathrm{SNR} = 30$ [dB] (b).

Figure 10 illustrates the SIR statistics obtained with 100 MC runs of the following algorithms: RASLS-L1(d), RASLS-L2(d), FC-NNLS, ALS, LPG, MUE, and NNLS-SUNSAL. NNLS-SUNSAL-TV is omitted in this experiment since it is extremely slow. For VCA, MVSA and SISAL, the SIR performance is 23.45, **273**, 92.3 [dB] for noise-free data, and 12.91, 7.55, and 7.69 [dB] for the noisy data, respectively.

In this test, the best SIR performance is obtained with the MVSA and SISAL algorithms, but only for the noise-free data. The noise considerably deteriorates this result, and the best SIR performance for the noisy mixtures is obtained with the NNLS-SUNSAL algorithm. RASLS algorithms give somehow worse performance than NNLS-SUNSAL, but still considerably better than the other NMF algorithms.

6. Conclusions

We demonstrated by a geometrical analysis and numerical experiments that RASLS NMF algorithms may be very useful for blind spectral signal unmixing. When the NMF problem is not unique but one of the estimated factors is sufficiently sparse, the regularization parameter should decrease gradually with alternating steps, starting from a large initial value that can be set up in a large range (e.g., 10^2–10^{15}) using the lower bound in (19). Obviously, a larger value of λ_0 needs more alternating steps to run, but the risk of getting stuck into unfavorable local minima is lower. We used the rule $\lambda \longleftarrow \lambda/2$, although, other rules are also possible. For example, the exponential rule $\lambda = \lambda_0 \exp\{-\tau k\}$ works even better, but it needs two parameters to be set up in advance. Moreover, the results can be more stable if some additional l_1-norm constraints are added to impose sparsity on one of the estimated factors. This approach can be found in the RASLS-L1-L2(d) algorithm.

The computational complexity of the proposed algorithm strongly depends on the number of endmembers. When this number is large, the algorithm can be very slow. Hence, further research is needed to tackle this problem.

Summing up, the proposed approach is very simple and may be useful for improving the performance of NMF

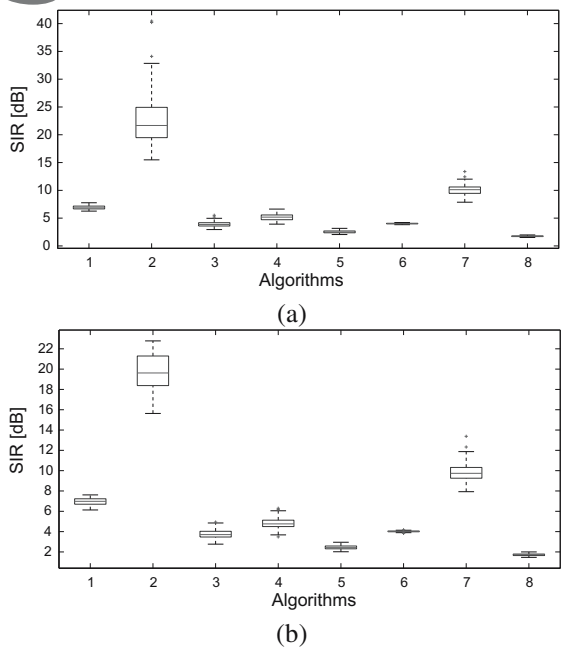

Fig. 8. SIR statistics for the estimation of the mixing matrix $\boldsymbol{A}$ using various NMF algorithms (1: RASLS-L1(d), 2: RASLS-L2(d), 3: FC-NNLS, 4: ALS, 5: LPG, 6: MUE, 7: NNLS-SUNSAL, 8: NNLS-SUNSAL-TV) in Test C ($I = 150$, $J = 50$): noise-free data (a), noisy data with $\mathrm{SNR} = 30$ [dB] (b).

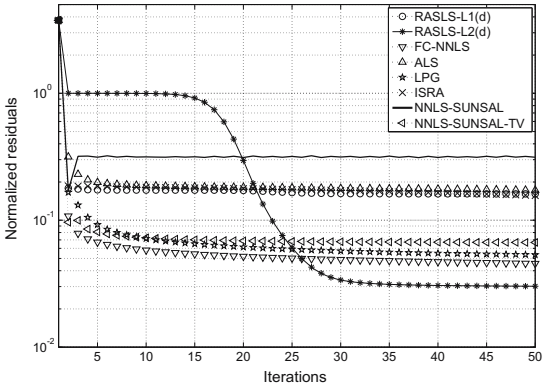

Fig. 9. Normalized residual error: $||\boldsymbol{Y} - \boldsymbol{AX}||_F / ||\boldsymbol{Y}||_F$ versus iterations for Test C.

algorithms, especially for factorizable and nonunique NMF problems. Apart from the data spectral analysis, it might find numerous applications in machine learning and artificial intelligence, including supervised classification (e.g., Woźniak and Krawczyk, 2012; Górecki and Łuczak, 2013), clustering (e.g., Kulczycki and Charytanowicz, 2010), and image processing (e.g., Cichocki *et al.*, 2009; Hansen, 1998).

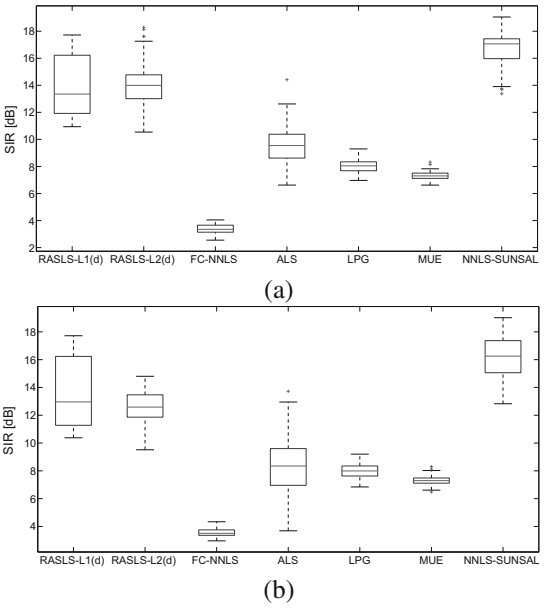

Fig. 10. SIR statistics for the estimation of the mixing matrix $\boldsymbol{A}$ using various NMF algorithms in Test D: noise-free data (a), noisy data with $\mathrm{SNR} = 30$ [dB] (b).

Acknowledgment

The author would like to thank Prof. Jose M. Bioucas-Dias and Dr. Marian-Daniel Iordache for providing the codes for the SUNSAL, SUNSAL-TV and SISAL algorithms, and Dr. Hairong Qi for providing the code for MVC-NMF.

References

Benthem, M.H.V. and Keenan, M.R. (2004). Fast algorithm for the solution of large-scale non-negativity-constrained least squares problems, *Journal of Chemometrics* **18**(10): 441–450.

Berry, M., Browne, M., Langville, A.N., Pauca, P. and Plemmons, R.J. (2007). Algorithms and applications for approximate nonnegative matrix factorization, *Computational Statistics and Data Analysis* **52**(1): 155–173.

Bioucas-Dias, J.M. (2009). A variable splitting augmented Lagrangian approach to linear spectral unmixing, *Proceedings of the 1st IEEE GRSS Workshop on Hyperspectral Image and Signal Processing, WHISPERS, Grenoble, France*.

Bioucas-Dias, J.M. and Figueiredo, M. (2010). Alternating direction algorithms for constrained sparse regression: Application to hyperspectral unmixing, *Proceedings of the 2nd IEEE GRSS Workshop on Hyperspectral Image and Signal Processing, WHISPERS, Raykjavik, Iceland*.

Bioucas-Dias, J.M., Plaza, A., Dobigeon, N., Parente, M., Du, Q., Gader, P. and Chanussot, J. (2012). Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches, *IEEE Journal of Selected*

Topics in Applied Earth Observations and Remote Sensing **5**(2): 354–379.

Bro, R. and Jong, S.D. (1997). A fast non-negativity-constrained least squares algorithm, *Journal of Chemometrics* **11**(5): 393–401.

Calvetti, D., Lewis, B. and Reichel, L. (2001). On the choice of subspace for iterative methods for linear discrete ill-posed problems, *International Journal of Applied Mathematics and Computer Science* **11**(5): 1069–1092.

Chan, T.-H., Ma, W.-K., Ambikapathi, A.-M. and Chi, C.-Y. (2011). A simplex volume maximization framework for hyperspectral endmember extraction, *IEEE Transactions on Geoscience and Remote Sensing* **49**(11): 4177–4193.

Chen, D. and Plemmons, R.J. (2009). Nonnegativity constraints in numerical analysis, *in* A. Bultheel and R. Cools (Eds.), *The Birth of Numerical Analysis*, World Scientific, Singapore, pp. 109–139.

Chu, M.T. and Lin, M. M. (2008). Low dimensional polytype approximation and its applications to nonnegative matrix factorization, *SIAM Journal of Scientific Computing* **30**(3): 1131–1151.

Cichocki, A., Zdunek, R., Phan, A.H. and Amari, S.-I. (2009). *Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-way Data Analysis and Blind Source Separation*, Wiley and Sons, Chichester.

Dabrowski, A. and Cetnarowicz, D. (2008). Iterative SVD algorithm as a BSS solution, *Proceedings of the International Conference on Signals and Electronic Systems, ICSES 2008, Cracow, Poland*, pp. 401–404.

Donoho, D. and Stodden, V. (2004). When does non-negative matrix factorization give a correct decomposition into parts?, *in* S. Thrun, L. Saul and B. Schölkopf (Eds.), *Advances in Neural Information Processing Systems (NIPS)*, Vol. 16, MIT Press, Cambridge, MA, pp. 1141–1148.

Elden, L. (1977). Algorithms for the regularization of ill-conditioned least squares problems, *BIT* **17**(2): 134–145.

Garda, B. and Galias, Z. (2012). Non-negative least squares and the Tikhonov regularization methods for coil design problems, *Proceedings of the International Conference on Signals and Electronic Systems, ICSES'12, Wrocław, Poland*.

Gobinet, C., Perrin, E. and Huez, R. (2004). Application of non-negative matrix factorization to fluorescence spectroscopy, *Proceedings of the European Signal Processing Conference, EUSIPCO 2004, Vienna, Austria*, pp. 1095–1098.

Górecki, T. and Łuczak, M. (2013). Linear discriminant analysis with a generalization of the Moore–Penrose pseudoinverse, *International Journal of Applied Mathematics and Computer Science* **23**(2): 463–471, DOI: 10.2478/amcs-2013-0035.

Guo, Z., Wittman, T. and Osher, S. (2009). L1 unmixing and its application to hyperspectral image enhancement, *in* S.S. Shen and P.E. Lewis (Eds.), *Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XV*, Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, Vol. 7334, International Society for Optical Engineering, Orlando, FL, p. 73341M+.

Hamza, A. and Brady, D. (2006). Reconstruction of reflectance spectra using robust nonnegative matrix factorization, *IEEE Transactions on Signal Processing* **54**(9): 3637–3642.

Hancewicz, T.M. and Wang, J.-H. (2005). Discriminant image resolution: A novel multivariate image analysis method utilizing a spatial classification constraint in addition to bilinear nonnegativity, *Chemometrics and Intelligent Laboratory Systems* **77**(1–2): 18–31.

Hansen, P.C. (1998). *Rank-Deficient and Discrete Ill-Posed Problems*, SIAM, Philadelphia, PA.

Heylen, R., Burazerovic, D. and Scheunders, P. (2011). Fully constrained least squares spectral unmixing by simplex projection, *IEEE Transactions on Geoscience and Remote Sensing* **49**(11): 4112–4122.

Huck, A., Guillaume, M. and Blanc-Talon, J. (2010). Minimum dispersion constrained nonnegative matrix factorization to unmix hyperspectral data, *IEEE Transactions on Geoscience and Remote Sensing* **48**(6): 2590–2602.

Hyvrinen, A., Karhunen, J. and Oja, E. (2001). *Independent Component Analysis*, John Wiley, New York, NY.

Igual, J. and Llinares, R. (2008). Nonnegative matrix factorization of laboratory astrophysical ice mixtures, *IEEE Journal of Selected Topics in Signal Processing* **2**(5): 697–706.

Igual, J., Llinares, R. and Salazar, A. (2006). Source separation of astrophysical ice mixtures, *Proceedings of the 6th International Conference on Independent Component Analysis and Blind Signal Separation, Charleston, IL, USA*, pp. 368–375.

Iordache, M., Dias, J. and Plaza, A. (2011). Sparse unmixing of hyperspectral data, *IEEE Transactions on on Geoscience and Remote Sensing* **49**(2): 2014–2039.

Iordache, M., Dias, J. and Plaza, A. (2012). Total variation spatial regularization for sparse hyperspectral unmixing, *IEEE Transactions on Geoscience and Remote Sensing* **50**(11): 4484–4502.

Jia, S. and Qian, Y. (2009). Constrained nonnegative matrix factorization for hyperspectral unmixing, *IEEE Transactions on Geoscience and Remote Sensing* **47**(1): 161–173.

Kim, D., Sra, S. and Dhillon, I.S. (2007). Fast Newton-type methods for the least squares nonnegative matrix approximation problem, *Proceedings of the 6th SIAM International Conference on Data Mining, Minneapolis, MN, USA*, pp. 343–354.

Kim, H. and Park, H. (2008). Non-negative matrix factorization based on alternating non-negativity constrained least squares and active set method, *SIAM Journal on Matrix Analysis and Applications* **30**(2): 713–730.

Kim, J. and Park, H. (2011). Fast nonnegative matrix factorization: An active-set-like method and comparisons, *SIAM Journal on Scientific Computing* **33**(6): 3261–3281.

Krawczyk-Stańdo, D. and Rudnicki, M. (2007). Regularization parameter selection in discrete ill-posed problems—The use of the U-curve, *International Journal of Applied Mathematics and Computer Science* **17**(2): 157–164, DOI: 10.2478/v10006-007-0014-3.

Kulczycki, P. and Charytanowicz, M. (2010). A complete gradient clustering algorithm formed with kernel estimators, *International Journal of Applied Mathematics and Computer Science* **20**(1): 123–134, DOI: 10.2478/v10006-010-0009-3.

Lawson, C.L. and Hanson, R.J. (1974). *Solving Least Squares Problems*, Prentice-Hall, Englewood Cliffs, NJ.

Lee, D.D. and Seung, H.S. (1999). Learning the parts of objects by non-negative matrix factorization, *Nature* **401**(6755): 788–791.

Li, H., Adali, T., Wang, W., Emge, D. and Cichocki, A. (2007). Non-negative matrix factorization with orthogonality constraints and its application to Raman spectroscopy, *The Journal of VLSI Signal Processing* **48**(1–2): 83–97.

Li, J. and Bioucas-Dias, J.M. (2008). Minimum volume simplex analysis: A fast algorithm to unmix hyperspectral data, *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium, IGARSS 2008, Boston, MA, USA*, Vol. 3, pp. 250–253.

Likic, V.A. (2009). Extraction of pure components from overlapped signals in gas chromatography-mass spectrometry (GC-MS), *BioData Mining* **2**(6): 1–11.

Lin, C.-J. (2007). Projected gradient methods for non-negative matrix factorization, *Neural Computation* **19**(10): 2756–2779.

Llinares, R., Igual, J., Miró-Borrás, J. and Camacho, A. (2010). Analysis of astrophysical ice analogs using regularized alternating least squares, *Proceedings of the 20th International Conference on Artificial Neural Networks, ICANN 2010, Thessaloniki, Greece*, pp. 199–204.

Makowski, R. (2003). Source pulse estimation of mine shocks by blind deconvolution, *Pure and Applied Geophysics* **160**(7): 1191–1205.

Miao, L. and Qi, H. (2007). Endmember extraction from highly mixed data using minimum volume constrained nonnegative matrix factorization, *IEEE Transactions on Geoscience and Remote Sensing* **45**(3): 765–777.

Miron, S., Dossot, M., Carteret, C., Margueron, S. and Brie, D. (2011). Joint processing of the parallel and crossed polarized Raman spectra and uniqueness in blind nonnegative source separation, *Chemometrics and Intelligent Laboratory Systems* **105**(1): 7–18.

Nascimento, J.M.P. and Bioucas-Dias, J.M. (2005). Vertex component analysis: A fast algorithm to unmix hyperspectral data, *IEEE Transactions on Geoscience and Remote Sensing* **43**(4): 898–910.

Pauca, V.P., Pipera, J. and Plemmons, R.J. (2006). Nonnegative matrix factorization for spectral data analysis, *Linear Algebra and Its Applications* **416**(1): 29–47.

Pengo, T., Munoz-Barrutia, A. and de Solorzano, C.O. (2010). Spectral unmixing of multiply stained fluorescence samples, *in* A. Mendez-Vilas and J. Diaz (Eds.), *Microscopy: Science, Technology, Applications and Education*, Microscopy Book Series, No. 4, Formatex Research Center, Badajoz, pp. 2079–2087.

Plaza, J., Hendrix, E.M.T., García, I., Martin, G. and Plaza, A. (2012). On endmember identification in hyperspectral images without pure pixels: A comparison of algorithms, *Journal of Mathematical Imaging and Vision* **42**(2–3): 163–175.

Qian, Y., Jia, S., Zhou, J. and Robles-Kelly, A. (2011). Hyperspectral unmixing via $l_{1/2}$ sparsity-constrained nonnegative matrix factorization, *IEEE Transactions on Geoscience and Remote Sensing* **49**(11): 4282–4297.

Rojas, M. and Steihaug, T. (2002). An interior-point trust-region-based method for large-scale non-negative regularization, *Inverse Problems* **18**(5): 1291–1307.

Sajda, P., Du, S., Brown, T., Parra, L. and Stoyanova, R. (2003). Recovery of constituent spectra in 3D chemical shift imaging using nonnegative matrix factorization, *Proceedings of the 4th International Symposium on Independent Component Analysis and Blind Signal Separation, Nara, Japan*, pp. 71–76.

Sajda, P., Du, S., Brown, T.R., Stoyanova, R., Shungu, D.C., Mao, X. and Parra, L.C. (2004). Nonnegative matrix factorization for rapid recovery of constituent spectra in magnetic resonance chemical shift imaging of the brain, *IEEE Transactions on Medical Imaging* **23**(12): 1453–1465.

Siwek, K., Osowski, S. and Szupiluk, R. (2009). Ensemble neural network approach for accurate load forecasting in a power system, *International Journal of Applied Mathematics and Computer Science* **19**(2): 303–315, DOI: 10.2478/v10006-009-0026-2.

Tong, L., van der Veen, A.-J., Dewilde, P. and Sung, Y. (2003). Blind decorrelating RAKE receivers for long-code WCDMA, *IEEE Transactions on Signal Processing* **51**(6): 1642–1655.

Woźniak, M. and Krawczyk, B. (2012). Combined classifier based on feature space partitioning, *International Journal of Applied Mathematics and Computer Science* **22**(4): 855–866, DOI: 10.2478/v10006-012-0063-0.

Zdunek, R. (2011). Regularized active set least squares algorithm for nonnegative matrix factorization in application to Raman spectra separation, *in* J. Cabestany, I. Rojas and G. Joya (Eds.), *Advances in Computational Intelligence*, Lecture Notes in Computer Science, Vol. 6692, Springer, Berlin/Heidelberg, pp. 492–499.

Zdunek, R. (2012). Hyperspectral image unmixing with nonnegative matrix factorization, *Proceedings of the IEEE International Conference on Signals and Electronic Systems, ICSES 2012, Wrocław, Poland*.

Zdunek, R. and Cichocki, A. (2007). Nonnegative matrix factorization with constrained second-order optimization, *Signal Processing* **87**(8): 1904–1916.

Zhang, J., Rivard, B. and Rogge, D.M. (2008). The successive projection algorithm (SPA), an algorithm with a spatial constraint for the automatic search of endmembers in hyperspectral data, *Sensors* **8**(2): 1321–1342.

Zymnis, A., Kim, S.-J., Skaf, J., Parente, M. and Boyd, S. (2007). Hyperspectral image unmixing via alternating projected subgradients, *Proceedings of the 41st Asilomar Con-*

ference on Signals, Systems and Computers, ACSSC 2007, Pacific Grove, CA, USA, pp. 1164–1168.

Rafał Zdunek received the M.Sc. and Ph.D. degrees in telecommunications from the Wrocław University of Technology, Poland, in 1997 and 2002, respectively. Since 2002, he has been a lecturer in the Department of Electronics of the same university. In 2004, he was a visiting associate professor in the Institute of Statistical Mathematics, Tokyo, Japan. In 2005–2007, he worked as a research scientist in the Brain Science Institute, RIKEN, Saitama, Japan. His current research interests include numerical methods in application to nonnegative matrix and tensor factorisation. He has published over 80 technical papers in journals and refereed conference proceedings, including the co-authored monograph *Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-way Data Analysis and Blind Source Separation* (Wiley, 2009).

Received: 9 January 2013
Revised: 7 December 2013
Re-revised: 8 January 2014