Journal of VLSI Signal Processing 37, 263–279, 2004

Bayesian Unsupervised Learning for Source Separation with Mixture of Gaussians Prior

HICHEM SNOUSSI AND ALI MOHAMMAD-DJAFARI
Laboratoire des Signaux et Systèmes (CNRS, SUPÉLEC, UPS), SUPÉLEC, Plateau de Moulon,
91192 Gif-sur-Yvette Cedex, France

Abstract. This paper considers the problem of source separation in the case of noisy instantaneous mixtures. In a previous work [1], sources have been modeled by a mixture of Gaussians leading to an hierarchical Bayesian model by considering the labels of the mixture as i.i.d hidden variables. We extend this modelization to incorporate a Markovian structure for the labels. This extension is important for practical applications which are abundant: unsu- pervised classification and segmentation, pattern recognition and speech signal processing.

In order to estimate the mixing matrix and the *a priori* model parameters, we consider observations as incomplete data. The missing data are sources and labels: sources are missing data for observations and labels are missing data for incomplete missing sources. This hierarchical modelization leads to specific restoration maximization type algorithms. Restoration step can be held in three different manners: (i) Complete likelihood is estimated by its conditional expectation. This leads to the EM (expectation-maximization) algorithm [2], (ii) Missing data are estimated by their maximum *a posteriori*. This leads to JMAP (Joint maximum *a posteriori*) algorithm [3], (iii) Missing data are sampled from their *a posteriori* distributions. This leads to the SEM (stochastic EM) algorithm [4]. A Gibbs sampling scheme is implemented to generate missing data. We have also introduced a relaxation strategy into these algorithms to reduce the computational cost which is due to the exponential influence of the number of source components and the number of the mixture Gaussian components.

Keywords: source separation, HMM models, EM algorithm, Gibbs sampling

Introduction

We consider the problem of source separation in the noisy linear instantaneous case:

$$\boldsymbol{x}(t) = \boldsymbol{A}\boldsymbol{s}(t) + \boldsymbol{\epsilon}(t), t = 1..T \tag{1}$$

$\boldsymbol{x}(t)$ is the m-vector of observations, $\boldsymbol{s}(t)$ the n-vector of sources, $\epsilon(t)$ an additive Gaussian white noise with covariance $\boldsymbol{R}_\epsilon$ and $\boldsymbol{A}$ the $m \times n$ mixing matrix. Source separation problem consists of two sub-problems: Sources restoration and mixing matrix identification. Therefore, three directions can be followed:

1. *Supervised learning*: Identify $\boldsymbol{A}$ knowing a training sequence of sources s, then use it to reconstruct the sources.
2. *Unsupervised learning*: Identify $\boldsymbol{A}$ directly from a part or the whole observations and then use it to recover $\boldsymbol{s}$.
3. *Unsupervised joint estimation*: Estimate jointly $\boldsymbol{s}$ and $\boldsymbol{A}$.

Many techniques were proposed to solve the source separation problem based on entropy and information theoretic approach [5–9] and the maximum likelihood principle [10–16] leading to contrast functions [17–20] and estimating functions [21–24]. Among the limitations of these methods, we can mention: (i) the lack of possibility to account for some prior information about the mixing coefficients or other parameter involved in the problem, (ii) the lack of information about the degree of uncertainty of the mixing matrix estimate particularly in the noisy mixture, (iii) the objective functions are intractable or difficult to optimize when the source model is more elaborate... Recently, a few works using the Bayesian approach have been presented to push further the limits of these methods [6, 25–31]. For example, in the Bayesian framework, we can introduce some *a priori* information on the sources and on the mixing elements as well as on the hyperparameters by assigning appropriate prior laws for them. Also, thanks to the posterior laws, we can quantify the uncertainty of any estimated parameter. Finally, thanks to sampling schemes, we can propose tractable estimation algorithms.

In this paper, we introduce a double stochastic model for sources which has at least two advantages: (i) first, it is a parametric model so that the update of its parameters in the separating algorithm is an easy task, moreover, it is based on hidden variables so the estimation of its parameters has the same nature as the source separation problem, (ii) second, it is a good alternative to non parametric modeling since it is able to approach any probability distribution when increasing the number of components.

The paper is organized as follows: We begin by proposing a Bayesian approach to source separation. We set up the notations, present the prior laws for sources, mixing coefficients and hyperparameters involved in the parametric distributions. The sources are modeled by a double stochastic process by the introduction of hidden variables representing the labels of the mixture of Gaussians. The case of independent labels has been considered in previous works [1, 32, 33]. In this paper, we consider a Markovian structure of the labels. The mixing coefficients are supposed to have Gaussian distributions. It is known that the estimation of the variances by maximum likelihood is a degenerate problem (likelihood function goes to infinity when the variances approach zero) and the retained solution in [34] is to constrain the variances to belong to a strictly positive interval but this leads to a sophisticated constrained optimization. Recently, a Bayesian approach was proposed to eliminate degeneracy when directly observing the sources [35]. It consists in the penalization of the likelihood by an Inverted Gamma prior. In a previous work, we have shown that this degeneracy still occurs in the source separation problem and that an Inverted Gamma prior eliminates this degeneracy [36].

The incomplete data structure of the problem suggests the use of restoration maximization algorithms. Recently, in [32, 33, 37] the EM algorithm has been used in source separation with mixture of Gaussians as sources prior. In this work, we show that:

1. This algorithm fails in estimating jointly the variances of Gaussian mixture and noise covariance matrix. We proved that this is due to the degeneracy of the estimated variance to zero.
2. The computational cost of this algorithm is very high.
3. The algorithm is very sensitive to initial conditions.
4. In [32], there is neither an *a priori* distribution on the mixing matrix $\boldsymbol{A}$ nor on the hyperparameters $\boldsymbol{\eta}$.

Here, we propose to extend this algorithm by:

1. Introducing an *a priori* distribution for the hyperparameters to eliminate the aforementioned degeneracy.
2. Introducing an *a priori* distribution for $\boldsymbol{A}$ to express our previous knowledge on the mixing matrix elements.
3. Giving a Markovian structure to the labels of the mixture.

In Section 2, first we present the basics of general restoration-maximization algorithms, then we give the exact EM algorithm and discuss its computational cost. Then, we present other restoration-maximization algorithms:

(i) Viterbi-EM algorithm and Gibbs-EM algorithm. The Viterbi and Gibbs modifications of the exact EM algorithm breaks the temporal structure of the hidden Markov chain and consequently reduce the computational cost;

(ii) A fast version of the Viterbi-EM and Gibbs-EM algorithms will be considered to reduce the computational cost exponentially growing with the number of sources and the number of Gaussians of each source component.

In Section 3, simulation results are presented to show the performances of the proposed algorithms.

1. Bayesian Approach to Source Separation

Given the observations $\boldsymbol{x}_{1..T}$, the joint *a posteriori* distribution of unknown variables $\boldsymbol{s}_{1..T}$ and $\boldsymbol{A}$ is:

$$p(\boldsymbol{A}, \boldsymbol{s}_{1..T}, \boldsymbol{\eta} \mid \boldsymbol{x}_{1..T}) \propto p(\boldsymbol{x}_{1..T} \mid \boldsymbol{A}, \boldsymbol{s}_{1..T}, \boldsymbol{\eta}_n) \times p(\boldsymbol{A} \mid \eta_a) p(\boldsymbol{s}_{1..T} \mid \boldsymbol{\eta}_s) p(\boldsymbol{\eta}) \tag{2}$$

where $p(\boldsymbol{A} \mid \boldsymbol{\eta}_a)$ and $p(\boldsymbol{s}_{1..T} \mid \boldsymbol{\eta}_s)$ are the prior distributions through which we model our *a priori* information about mixing matrix $\boldsymbol{A}$ and sources $\boldsymbol{s}$. $p(\boldsymbol{x}_{1..T} \mid \boldsymbol{A}, \boldsymbol{s}_{1..T}, \boldsymbol{\eta}_n)$ is the joint likelihood distribution. $\boldsymbol{\eta} = (\boldsymbol{\eta}_n, \boldsymbol{\eta}_a, \boldsymbol{\eta}_s)$ are the hyperparameters. From here, we have two directions for unsupervised learning and separation:

1. First, estimate jointly $\boldsymbol{s}_{1..T}$, $\boldsymbol{A}$ and $\boldsymbol{\eta}$:

$$(\hat{\boldsymbol{A}}, \hat{\boldsymbol{s}}_{1..T}, \hat{\boldsymbol{\eta}}) = \underset{(\boldsymbol{A}, \boldsymbol{s}_{1..T}, \boldsymbol{\eta})}{\operatorname{argmax}} \{J(\boldsymbol{A}, \boldsymbol{s}_{1..T}, \boldsymbol{\eta}) = \ln p(\boldsymbol{A}, \boldsymbol{s}_{1..T}, \boldsymbol{\eta} \mid \boldsymbol{x}_{1..T})\} \tag{3}$$

2. Second, integrate (2) with respect to $\boldsymbol{s}_{1..T}$ to obtain the marginal in $(\boldsymbol{A}, \boldsymbol{\eta})$ and estimate them by:

$$(\hat{\boldsymbol{A}}, \hat{\boldsymbol{\eta}}) = \underset{(\boldsymbol{A}, \boldsymbol{\eta})}{\operatorname{argmax}} \{J(\boldsymbol{A}, \boldsymbol{\eta}) = \ln p(\boldsymbol{A}, \boldsymbol{\eta} \mid \boldsymbol{x}_{1..T})\} \tag{4}$$

   Then estimate $\hat{\boldsymbol{s}}_{1..T}$ using the posterior $p(\boldsymbol{s}_{1..T} | \boldsymbol{x}_{1..T}, \hat{\boldsymbol{A}}, \hat{\boldsymbol{\eta}})$.

The first direction was investigated in a previous work [1]. In this paper, we focus on the second procedure that is the identification of the mixing matrix $\boldsymbol{A}$.

1.1. Choice of Prior Distributions

Sources Model. We model the component s^j by a hidden Markov chain distribution. A basic presentation of this model is to consider it as a double stochastic process:

1. A continuous stochastic process $(s_1^j, s_2^j, \ldots, s_T^j)$ taking its values in $\mathbb{R}$.
2. A hidden discrete stochastic process $(z_1^j, z_2^j, \ldots, z_T^j)$ taking its values in $\{1..K_j\}$.

The $(z_t^j)_{t=1..T}$ form an homogeneous Markov chain with initial probability vector $[p_l = P(z_1^j = l)]_{l=1..K_j}$ and transition matrix $P_{lk} = [P(z_{t+1}^j = k \mid z_t^j = l)]_{l,k=1..K_j}$. Conditionally to this chain the source s^j is time independent:

$$p\big(s_{1..T}^j \mid z_{1..T}^j\big) = \prod_{t=1}^{T} p\big(s_t^j \mid z_t^j\big) \tag{5}$$

and has a Gaussian law $p(s_t^j \mid z_t^j = l) = \mathcal{N}(m_{jl}, \sigma_{jl})$.

This modeling is very convenient for at least two reasons:

- It is an interesting alternative to non parametric modeling.
- It is a convenient representation of weakly dependent phenomena.

HMM models were successfully applied to represent real speech signals and more elaborated HMM models can be found in [38].

The case of time independent hidden labels has been studied in [1, 32, 33].

Mixing Matrix Model. To account for some model uncertainty, we assign a Gaussian prior law to each element of the mixing matrix $\boldsymbol{A}$:

$$p(\boldsymbol{A}_{ij}) = \mathcal{N}\big(\boldsymbol{M}_{ij}, \sigma_{a,ij}^2\big) \tag{6}$$

which can be interpreted as knowing every element $(\boldsymbol{M}_{ji})$ with some uncertainty $(\sigma_{a,ij}^2)$. We underline here the advantage of estimating the mixing matrix $\boldsymbol{A}$ and not a separating matrix $\boldsymbol{B}$ (inverse of $\boldsymbol{A}$) which is the case of almost all the existing methods for source separation (see for example [39]). This approach has at least two advantages: (i) $\boldsymbol{A}$ does not need to be invertible ($n \neq m$), (ii) naturally, we have some *a priori* information on the mixing matrix not on its inverse which may not exist.

Choosing $\boldsymbol{M}_{ij} = 0$ and large values for $\sigma_{a,ij}^2$ corresponds to the classical case where we do not know a lot about this matrix. But, it happens that in some applications we have some prior knowledge about the elements of this matrix. For example, in the separation of cosmic microwave background observations, we may know or want to impose some soft constraints on these elements by fixing the means $\boldsymbol{M}_{ij}$ to the known values and choosing small values for the variances $\sigma_{a,ij}^2$.

Hyperparameters a Priori. We propose to assign an inverted Gamma prior $\mathcal{IG}(a, b)\,(a > 0$ and $b > 1)$ to mixture variances. This prior is necessary to avoid the likelihood degeneracy when some variances σ_{ij}^2 approach to zero together with noise variance. A more complete study of degeneracies in source separation problem is presented in [36].

2. Data Augmentation Algorithms

The sources $(s_t)_{t=1..T}$ are not directly observed, so that they form a second level of hidden variables, the first level being represented by the labels $(z_t^j)_{t=1..T}$ of the density mixture. Thus, the separation problem consists of two mixing operations, a mixture of densities which is a mathematical representation of our *a priori* distribution with unknown hyperparameters $\boldsymbol{\eta}_s$ and a real physical mixture of sources with unknown mixing matrix $\boldsymbol{A}$:

$$\begin{tikzcd}
z \arrow[r] & \boxed{\text{Mixing densities}} \arrow[r] & \boldsymbol{s} \arrow[r] & \boxed{\text{Mixing sources}} \arrow[r] & \oplus \arrow[r] & \boldsymbol{x} \\
 & \boxed{\boldsymbol{\eta}_s?} \arrow[u] & & \boxed{\boldsymbol{A}?} \arrow[u] & \epsilon \arrow[u] &
\end{tikzcd}$$

We have an incomplete data problem. The incomplete data are the observations $(\boldsymbol{x}_t)_{t=1..T}$, the missing data are the sources $(\boldsymbol{s}_t)_{t=1..T}$ and the vector labels $(z_t)_{t=1..T}$. The parameters to be estimated are $\boldsymbol{\theta} = (\boldsymbol{A}, \boldsymbol{\eta})$. This incomplete data structure suggests the development of restoration-maximization algorithms: Starting with an initial point $\boldsymbol{\theta}^0$, perform two steps:

- *Restoration*: Given the current estimate $\boldsymbol{\theta}^k$, any function of the missing data $f(s, z)$ is replaced by an attributed value f^k.
- *Maximization*: Find $\boldsymbol{\theta}^{k+1}$ which maximizes the penalized complete likelihood $p(\boldsymbol{x}, \boldsymbol{s}, \boldsymbol{z} \mid \boldsymbol{\theta})p(\boldsymbol{\theta})$.

The restoration step can be carried in three different manners:

1. f^k is the conditional expectation of $f(\boldsymbol{s}, \boldsymbol{z})$ which is computed given the current estimate of the parameter $\boldsymbol{\theta}^{(k-1)}$ at the previous iteration:

$$f^k = \int_{\boldsymbol{s},z} f(\boldsymbol{s}, z)p\big(\boldsymbol{s}, z \mid \boldsymbol{x}, \boldsymbol{\theta}^{(k-1)}\big)\,ds\,dz \qquad (7)$$

This leads to the classical EM algorithm. A fundamental property of the EM algorithm is the fact that it ensures the monotonous increasing of the incomplete likelihood function. Any value of $\boldsymbol{\theta}$ increasing the expected complete log-likelihood increases as well the incomplete log-likelihood, i.e., $\mathcal{L}(\boldsymbol{\theta}) \geq \mathcal{L}_i(\boldsymbol{\theta}_j)$. Moreover, $\hat{\boldsymbol{\theta}}$ is a critical point of the incomplete likelihood $p(\boldsymbol{x} \mid \boldsymbol{\theta})$ if and only if it is a fixed point of the re-estimation transformation. A more detailed description of the convergence properties of the EM algorithm can be found in [2].

2. The hidden variables are replaced by their maximum *a posteriori*. The *a posteriori* distribution is constructed given the observed data x and the current estimate $\boldsymbol{\theta}^{(k-1)}$. Here, we have two levels of hidden variables: the sources s and the labels z. Given z, the *a posteriori* of $\boldsymbol{s}$ is Gaussian so the computation of its mode $\hat{\boldsymbol{s}}$ and its covariance matrix can be done analytically. This remark leaded us to estimate first the labels z and then, like the EM algorithm, to replace any function of $\boldsymbol{s}$ by its *a posteriori* expectation value.
3. The hidden variables are sampled according to their *a posteriori* distribution. This strategy has the same scheme as the second strategy except that here the *a posteriori* distribution of labels are simulated and not summarized by just taking its maximum.

In the following, we give an overview of each strategy.

Exact EM Algorithm

The functional $\mathcal{Q} = E[\log p(\boldsymbol{x}, \boldsymbol{s}, \boldsymbol{z} \mid \boldsymbol{\theta}) + \log p(\boldsymbol{\theta}) \mid \boldsymbol{x}, \boldsymbol{\theta}^k]$, computed in the first step of the EM algorithm, is separable into three functionals $\mathcal{Q}_a$, $\mathcal{Q}_{\eta_g}$ and $\mathcal{Q}_{\eta_p}$

$$\mathcal{Q} = \mathcal{Q}_a + \mathcal{Q}_{\eta_g} + \mathcal{Q}_{\eta_p}$$

- The first functional $\mathcal{Q}_a$ depends on $\boldsymbol{A}$ and $\boldsymbol{R}_\epsilon$.
- The second functional $\mathcal{Q}_{\eta_g}$ depends on $\boldsymbol{\eta}_g = (m_{lk}, \sigma_{lk})_{l=1..n,k=1..K_l}$: means and variances of the Gaussian mixture.
- The third functional Q_{η_p} depends on $\eta_p = (\boldsymbol{p}_l, \boldsymbol{P}_l)_{l=1..n}$ initial probabilities and transition matrices of the Markov chains.

$\mathcal{Q}_a$-Maximization. The functional to be optimized at each iteration is:

$$\begin{aligned}\mathcal{Q}(\boldsymbol{A}, \boldsymbol{R}_\epsilon \mid \boldsymbol{\theta}^0) = {} & -\frac{T}{2} \log \mid 2\pi \boldsymbol{R}_\epsilon \mid \\ & - \frac{T}{2} \mathrm{Tr}\big(\boldsymbol{R}_\epsilon^{-1}(\boldsymbol{R}_{xx} - \boldsymbol{A}\boldsymbol{R}_{sx} - \boldsymbol{R}_{sx}^* \boldsymbol{A}^* \\ & + \boldsymbol{A}\boldsymbol{R}_{ss}\boldsymbol{A}^*)\big) + \log p(\boldsymbol{A}) \end{aligned} \tag{8}$$

where (*) refers to the matrix transpose.

Defining the following statistics:

$$\begin{cases} \boldsymbol{R}_{xx} = \dfrac{1}{T} \displaystyle\sum_{t=1}^{T} \boldsymbol{x}_t \boldsymbol{x}_t^* \\ \boldsymbol{R}_{sx} = \dfrac{1}{T} \displaystyle\sum_{t=1}^{T} E[\boldsymbol{s}_t \mid \boldsymbol{x}_{1..T}, \theta^0] \boldsymbol{x}_t^* \\ \boldsymbol{R}_{ss} = \dfrac{1}{T} \displaystyle\sum_{t=1}^{T} E[\boldsymbol{s}_t \boldsymbol{s}_t^* \mid \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0] \end{cases} \tag{9}$$

the updates of $\boldsymbol{A}$ and $\boldsymbol{R}_\epsilon$ become:

$$\begin{cases} \mathbf{Vec}\big(\boldsymbol{A}^{(k+1)}\big) = [T\hat{\boldsymbol{R}}_{ss}^* \otimes \boldsymbol{R}_\epsilon^{-1} + diag(Vec(\boldsymbol{\Gamma}))]^{-1} \\ \quad \times Vec\big(T\boldsymbol{R}_\epsilon^{-1}\hat{\boldsymbol{R}}_{xs} + \boldsymbol{\Gamma} \odot \boldsymbol{M}\big) \\ \boldsymbol{R}_\epsilon^{(k+1)} = \boldsymbol{R}_{xx} - \boldsymbol{A}^{(k+1)}\boldsymbol{R}_{sx} - \boldsymbol{R}_{xs}\big(\boldsymbol{A}^{(k+1)}\big)^* \\ \quad + \boldsymbol{A}^{(k+1)}\boldsymbol{R}_{ss}\big(\boldsymbol{A}^{(k+1)}\big)^* \end{cases} \tag{10}$$

where $\otimes$ is the Kronecker product [40], $\odot$ is the element-by-element product of two matrices, $\mathbf{Vec}(\cdot)$ is the column presentation of a matrix and $\boldsymbol{\Gamma}$ is the matrix $(1/\sigma_{a,ij}^2)$. Thus, we need to compute the conditional expectations $E[\boldsymbol{s}_t \mid \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0]$ and $E[\boldsymbol{s}_t\boldsymbol{s}_t^* \mid \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0]$. Generally:

$$\begin{aligned} E[f(\boldsymbol{s}_t) \mid \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0] = \sum E[f(\boldsymbol{s}_t) \mid \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0, z_t = \boldsymbol{i}] \\ \times p(z_t = \boldsymbol{i} \mid \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0) \end{aligned} \tag{11}$$

The vector $\boldsymbol{i} = [i_1, \ldots, i_n]$ belongs to $\mathcal{Z}_1 \times \mathcal{Z}_2 \times \ldots \mathcal{Z}_n$ with $\mathcal{Z}_l = \{1..K_l\}$. K_l is the number of Gaussians of each source component. Thus, we have $K = \prod_{l=1}^n K_l$ elements i in the previous sum.

The *a posteriori* expectations, given the variables $z = i$, are easily derived:

$$\begin{cases} E[\boldsymbol{s}_t \mid \boldsymbol{x}_t, \boldsymbol{\theta}^0, z_t = \boldsymbol{i}] & = \big[\boldsymbol{A}^*\boldsymbol{R}_\epsilon^{-1}\boldsymbol{A} + \boldsymbol{R}_i^{-1}\big]^{-1} \\ & \quad \times \big[\boldsymbol{A}^*\boldsymbol{R}_\epsilon^{-1}\boldsymbol{x}_t + \boldsymbol{R}_i^{-1}\boldsymbol{m}_i\big] \\ & = \boldsymbol{M}_{ti} \\ E[\boldsymbol{s}_t\boldsymbol{s}_t^* \mid \boldsymbol{x}_t, \boldsymbol{\theta}^0, z_t = \boldsymbol{i}] & = \big[\boldsymbol{A}^*\boldsymbol{R}_\epsilon^{-1}\boldsymbol{A} + \boldsymbol{R}_i^{-1}\big]^{-1} \\ & \quad + \boldsymbol{M}_{ti}\boldsymbol{M}_{ti}^* \end{cases} \tag{12}$$

However, the computation of the marginal probabilities $p(z_t = \boldsymbol{i} \mid \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0)$ represents the major part of the computation cost. The Baum-Welsh procedure [41] can be extended to the case when the sources are not directly observed. We define the Forward $\mathcal{F}_t(\boldsymbol{i})$ and Backward $\mathcal{B}_t(\boldsymbol{i})$ variables by:

$$\begin{cases} \mathcal{F}_t(\boldsymbol{i}) = P(z_t = \boldsymbol{i} \mid \boldsymbol{x}_{1..T}, \boldsymbol{\theta}) \\ \mathcal{B}_t(\boldsymbol{i}) = \dfrac{p(\boldsymbol{x}_{t+1..T} \mid z_t = \boldsymbol{i}, \boldsymbol{\theta})}{p(\boldsymbol{x}_{t+1..T} \mid \boldsymbol{x}_{1..T}, \boldsymbol{\theta})} \end{cases} \tag{13}$$

The computation of these variables is performed by recurrence formula as follows:

$$\begin{aligned} & \begin{cases} \mathcal{F}_1(\boldsymbol{i}) = M_1 p_i \mathcal{N}_{(\boldsymbol{A}\boldsymbol{m}_i, \boldsymbol{A}\boldsymbol{R}_i\boldsymbol{A}^* + \boldsymbol{R}_\epsilon)}[\boldsymbol{x}_1] \\ \mathcal{F}_t(\boldsymbol{i}) = M_t \displaystyle\sum_{\boldsymbol{j}} \mathcal{F}_{t-1}(\boldsymbol{j}) P_{ji} \mathcal{N}_{(\boldsymbol{A}\boldsymbol{m}_i, \boldsymbol{A}\boldsymbol{R}_i\boldsymbol{A}^* + \boldsymbol{R}_\epsilon)}[\boldsymbol{x}_t] \end{cases} \\ & \begin{cases} \mathcal{B}_T(\boldsymbol{i}) = 1 \\ \mathcal{B}_t(\boldsymbol{i}) = M_{t+1} \displaystyle\sum_{\boldsymbol{j}} \mathcal{B}_{t+1}(\boldsymbol{j}) P_{ij} \mathcal{N}_{(\boldsymbol{A}\boldsymbol{m}_j, \boldsymbol{A}\boldsymbol{R}_j\boldsymbol{A}^* + \boldsymbol{R}_\epsilon)}[\boldsymbol{x}_{t+1}] \end{cases} \end{aligned} \tag{14}$$

where the M_t are normalization constants:

$$\begin{cases} M_1 = \left[\displaystyle\sum_{\boldsymbol{i}} p_i \mathcal{N}_{(\boldsymbol{A}\boldsymbol{m}_i, \boldsymbol{A}\boldsymbol{R}_i\boldsymbol{A}^* + \boldsymbol{R}_\epsilon)}[\boldsymbol{x}_1] \right]^{-1} \\ M_t = \left[\displaystyle\sum_{\boldsymbol{i}} \sum_{\boldsymbol{j}} \mathcal{F}_{t-1}(\boldsymbol{j}) P_{ji} \mathcal{N}_{(\boldsymbol{A}\boldsymbol{m}_i, \boldsymbol{A}\boldsymbol{R}_i\boldsymbol{A}^* + \boldsymbol{R}_\epsilon)}[\boldsymbol{x}_t] \right]^{-1} \end{cases}$$

and

$$\boldsymbol{m}_i = \begin{pmatrix} \boldsymbol{m}_{i_1} \\ \vdots \\ m_{i_n} \end{pmatrix}, \quad \boldsymbol{R}_i = \begin{pmatrix} \sigma_{i_1}^2 & 0 & \ldots & 0 \\ 0 & \sigma_{i_2}^2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \\ \ldots & & & \sigma_{i_n}^2 \end{pmatrix}$$

Then $p(z_t = \boldsymbol{i} \mid \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0)$ is easily derived as:

$$p(z_t = \boldsymbol{i} \mid \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0) = \mathcal{F}_t(\boldsymbol{i})\mathcal{B}_t(\boldsymbol{i})$$

The spatial independence of sources components or more precisely the spatial independence of the labels implies:

$$\begin{cases} p_i & = \displaystyle\prod_{l=1}^{n} p_{i_l} = p_{i_1} \times p_{i_2} \ldots p_{i_n} \\ P_{ij} & = \displaystyle\prod_{l=1}^{n} P_{i_l j_l}^l \end{cases}$$

where p_{i_l} is the initial probability vector of the Markov chain of the component l and P^l its transition matrix.

The Forward-Backward computation complexity is of order $K^2 T$ where $K = \prod_{l=1}^{n} K_l$ is the number of the vectorial labels. We note that this complexity grows tremendously with the number of sources and the number of mixture components per source. If we choose the same number $K_l = k$ of mixture components for all the sources, the complexity $k^{2*n} T$ grows exponentially with the number of sources n.

$\mathcal{Q}_{\eta_g}$-Maximization. In order to establish the connection with the estimation of the parameters of hidden Markov models when the sources are directly observed and to elucidate the origin of the high computational cost of the hyperparameter re-estimation, we begin by the vectorial formula followed by the scalar expressions of interest:

The vector $\boldsymbol{i}$ refers to the vector label $(i_1, i_2, \ldots, i_n)^*$. The vector m_i designs $(m_{i_1}, m_{i_2} \ldots m_{i_n})^*$. The matrix $\boldsymbol{R}_i$ refers to diag $(\sigma_{i_1}^2, \sigma_{i_2}^2, \ldots, \sigma_{i_n}^2)$

The re-estimation of the vectorial means and covariances yields:

$$\begin{cases} \boldsymbol{m_i} = \dfrac{\sum_{t=1}^{T} E[\boldsymbol{s}_t \,|\, \boldsymbol{x}_t, z_{t=i}, \boldsymbol{\theta}^0] P(z_{t=i} \,|\, \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0)}{\sum_{t=1}^{T} P(z_{t=i} \,|\, \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0)} \\ \boldsymbol{R_i} = \dfrac{\sum_{t=1}^{T} [E(\boldsymbol{s}_t \boldsymbol{s}_t^*) - \boldsymbol{M}_{ti} \boldsymbol{m}_i^* - \boldsymbol{m}_i \boldsymbol{M}_{ti}^* + \boldsymbol{m_i} \boldsymbol{m}_i^*] P(z_t = \boldsymbol{i} \,|\, \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0) + 2b\boldsymbol{I}}{\sum_{t=1}^{T} P(z_t = \boldsymbol{i} \,|\, \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0) + 2(a-1)} \end{cases} \tag{15}$$

with $M_{ti} = E[\boldsymbol{s}_t \,|\, \boldsymbol{x}_t, z_t = \boldsymbol{i}, \boldsymbol{\theta}^0]$.

The re-estimation of the scalar means and variances is obtained by a spatial marginalization of the vector labels in the previous expressions:

$$\begin{cases} m_{lk} = \dfrac{\sum_{t=1}^{T} \sum_{(\boldsymbol{i} \,|\, \boldsymbol{i}(l)=k)} [E(\boldsymbol{s}_t \,|\, \boldsymbol{x}_t, z_t = \boldsymbol{i}, \boldsymbol{\theta}^0)]_l P(z_t = \boldsymbol{i} \,|\, \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0)}{\sum_{t=1}^{T} \sum_{(\boldsymbol{i} \,|\, \boldsymbol{i}(l)=k)} P(z_t = \boldsymbol{i} \,|\, \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0)} \\ \sigma_{lk}^2 = \dfrac{\sum_{t=1}^{T} \sum_{(\boldsymbol{i} \,|\, \boldsymbol{i}(l)=k)} ([E(\boldsymbol{s}_t \boldsymbol{s}_t^* \,|\, \boldsymbol{x}_t, z_t = \boldsymbol{i})]_{l,l} - m_{lk} [E(\boldsymbol{s} \,|\, \boldsymbol{x}_t, z_t = \boldsymbol{i})]_l + m_{lk}^2) P(z_t = \boldsymbol{i} \,|\, \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0) + 2b}{\sum_{t=1}^{T} \sum_{(\boldsymbol{i} \,|\, \boldsymbol{i}(l)=k)} P(z_t = \boldsymbol{i} \,|\, \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0) + 2(a-1)} \end{cases} \tag{16}$$

In the second expression of (16), We note the simple dependence of the variance update on the parameters a and b of the inverted Gamma prior which has the same form as in the non penalized case.

We can see clearly that, in addition to the marginalization in time to compute the quantities $P(z_t = \boldsymbol{i} \,|\, x_{1..T}, \boldsymbol{\theta}^0)$, we have to perform another marginalization in the spatial domain.

$\mathcal{Q}_{\eta_p}$-Maximization. The re-estimation of the initial probabilities and the stochastic matrices for the vectorial labels yields:

$$\begin{cases} p(\boldsymbol{i}) = P(z_1 = \boldsymbol{i} \,|\, \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0) \\ P(\boldsymbol{ij}) = \dfrac{\sum_{t=2}^{T} P(z_{t-1} = \boldsymbol{i}, z_t = \boldsymbol{j} \,|\, \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0)}{\sum_{t=2}^{T} P(z_{t-1} = \boldsymbol{i} \,|\, \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0)} \end{cases} \tag{17}$$

By the same way, the probabilities of the scalar labels are derived from the above expressions by spatial marginalization:

$$\begin{aligned} & p(\boldsymbol{i}(l) = k) = \sum_{(\boldsymbol{i} \,|\, \boldsymbol{i}(l)=k)} P(z_1 = \boldsymbol{i} \,|\, \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0) \\ & P(\boldsymbol{i}(l) = r, \boldsymbol{j}(l) = \boldsymbol{s}) \\ & = \frac{\sum_{t=2}^{T} \sum_{(\boldsymbol{i}, \boldsymbol{j} \,|\, \boldsymbol{i}(l)=r, \boldsymbol{j}(l)=\boldsymbol{s})} P(z_{t-1} = \boldsymbol{i}, z_t = \boldsymbol{j} \,|\, \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0)}{\sum_{t=2}^{T} \sum_{(\boldsymbol{i} \,|\, \boldsymbol{i}(l)=r)} P(z_{t-1} = \boldsymbol{i} \,|\, \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0)} \end{aligned} \tag{18}$$

The expressions of $P(z_{t-1} = i, z_t = j \,|\, x_{1..T}, \theta^0)$ are obtained directly from the Forward and Backward variables defined by (13):

$$\begin{aligned} & P(z_{t-1} = \boldsymbol{i}, z_t = \boldsymbol{j} \,|\, \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0) \\ & \quad = \mathcal{F}_{t-1}^0(i) P^0(\boldsymbol{i}, \boldsymbol{j}) \mathcal{N}_{(\boldsymbol{A}m_j, \boldsymbol{A}R_j \boldsymbol{A}^* + \boldsymbol{R}_\epsilon)}[\boldsymbol{x}_t] \mathcal{B}_t^0(\boldsymbol{j}) M_t \end{aligned}$$

Viterbi-EM Algorithm

When the number of labels $K = \prod_{l=1}^{n} K_l$ grows, the cost of the computation of the marginal probability $P(z_t = \boldsymbol{i} \,|\, \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0)$ and of the spatial marginalization for the re-estimation of the hyperparameters become very high. A solution to reduce the computational cost is to modify the restoration strategy. The labels are replaced by their maximum *a posteriori* values which corresponds to a classification step. This is performed by a relaxation strategy: At iteration k, $\hat{z}_t^k$ maximizes

$p(z_t \mid x_{1..T}, \hat{z}^k_{i<t}, \hat{z}^{k-1}_{i>t})$, which yields for $t = 1..T$:

$$z^k_t = \underset{l=1..K}{\operatorname{argmax}} \boldsymbol{T}_{[z^k_{t-1},l]} \phi(\boldsymbol{x}_t \mid \boldsymbol{\theta}_l, \boldsymbol{A}^k) \boldsymbol{T}_{[l, z^{k-1}_{t+1}]}$$

and

$$\begin{aligned} z^k_1 &= \underset{l=1..K}{\operatorname{argmax}} \phi(\boldsymbol{x}_1 \mid \boldsymbol{\theta}_l, \boldsymbol{A}^k) \boldsymbol{T}_{[l, z^{k-1}_2]} \\ z^k_T &= \underset{l=1..K}{\operatorname{argmax}} \boldsymbol{T}_{[z^k_{T-1},l]} \phi(\boldsymbol{x}_T \mid \boldsymbol{\theta}_l, \boldsymbol{A}^k) \end{aligned}$$

where $\boldsymbol{T}$ is the multidimensional transition matrix and $\phi(\boldsymbol{x} \mid \boldsymbol{\theta}_l, \boldsymbol{A}^k)$ the marginal distribution (s is integrated over) of x given the variable $z = \boldsymbol{l}$:

$$\begin{aligned} \phi(\boldsymbol{x} \mid \boldsymbol{\theta}_l, \boldsymbol{A}^k) &= \int_s p(\boldsymbol{x}, \boldsymbol{s} \mid z = \boldsymbol{l}, \boldsymbol{\theta}_l) ds \\ &= \mathcal{N}(\boldsymbol{x}, \boldsymbol{A}\boldsymbol{m}_l, \boldsymbol{A}\boldsymbol{R}_l\boldsymbol{A}^* + \boldsymbol{R}_\epsilon) \end{aligned}$$

Then, all the expectations involved in the EM algorithm are simply replaced by only one conditional expectation:

$$\begin{aligned} E[f(\boldsymbol{s}_t) \mid \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0] &= \sum_{\boldsymbol{i}} E[f(\boldsymbol{s}_t) \mid \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0, z_t = \boldsymbol{i}] \\ &\quad \times p(z_t = \boldsymbol{i} \mid \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0) \\ &\approx E[f(\boldsymbol{s}_t) \mid \boldsymbol{x}_{1..T}, \boldsymbol{\theta}^0, \hat{z}_t] \end{aligned}$$

Gibbs-EM Algorithm

The hidden labels z_t can also be generated according to their *a posteriori* distributions, which leads to a stochastic algorithm. Indeed, the advantage of this algorithm is double: reduction of the computational cost and the ability of the algorithm to avoid local maxima. The labels are generated by Gibbs sampling: At iteration k, $\hat{z}^k_t \sim p(z_t \mid \boldsymbol{x}_{1..T}, \hat{z}^k_{i<t}, \hat{z}^{k-1}_{i>t})$, which yields for $t = 1..T$:

$$z_t \sim T_{z_{t-1}z_t} \phi(\boldsymbol{x}_t \mid \boldsymbol{\theta}_z, \boldsymbol{A}^k) T_{z_t z_{t+1}}$$

and

$$\begin{aligned} z_1 &\sim \phi(\boldsymbol{x}_1 \mid \boldsymbol{\theta}_z, \boldsymbol{A}^k) T_{z_1 z_2} \\ z_T &\sim T_{z_{T-1} z_T} \phi(\boldsymbol{x}_T \mid \boldsymbol{\theta}_z, \boldsymbol{A}^k) \end{aligned}$$

This version of the Gibbs-EM algorithm has approximately the same computational cost as the Viterbi-EM algorithm because we have to compute the vector $[p(z_t = i \mid x_{1..T}, z_{s \neq t})]_{i=1..K}$.

As we have shown, the Viterbi and Gibbs versions of the EM algorithm reduces the computational cost due to the temporal structure of the discrete Markov chains $(z^j_t)^{j=1..n}_{t=1..T}$. The complexity K^2T where $K = \prod_{l=1}^n K_l$ of Forward-Backward computation is reduced with the Viterbi and Gibbs versions to KT (a reduction by a factor K). However, another source of a high computational cost is the number itself of the whole vector labels z: $K = \mid \mathcal{Z}_1 \times \mathcal{Z}_2 \times \ldots \mathcal{Z}_n \mid$. Its impact appears at two levels in the algorithms: First, in the computation of the K quantities $P(z_t = \boldsymbol{i} \mid \boldsymbol{x}_{1..T}, \boldsymbol{\theta})$ in the three proposed algorithms to, respectively, compute the expectations (11), estimate the hidden variables z and generate them according to their posterior. Second, in the spatial marginalization in the estimation of the hyperparameters $\boldsymbol{\eta}_g$ and $\boldsymbol{\eta}_p$ in the expressions (16) and (18). We show in the next section how we introduce a suitable approximation in order to reduce the computational cost due to the exponential number of the vector labels.

Fast Viterbi-EM Algorithm

The *a posteriori* distribution of the vector label z is:

$$\begin{aligned} p(z \mid \boldsymbol{x}, \boldsymbol{\theta}) &= \int_s p(z, s \mid \boldsymbol{x}, \boldsymbol{\theta}) ds \\ &\propto p(z) \int_s p(\boldsymbol{x} \mid s, \boldsymbol{\theta}) p(s \mid z, \boldsymbol{\theta}) ds \qquad (19) \end{aligned}$$

We see easily in the second line of the above equation that the distribution $p(\boldsymbol{x} \mid s, \boldsymbol{\theta})$ gives the components z^j of the vector z *a posteriori* a spatial dependence which is not the case *a priori* ($p(z) = \prod p(z^j)$). Consequently, to estimate or to generate the labels z^j, we need the manipulation of the whole vector z. This is the case, for example, when we want to compute the *a posteriori* marginal distribution of the component z^j, which needs the summation over all combinations of labels:

$$p(z_j(t) \mid \boldsymbol{x}, \boldsymbol{\theta}) = \sum_{z \in \mathcal{Z} \mid z(j) = z_j(t)} p(z(t) \mid \boldsymbol{x}(t), \boldsymbol{\theta}) \quad (20)$$

As solution to this issue, we introduce a relaxation strategy which consists in replacing the expression (20) by:

$$p(z_j(t) \mid \boldsymbol{x}, \boldsymbol{\theta}', \hat{s}_{l \neq j})$$

which is obtained by integrating only with respect to s_j, the other components are fixed and set to their MAP estimates in the previous iteration or drawn from their *a posteriori* distributions. Fixing the components $s_{l\neq j}$ breaks the vectorial structure of the mixture and reduces considerably the computational cost. In state of computing, at each time t, $k^n(k = K_1 = \cdots = K_n)$ probabilities $p(z_t \,|\, \boldsymbol{x}_t, \boldsymbol{\theta})$ in the Viterbi and Gibbs versions, we have with the relaxation strategy only $n \times k$ probabilities $(p(z_j(t) \,|\, \boldsymbol{x}, \boldsymbol{\theta}', \hat{s}_{l\neq j}))_{z=1..k}^{j=1..n}$. Moreover, the *a posteriori* distribution of the component s_j when fixing $s_{l\neq j}$ is a mixture of K_j Gaussians and its estimation is easier than dealing with the whole vector s which *a posteriori* distribution is a mixture of $\prod_{l=1}^n K_l$ multivariate Gaussians.

Now the Fast Viterbi algorithm contains a spatial relaxation (fixing $s_{l\neq j}$) besides its temporal relaxation (fixing $z_{i\neq t}$):

$$\begin{cases} z_j(t)^k = \underset{l=1..K_j}{\operatorname{argmax}} \boldsymbol{T}_{[z_{j,t-1}^k, l]} \phi(\boldsymbol{x}_t \,|\, s_{l\neq j}, \boldsymbol{\theta}_l, \boldsymbol{A}^k) \boldsymbol{T}_{[l, z_{j,t+1}^{k-1}]} \\ s_j \sim p(s_j \,|\, \boldsymbol{x}_t, z_j(t)^k, \boldsymbol{\theta}) \\ j = 1..n, t = 1..T \end{cases} \tag{21}$$

and

$$z_j(1)^k = \underset{l=1..K_j}{\operatorname{argmax}} \phi(\boldsymbol{x}_1 \,|\, s_{l\neq j}, \boldsymbol{\theta}_l, \boldsymbol{A}^k) \boldsymbol{T}_{[l, z_{j,2}^{k-1}]}$$
$$\{z_j(T)^k = \underset{l=1..K_j}{\operatorname{argmax}} \boldsymbol{T}_{[z_{j,T-1}^k, l]} \phi(\boldsymbol{x}_T \,|\, s_{l\neq j}, \boldsymbol{\theta}_l, \boldsymbol{A}^k)$$

where $\boldsymbol{T}$ is the transition matrix of the component j. We note that after each estimation of the label $z_j(t)^k$, the source component s_j is updated.

Fast Gibbs-EM Algorithm

The label components $z_j(t)$ are now generated according to their corresponding probabilities:

$$\begin{cases} z_j(t) \sim T_{z_{t-1} z_t} \phi(\boldsymbol{x}_t \,|\, s_{l\neq j}, \boldsymbol{\theta}_z, \boldsymbol{A}^k) T_{z_t z_{t+1}} \\ s_j \sim p(s_j \,|\, \boldsymbol{x}_t, z_j(t)^k, \boldsymbol{\theta}) \\ j = 1 \ldots n, t = 2 \ldots T - 1 \end{cases} \tag{22}$$

and

$$z_j(1) \sim \phi(\boldsymbol{x}_1 \,|\, s_{l\neq j}, \boldsymbol{\theta}_z, \boldsymbol{A}^k) T_{z_1 z_2}$$
$$z_j(T) \sim T_{z_{T-1} z_T} \phi(\boldsymbol{x}_T \,|\, s_{l\neq j}, \boldsymbol{\theta}_z, \boldsymbol{A}^k)$$

where $\boldsymbol{T}$ is the transition matrix of the component j.

The computational complexity concerning the update of the discrete probabilities is then reduced by a factor of about $\frac{\prod_{l=1}^n K_l}{\sum_{l=1}^n K_l}$. If the number of mixture components is the same for all the sources $k = K_1 = \cdots = K_l$, we note that the complexity is transformed from k^n to $n \times k$.

3. Simulation Results

To show the performances of the proposed algorithms, we consider the mixture of 2 sources:

- *Source 1*: The *a priori* distribution is a mixture of 4 Gaussians $(m, \sigma^2) \in \{(-3, 0.1), (-1, 0.1), (1, 0.1), (3, 0.1)\}$ with a transition matrix $\boldsymbol{T}_1$:

$$\boldsymbol{T}_1 = \begin{pmatrix} 0.9 & 0.05 & 0.03 & 0.02 \\ 0.8 & 0.1 & 0.05 & 0.05 \\ 0.7 & 0.02 & 0.08 & 0.2 \\ 0.5 & 0.2 & 0.2 & 0.1 \end{pmatrix}$$

- *Source 2*: The *a priori* distribution is a mixture of 4 Gaussians $(m, \sigma^2) \in \{(-3, 0.1), (-1, 0.1), (1, 0.1), (3, 0.1)\}$ with a transition matrix $\boldsymbol{T}_2$:

$$\boldsymbol{T}_2 = \begin{pmatrix} 0.25 & 0.25 & 0.25 & 0.25 \\ 0.25 & 0.25 & 0.25 & 0.25 \\ 0.25 & 0.25 & 0.25 & 0.25 \\ 0.25 & 0.25 & 0.25 & 0.25 \end{pmatrix}$$

The transition matrix $\boldsymbol{T}_1$ has a dominant first column, which means that the hidden labels z_t have a great probability to remain in the first class. However, the transition matrix $\boldsymbol{T}_2$ has the same line which leads to an i.i.d mixture. Figure 1 shows typical graphs of these signals. The two sources are mixed with a matrix $\boldsymbol{A} = \left(\begin{smallmatrix} 1 & 0.6 \\ -0.5 & 1 \end{smallmatrix}\right)$, a white Gaussian noise is added to the mixture with a covariance matrix $\boldsymbol{R}_\epsilon = \left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}\right)$ (SNR = 8 dB). The number of observations is 1000. Figure 1 illustrates typical graphs of the mixed sources $(x_1(t))_{t=1..T}$ and $(x_2(t))_{t=1..T}$.

In order to characterize the mixing matrix identification achievement, we use the performance index defined in [42]:

$$ind(S = \hat{\boldsymbol{A}}^{-1}\boldsymbol{A}) = \frac{1}{2}\left[\sum_i \left(\sum_j \frac{|S_{ij}|^2}{max_l |S_{il}|^2} - 1\right) + \sum_j \left(\sum_i \frac{|S_{ij}|^2}{max_l |S_{lj}|^2} - 1\right)\right]$$

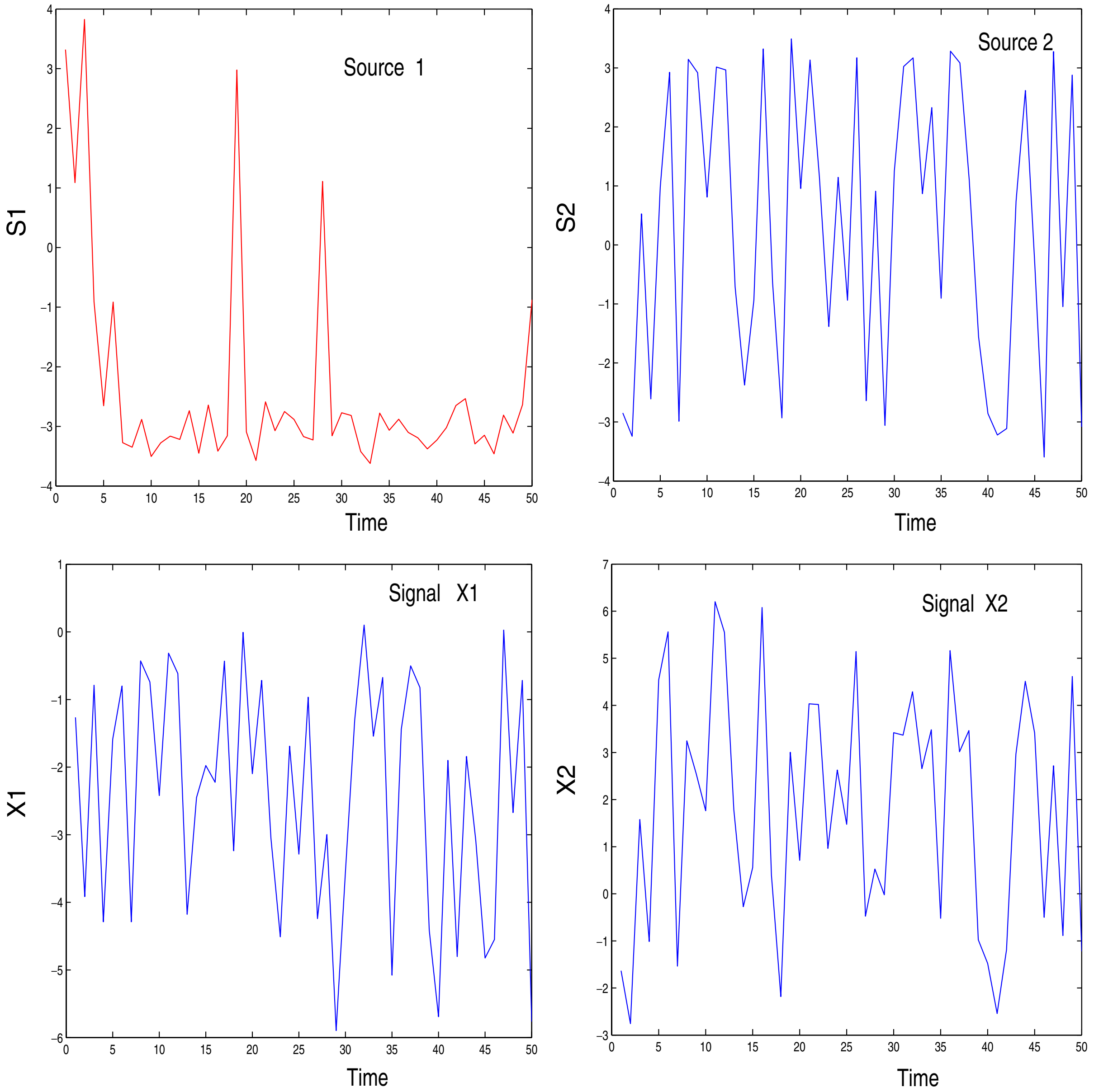

Figure 1. First line: Typical graphs of the sources s_1 and s_2. Even if in simulations we generated 1000 samples, here only 50 samples are shown. Second line: Typical graphs of the mixed sources $X_1 = a_{11}S_1 + a_{12}S_2$ and $X_2 = a_{21}S_1 + a_{22}S_2$.

Figure 2(a) illustrates the evolution of the mixing coefficient estimates with the exact EM algorithm through iterations. The horizontal line indicates the original value. Note the convergence of the algorithm close the original values after about 20 iterations. In these experiments, we fix the hyperparameters to their original values and we focus on the estimation of the mixing matrix in order to compare easily the different proposed algorithms to the exact EM algorithm. In fact, the hyperparameter estimation with the exact EM algorithm is very computational consuming. But, with the proposed Gibbs/Viterbi proposed algorithms, the hyperparameter estimation is easily performed and the convergence is little slower when we jointly estimate the hyperparameters (convergence after 100 iterations instead of 20 iterations as shown in Fig. 7). Figure 2(b) illustrates the convergence of the performance index with the EM algorithm to a satisfactory

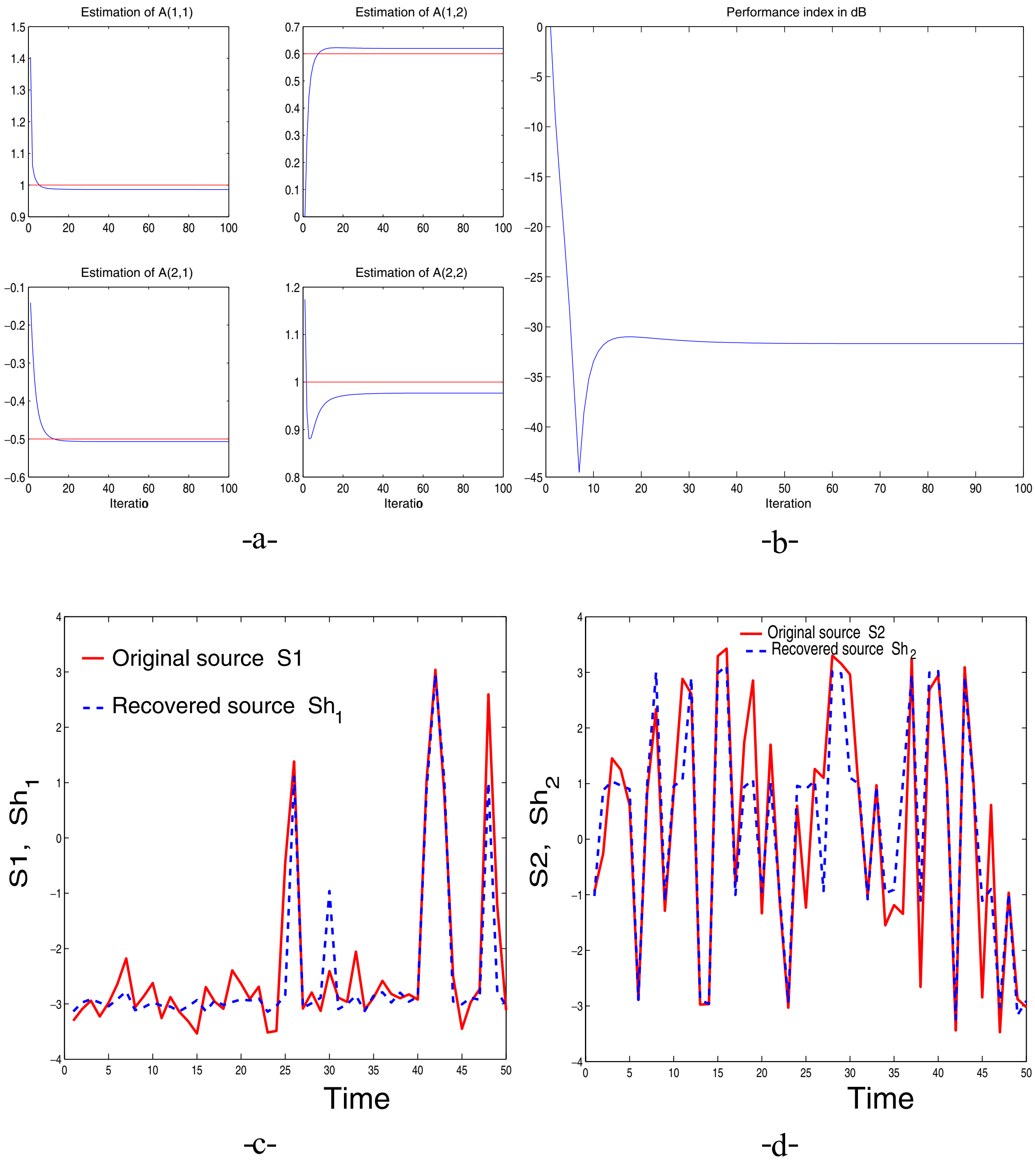

Figure 2. (a) Evolution through iterations of the estimates of the mixing coefficients with EM algorithm, (b) Evolution through iterations of the performance criteria with EM algorithm. (c) and (d) Results of the reconstruction of the two sources using the EM algorithm.

value of −31 dB. Figure 2(c) and (d) shows the results of the source reconstruction by plotting on the same graph the original sources and the recovered sources. Note the success of the algorithm to recover the sources.

Figure 3 shows the same simulation results with the Viterbi-EM algorithm. We can note an expected small bias for the estimation of the mixing matrix coefficients. We can explain this bias by the fact that we estimate jointly the hidden variables z_t in state of

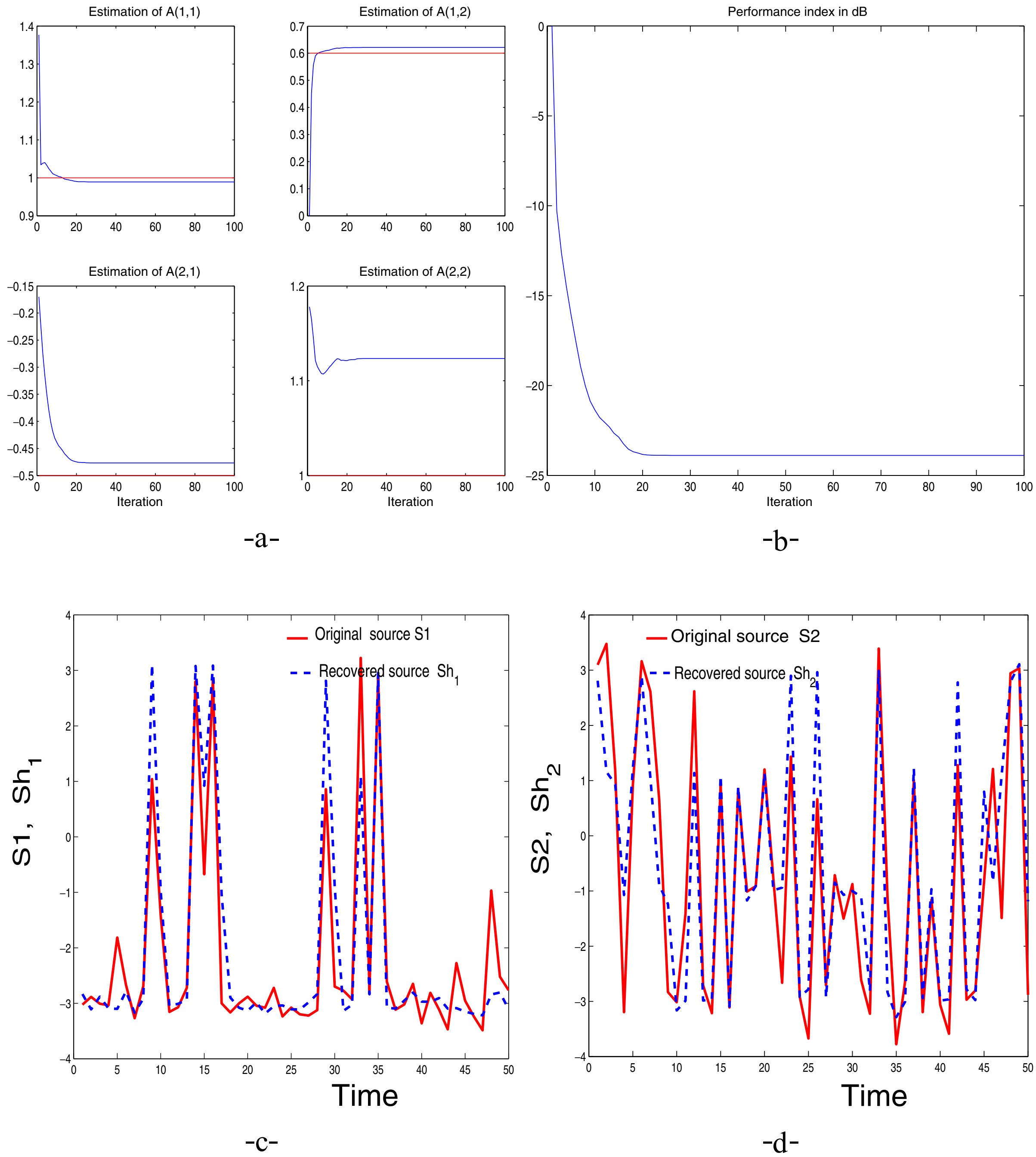

Figure 3. (a) Evolution through iterations of the estimates of the mixing coefficients with Viterbi-EM algorithm, (b) Evolution through iterations of the performance criteria with Viterbi-EM algorithm. (c) and (d) Results of the reconstruction of the two sources using the Viterbi-EM algorithm.

integrating it over the problem and so the estimate is biased with respect to the maximum likelihood estimate. However, the ML estimate itself can be biased when the number of observed data T is small because we have no more the efficiency of the likelihood estimation and the property that the maximum likelihood estimate is normally distributed around the true value of the parameter. The maximum likelihood

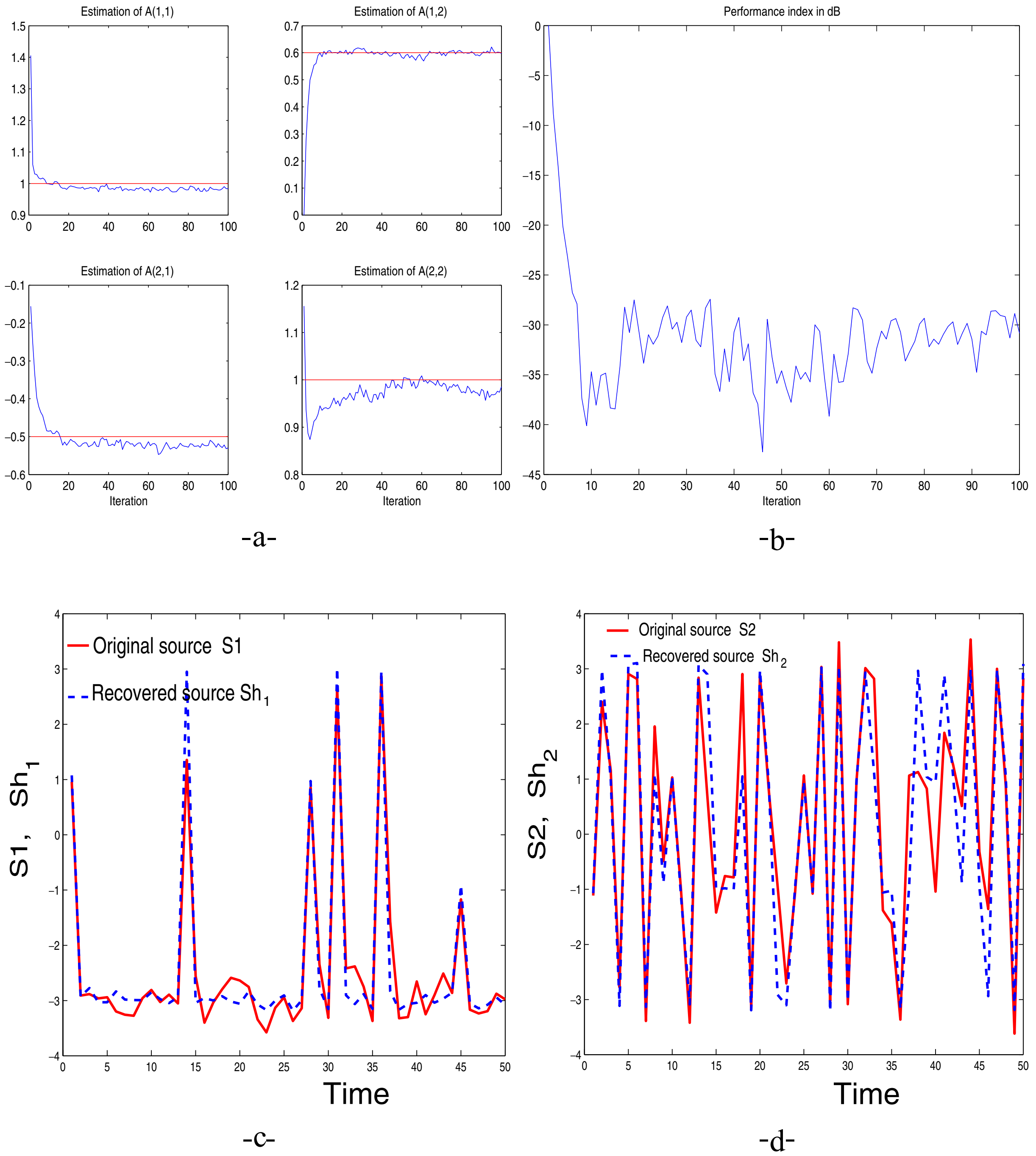

Figure 4. (a) Evolution through iterations of the estimates of the mixing coefficients with Gibbs-EM algorithm, (b) Evolution through iterations of the performance criteria with Gibbs-EM algorithm. (c) and (d) Results of the reconstruction of the two sources using the Gibbs-EM algorithm.

estimate is shown to be unbiased in the asymptotic case but with a moderate number of samples, we can loose this property. Therefore, the joint estimation of the hidden variables is not necessary worse than the optimization of the incomplete likelihood (note the bias with the EM estimate in Fig. 2(a). We note that the performance index has a satisfactory value of −24 dB. The computational cost reduction

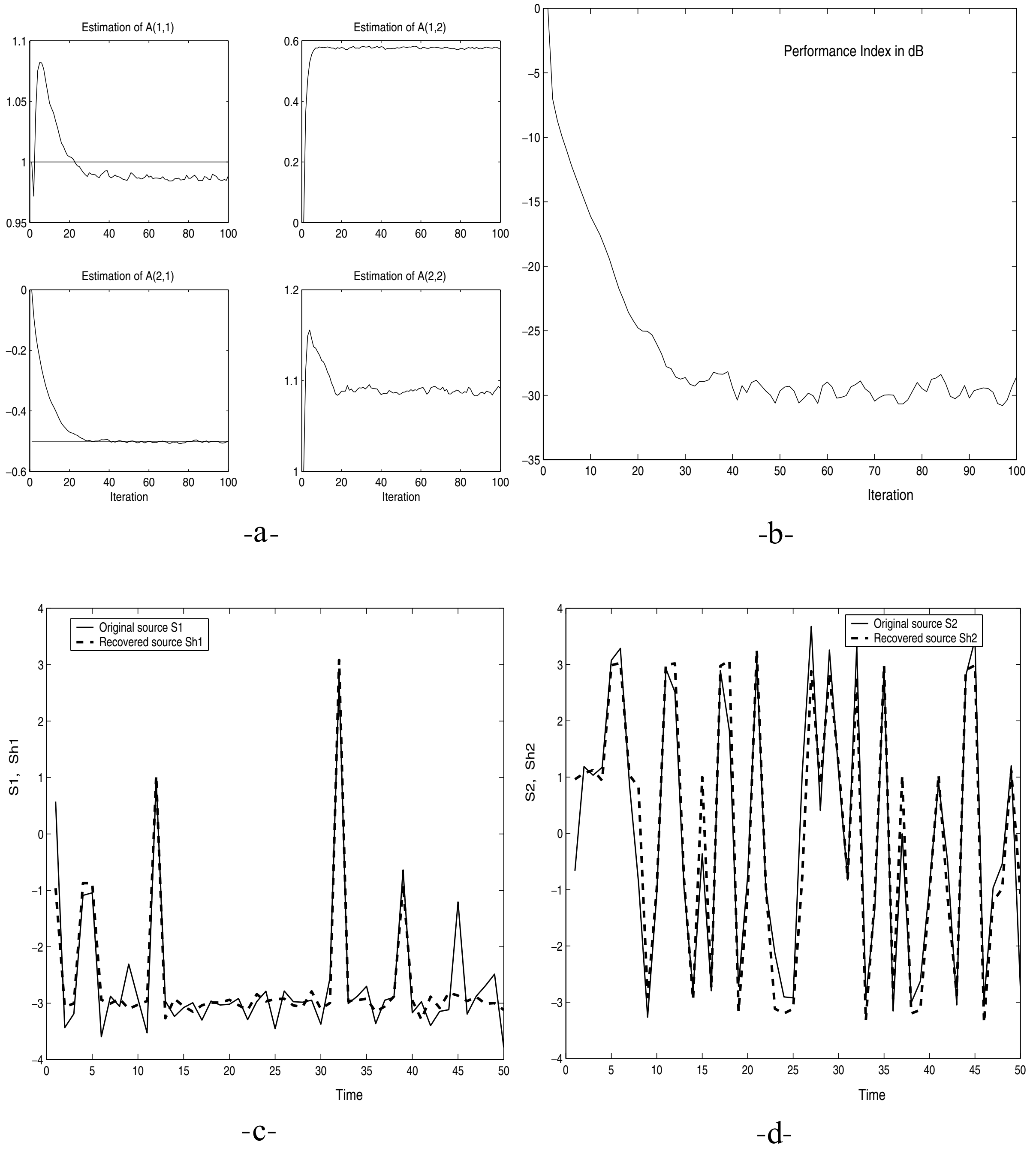

Figure 5. (a) Evolution through iterations of the estimates of the mixing coefficients with the Fast Viterbi algorithm, (b) Evolution through iterations of the performance criteria with the Fast Viterbi algorithm. (c) and (d) Results of the reconstruction of the two sources using the Fast Viterbi algorithm.

proportion with respect to the EM algorithm is about $K = 16$.

Figure 4 illustrates the results for the Gibbs-EM algorithm. We note the fluctuations due to the stochastic aspect of the algorithm but we can add a simulated annealing procedure to switch to the EM algorithm at convergence. The natural extension of the Gibbs-EM algorithm is to simulate the parameter $\boldsymbol{\theta}$ according to

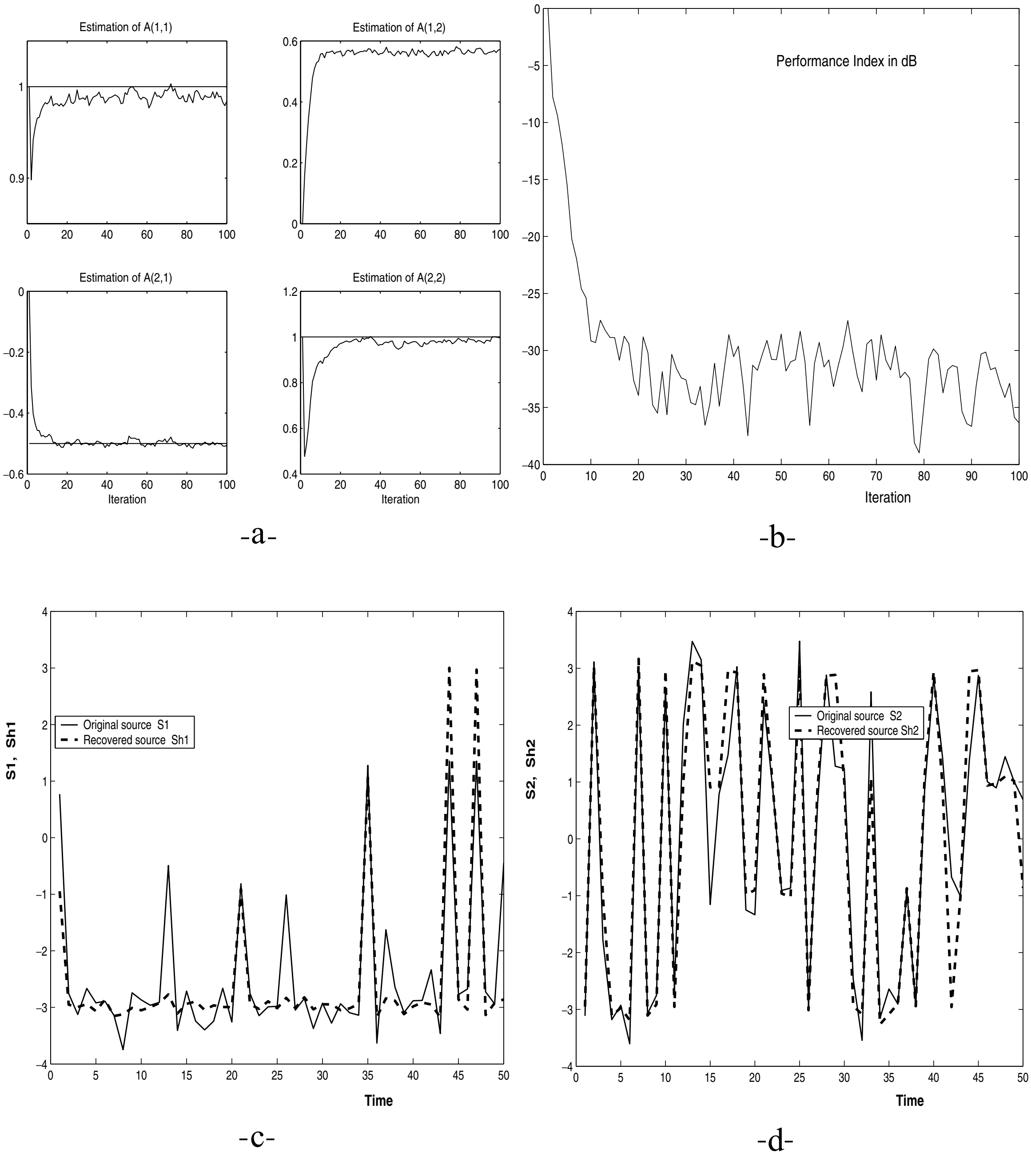

Figure 6. (a) Evolution through iterations of the estimates of the mixing coefficients with the Fast Gibbs algorithm, (b) Evolution through iterations of the performance criteria with the Fast Gibbs algorithm. (c) and (d) Results of the reconstruction of the two sources using the Fast Gibbs algorithm.

the complete likelihood and then we have a sequence $(z^k, \boldsymbol{\theta}^k)$ of generated variables and the Markov chain $(\boldsymbol{\theta}^k)$ has a stationary distribution which is its incomplete likelihood.

Figure 5 illustrates the results for the Fast Viterbi-EM algorithm. Figure 6 illustrates the results for the Fast Gibbs-EM algorithm. We note that the Fast versions have numerically the same convergence performances as the Gibbs/Viterbi algorithms but with a smaller time duration per iteration.

Conclusion

The estimation of the parameters of an hidden Markov model HMM is an incomplete data problem, the missing data being the labels of the mixture. Extending this problem to the blind separation of sources modeled by hidden Markov models introduces a second level of missing data which are the sources themselves. Therefore, restoration maximization algorithms represent a powerful tool for the estimation of the mixing matrix and the hyperparameters which are the HMM parameters. We proposed three different restoration maximization algorithms distinguished by their respective restoration strategies and having different convergence properties and complexities:

- Exact EM algorithm: The expectation functional is separable into three different parts corresponding to the three sets of parameters: those of $p(x|s, z)$, those of $p(s|z)$ and those of $p(z)$.
- Viterbi-EM algorithm: The labels are replaced by their maximum *a posteriori* MAP.
- Gibbs-EM algorithm: The labels are sampled according to their *a posteriori* distribution.

A relaxation step is proposed to accelerate the above algorithms when the number of source components and the number of mixture Gaussians grow. It is worth noting that in this paper we have supposed that the number of sources and the number of Gaussians are known. However, we are working on this problem that the Bayesian approach seems to be able to solve, by considering these numbers as unknown parameters to be estimated.

References

1. H. Snoussi and A. Mohammad-Djafari, “Bayesian Source Separation with Mixture of Gaussians Prior for Sources and Gaussian Prior for Mixture Coefficients,” in *Bayesian Inference and Maximum Entropy Methods*, A. Mohammad-Djafari (Ed.), Gif-sur-Yvette, France, July 2000, pp. 388–406, Proc. of MaxEnt, *Amer. Inst. Physics*.
2. A.P. Dempster, N.M. Laird, and D.B. Rubin, “Maximum Likelihood from Incomplete Data via the EM Algorithm,” *J. R. Statist. Soc. B*, vol. 39, 1977, pp. 1–38.
3. W. Qian and D.M. Titterington, “Bayesian Image Restoration: An Application to Edge-Preserving Surface Recovery,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, no. 7, 1993, pp. 748–752.
4. G. Celeux and J. Diebolt, “The SEM algorithm: A Probabilistic Teacher Algorithm Derived from the EM algorithm for the Mixture Problem,” *Comput. Statist. Quat.*, vol. 2, 1985, pp. 73–82.
5. A. Cichocki and R. Unbehaunen, “Robust Neural Networks with On-Line Learning for Blind Identification and Blind Separation of Sources,” *IEEE Trans. on Circuits and Systems*, vol. 43, no. 11, 1996, pp. 894–906.
6. S.J. Roberts, “Independent Component Analysis: Source Assessment, and Separation, a Bayesian Approach,” *IEE Proceedings—Vision, Image, and Signal Processing*, vol. 145, no. 3, 1998.
7. T. Lee, M. Lewicki, and T. Sejnowski, “Unsupervised Classification with non Gaussian Mixture Models Using ICA,” *Advances in Neural Information Processing Systems*, 1999, (in press).
8. T. Lee, M. Lewicki, and T. Sejnowski, “Independent Component Analysis using an Extended Infomax Algorithm for Mixed Sub-Gaussian and Super-Gaussian Sources,” *Neural Computation*, vol. 11, no. 2 1999, pp. 409–433.
9. T. Lee, M. Girolami, A. Bell, and T. Sejnowski, “A Unifying Informationtheoretic Framework for Independent Component Analysis,” *Int. Journal of Computers and Mathematics with Applications Computation*, 1999, (in press).
10. I. Ziskind and M. Wax, “Maximum Likelihood Localization of Multiple Sources by Alternating Projection,” *IEEE Trans. Acoust. Speech, Signal Processing*, vol. ASSP-36, no. 10, 1988, pp. 1553–1560.
11. M. Wax, “Detection and Localization of Multiple SSources via the Stochastic Signals Model,” *IEEE Trans. Signal Processing*, vol. 39, no. 11, 1991, pp. 2450– 2456.
12. J.-F. Cardoso, “Infomax and Maximum Likelihood for Source Separation,” *IEEE Letters on Signal Processing*, vol. 4, no. 4, 1997, pp. 112–114.
13. J.-L. Lacoume, “A Survey of Source Separation,” in *Proc. First International Conference on Independent Component Analysis and Blind Source Separation ICA’99*, Aussois, France, Jan. 11–15, 1999, pp. 1–6.
14. E. Oja, “Nonlinear PCA Criterion and Maximum Likelihood in Independent Component Analysis,” in *Proc. First International Conference on Independent Component Analysis and Blind Source Separation ICA’99*, Aussois, France, Jan. 11–15, 1999, pp. 143–148.
15. R.B. MacLeod and D.W. Tufts, “Fast Maximum Likelihood Estimation for Independent Component Analysis,” in *Proc. First International Conference on Independent Component Analysis and Blind Source Separation ICA’99*, Aussois, France, January 11–15, 1999, pp. 319–324.
16. O. Bermond and J.-F. Cardoso, “Approximate Likelihood for Noisy Mixtures,” in *Proc. First International Conference on Independent Component Analysis and Blind Source Separation ICA’99*, Aussois, France, Jan. 11–15, 1999, pp. 325–330.
17. P. Comon, C. Jutten, and J. Herault, “Blind Separation of Sources .2. Problems Statement,” *Signal Processing*, vol. 24, no. 1, 1991, pp. 11–20.
18. C. Jutten and J. Herault, “Blind Separation of Sources .1. An Adaptive Algorithm based on Neuromimetic Architecture,” *Signal Processing*, vol. 24, no. 1, 1991, pp. 1–10.
19. E. Moreau and B. Stoll, “An Iterative Block Procedure for the Optimization of Constrained Contrast Functions,” in *Proc. First*

International Conference on Independent Component Analysis and Blind Source Separation ICA'99, Aussois, France, Jan. 11–15, 1999, pp. 59–64.

20. P. Comon and O. Grellier, "Non-linear Inversion of Underdetermined Mixtures," in *Proc. First International Conference on Independent Component Analysis and Blind Source Separation ICA'99*, Aussois, France, Jan. 11– 15, 1999, pp. 461–465.
21. J.-F. Cardoso and B. Laheld, "Equivariant Adaptive Source Separation," *IEEE Trans. on Sig. Proc.*, vol. 44, no. 12, 1996, pp. 3017–3030.
22. A. Belouchrani, K. Abed Meraim, J.-F. Cardoso, and Éric Moulines, "A Blind Source Separation Technique Based on Second order Statistics," *IEEE Trans. on Sig. Proc.*, vol. 45, no. 2, 1997, pp. 434–44.
23. S.-I. Amari and J.-F. Cardoso, "Blind Source Separation—Semiparametric Statistical Approach," *IEEE Trans. on Sig. Proc.*, vol. 45, no. 11, 1997, pp. 2692–2700.
24. J.-F. Cardoso, "Blind Signal Separation: Statistical Principles," *Proceedings of the IEEE*, vol. 90, no. 8, pp. 2009–2026, Oct. 1998, *Special Issue on Blind Identification and Estimation*, R.-W. Liu and L. Tong (Eds.).
25. J.J. Rajan and P.J.W. Rayner, "Decomposition and the Discrete Karhunenloeve Transformation Using a Bayesian Approach," *IEE Proceedings-Vision, Image, and Signal Processing*, vol. 144, no. 2, 1997, pp. 116–123.
26. K. Knuth, "Bayesian Source Separation and Localization," in *SPIE'98 Proceedings: Bayesian Inference for Inverse Problems*, A. Mohammad-Djafari (Ed.), San Diego, CA, July 1998, pp. 147–158.
27. K.H. Knuth and H.G. Vaughan Jr., "Convergent Bayesian Formulations of Blind Source Separation and Electromagnetic Source Estimation," in *Maximum Entropy and Bayesian Methods, Munich 1998*, W. von der Linden, V. Dose, R. Fischer, and R. Preuss (Eds.), Dordrecht, Kluwer, 1999, pp. 217–226.
28. S.E. Lee and S.J. Press, "Robustness of Bayesian Factor Analysis Estimates," *Communications in Statistics—Theory And Methods*, vol. 27, no. 8, 1998.
29. K. Knuth, "A Bayesian Approach to Source Separation," in *Proceedings of the First International Workshop on Independent Component Analysis and Signal Separation: ICA'99*, C.J.J.-F. Cardoso and P. Loubaton (Eds.), Aussios, France, 1999, pp. 283–288.
30. T. Lee, M. Lewicki, M. Girolami, and T. Sejnowski, "Blind Source Separation of more Sources than Mixtures Using Overcomplete Representation," *IEEE Signal Processing Letters*, 1999 (in press).
31. A. Mohammad-Djafari, "A Bayesian Approach to Source Separation," in *Bayesian Inference and Maximum Entropy Methods*, J.R.G. Erikson and C. Smith (Eds.), Boise, IH, July 1999, MaxEnt Workshops, Amer. Inst. Physics.
32. O. Bermond, *Méthodes statistiques pour la séparation de Sources*, Phd thesis, Ecole Nationale Supérieure des Télécommunications, 2000.
33. H. Attias, "Blind Separation of Noisy Mixture: An EM Algorithm for Independent Factor Analysis," *Neural Computation*, vol. 11, 1999, pp. 803–851.
34. R.J. Hathaway, "A Constrained EM Algorithm for Univariate Normal Mixtures," *J. Statist. Comput. Simul.*, vol. 23, 1986, pp. 211–230.
35. A. Ridolfi and J. Idier, "Penalized Maximum Likelihood Estimation for Univariate Normal Mixture Distributions," in *Actes 17e coll. GRETSI*, Vannes, France, Sept. 1999, pp. 259–262.
36. H. Snoussi and A. Mohammad-Djafari, "Penalized Maximum Likelihood for Multivariate Gaussian Mixture," in *Bayesian Inference and Maximum Entropy Methods*, MaxEnt Workshops, Aug. 2001, to appear in Amer. Inst. Physics.
37. A. Belouchrani, "Séparation Autodidacte de Sources: Algorithmes, Performances et Application à des Signaux Expérimentaux," Phd thesis, Ecole Nationale Supérieure des Télécommunications, 1995.
38. Z. Ghahramani and M. Jordan, "Factorial Hidden Markov Models," *Machine Learning*, no. 29, 1997, pp. 245–273.
39. J. Cardoso and B. Labeld, "Equivariant Adaptative Source Separation," *Signal Processing*, vol. 44, 1996, pp. 3017–3030.
40. J.W. Brewer, "Kronecker Products and Matrix Calculus in System Theory," *IEEE Trans. Circ. Syst.*, vol. CS-25, no. 9, 1978, pp. 772–781.
41. L.R. Rabiner and B.H. Juang, "An Introduction to Hidden Markov Models," *IEEE ASSP Mag.*, 1986, pp. 4–16.
42. E. Moreau and O. Macchi, "High-Order Contrasts for Self-Adaptative Source Separation," *Adaptative Control Signal Process*, vol. 10, 1996, pp. 19–46.

Hichem Snoussi was born in Bizerta, Tunisia, in 1976. He received the diploma degree in electrical engineering from the École Supérieure d'Électricité (Supélec), Gif-sur-Yvette, France, in 2000. He also received the DEA degree in signal processing from the Université de Paris-Sud, Orsay, France, in 2000. Since 2000, he has been working towards his Ph.D at the Laboratoire des Signaux et Systèmes, Centre National de la Recherche scientifique. His research interests include Bayesian technics for source separation, information geometry and latent variable models.
snoussi@lss.supelec.fr

Ali Mohammad-Djafari was born in Iran. He received the B.Sc. degree in electrical engineering from Polytechnique of Teheran,

in 1975, the diploma degree (M.Sc.) from Ecole Supérieure d'Electricité (Supélec), Gif sur Yvette, France, in 1977 and the "Docteur-Ing_enieur" (Ph.D.) degree and "Doctorat d'Etat" in Physics, from the Université Paris-Sud (UPS), Orsay, France, respectively in 1981 and 1987. He was Associate Professor at UPS for two years (1981–1983). Since 1984, he has a permanent position at "Centre National de la Recherche Scientifique (CNRS)" and works at "Laboratoire des Signaux et Systémes (L2S)" at Supélec. From 1998 to 2002, he has been at the head of Signal and Image Processing division at this laboratory. Presently, he is "Directeur de recherche" and his main scientific interests are in developing new probabilistic methods based on Information Theory, Maximum Entropy and the Bayesian inference approaches for inverse problems in general, and more specifically signal and image reconstruction and restoration. The main application domain of his interests are Computed Tomography (X rays, PET, SPECT, MRI, microwave, ultrasound and eddy current imaging) either for medical imaging or for non destructive testing (NDT) in industry.
djafari@lss.supelec.fr