Circuits Syst Signal Process (2017) 36:2697–2726
DOI 10.1007/s00034-016-0424-2

An Expectation-Maximization Algorithm for Blind Separation of Noisy Mixtures Using Gaussian Mixture Model

Fanglin Gu[1] · **Hang Zhang**[2] · **Wenwu Wang**[3] · **Shan Wang**[1]

Received: 28 October 2015 / Revised: 11 September 2016 / Accepted: 13 September 2016 /
Published online: 25 October 2016

Abstract In this paper, we propose a new expectation-maximization (EM) algorithm, named GMM-EM, to blind separation of noisy instantaneous mixtures, in which the non-Gaussianity of independent sources is exploited by modeling their distribution using the Gaussian mixture model (GMM). The compatibility between the incomplete-data structure of the GMM and the hidden variable nature of the source separation problem leads to an efficient hierarchical learning and alternative method for estimating the sources and the mixing matrix. In comparison with conventional blind source separation algorithms, the proposed GMM-EM algorithm has superior performance for the separation of noisy mixtures due to the fact that the covariance matrix of the additive Gaussian noise is treated as a parameter. Furthermore, the GMM-EM algorithm works well in underdetermined cases by incorporating any prior information one may have and jointly estimating the mixing matrix and source signals in a Bayesian framework. Systematic simulations with both synthetic and real speech signals are used to show

✉ Shan Wang
chinafir@nudt.edu.cn

Fanglin Gu
gu.fanglin@gmail.com

Hang Zhang
hangzh_2002@163.com

Wenwu Wang
w.wang@surrey.ac.uk

[1] College of Electronic Science and Engineering, National University of Defense Technology, Changsha 410073, China

[2] College of Communication Engineering, PLA University of Science and Technology, Nanjing 210073, China

[3] Department of Electrical and Electronic Engineering, University of Surrey, Guildford GU2 7XH, UK

the advantage of the proposed algorithm over conventional independent component analysis techniques, such as FastICA, especially for noisy and/or underdetermined mixtures. Moreover, it can even achieve similar performance to a recent technique called null space component analysis with less computational complexity.

Keywords Blind source separation · Gaussian mixture model · Expectation-maximization · Underdetermined mixture

1 Introduction

Blind source separation (BSS) aims to estimate unknown sources from the observed sensor signals without (or with very limited) prior information about the sources and how the sources propagate to the sensors. It has drawn great attention due to its wide range of applications in signal processing. Many algorithms have been developed to solve the BSS problem based on the assumption that the sources to be recovered are statistically independent, leading to a family of well-known methods called independent component analysis (ICA) [1,2,5,6,23,41], such as the information maximization based Infomax algorithm [5], the joint approximate diagonalization of eigenmatrices (JADE) algorithm [23] and the FastICA algorithm [41]. However, most of the ICA methods are developed for the case of determined/overdetermined mixtures (i.e., the number of sources P is equal to or smaller than the number of sensors Q), and consider a noiseless source separation model. These methods are not directly applicable to the problem of source separation from noisy and/or underdetermined mixtures.

There have been a number of attempts to extend the ICA approach in order to address the noisy and/or underdetermined BSS problem, such as the noise-model-based FastICA algorithm [26], cumulant-based separation algorithms [22,36], and the characteristic functions-based blind identification methods [10,15,24,27]. These methods have offered new ideas for estimating the mixing system; however, their performance is still limited for recovering the source signals. The underdetermined source separation problem is, in particular, very challenging since, as opposed to dealing with determined/over-determined mixtures, even provided with the information about the mixing system, the sources cannot be uniquely reconstructed, simply because, for $P > Q$, where P is the number of sources and Q is the number of sensors, the mixing matrix is not invertible. In contrast to the ICA approach, the recently proposed null space component analysis (NCA) approach can solve the noisy and/or underdetermined BSS problem effectively [21,32]. Given a set of signals, the NCA approach constructs an operator for each signal so that only the signal of interest is in the operator's null space, and all the other signals are excluded. Furthermore, an additional constraint on the rank of the operators is imposed to remove the rotation ambiguity.

In fact, the methods discussed above can be considered as special cases under the Bayesian framework. In a Bayesian technique, a statistical model defined by a set of parameters is used to describe the source separation problem [8,13,14,31]. The parameters of the model can be inferred from the acquired data, with the help of some prior information about the physical system under consideration. As compared with the classical ICA methods [1,2,5,6,23,41], the Bayesian approach provides advantages in several scenarios. For example, the Bayesian approach is often much more robust to noise since the noise levels in the data are taken into account through the parameterization of the noise covariance matrix within the Bayesian model [8,13]. Second, the Bayesian approach also enables any prior knowledge about the physical application systems to be exploited in the model where appropriate prior distributions for the unknown parameters can be assigned. Moreover, the BSS problem can be reformulated as a problem of joint maximum a posteriori (MAP) probability estimation of the mixing matrix and the sources, and as a result, the Bayesian approach can be extended to address the underdetermined BSS problem [14,31].

Under the Bayesian framework, a number of BSS approaches have been presented in the literature. Belouchrani et al. [3] developed a maximum-likelihood (ML) method for jointly estimating the mixing matrix and noise covariance matrix via the expectation-maximization (EM) algorithm [18,19] where the sources are drawn from a finite alphabet set. However, many natural signals, such as speech signals, are continuous signals (rather than the discrete sources). It has been shown in [20] that many PDFs can be closely approximated by a finite-order Gaussian mixture model (GMM) via the Kullback–Leibler (KL) divergence [4]. Following this route, several GMM-based BSS approaches have been proposed. For example, an approximate ML approach was developed by Moulines et al. [38] for blind separation and deconvolution of noisy linear mixtures. This method primarily considered the use of GMMs to model the distribution of sources, and the parameters of this model, along with the unknown mixing matrix were optimized to best represent the observed data. Some related works can also be found in [3,13,14,17,31,42], in which different mixture models, such as generalized Gaussian mixture model [3], are used. The application of these models for blind separation of underdetermined mixtures has been studied in [3,14,31]. On the other hand, an alternative strategy, in which the GMM is fitted to the observed data, rather than the sources, is developed in [33,40]. In this approach, the mixtures are separated by finding the rotation matrix that approximately diagonalizes all of the correlation matrices resulting from the GMM. However, it is limited to the noiseless determined case.

Recently, we proposed an EM algorithm for separating noisy determined /underdetermined mixtures with non-stationary sources, in which the continuous density hidden Markov model (CDHMM) is used to model the PDF and to track the non-stationarity of the sources [11]. Preliminary study on synthesized data has shown great potentials of this algorithm, despite the challenge in initializing appropriately the large number of hyper-parameters in practical scenarios. In practice, the distribution model plays a vital role in Bayesian approach. On one hand, the distribution model is required to depict as many distribution forms as possible to make the approach more flexible and to potentially improve separation performance. On the other hand, it is also required to involve as few parameters as possible such that the Bayesian approach can be implemented easily. Hence, there is a trade-off between the generalization of the distribution model and estimation precision.

In this paper, we also consider the challenging noisy and/or underdetermined BSS problem. In order to address the above issue, we propose to exploit the non-Gaussianity of the sources by modeling their distributions using a GMM, and to incorporate prior information by assigning conjugate priors for the parameters of the GMM and mixing matrix for improving the separation performance. In such a case, the BSS problem can be treated as a problem of estimating parameters from incomplete data. The EM algorithm is probably the most well-known algorithm for obtaining the ML estimates in parametric models for incomplete data. It is an iterative algorithm alternating between the E-step and M-step, respectively. In the E-step, the conditional expectation of the complete-data log-likelihood is computed on the basis of the observed data and parameter estimates. In the M-step, the parameters are estimated by maximizing the complete-data log-likelihood from the E-step. Therefore, an EM algorithm is proposed for obtaining the MAP estimates of the mixing matrix, the sources and the noise covariance matrix in a joint manner. Although there are some similar works in the literature, our approach differs from these works in the following aspects. First, different from [38], the conjugate priors used for incorporating prior information have not been considered. Second, as opposed to the variational Bayesian method in [42], a new GMM-EM method is used to obtain the MAP estimates of the sources and parameters due to its advantage of providing fast and stable convergence [9]. Thanks to the prior information incorporated by the conjugate priors, the proposed EM method works well even for underdetermined mixtures. Third, in comparison with the method based on CDHMM [11], the proposed GMM-EM method is easier to implement.

The remainder of this paper is organized as follows. In Sect. 2, the BSS problem and the assumptions made in our work are presented. The source distribution model based on GMM is given in Sect. 3. The notations describing the prior laws for the mixing coefficients, noise covariance matrix and the model parameters are presented in Sect. 4. In Sect. 5, a new GMM-EM algorithm is derived for the estimation of the mixing coefficients, the noise covariance matrix and the model parameters, in order to estimate the source signals. Issues regarding the practical implementation and performance of the proposed algorithm are discussed in Sect. 6, where the initialization scheme for the parameters, the convergence performance and computational complexity are analyzed. In Sect. 7, simulations are provided to show the performance of the proposed algorithm. Finally, conclusions are drawn in Sect. 8.

2 Problem Formulation

We consider the well-known instantaneous linear mixing model given as [41]

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{w}(t), \quad t = 1, \ldots, T \tag{1}$$

The random vector $\mathbf{s}(t) = [s_1(t), \ldots, s_P(t)]^{\mathrm{T}}$, representing P statistically independent sources at discrete time instance t, is mixed by a time-invariant unknown mixing matrix $\mathbf{A}$, where $(\cdot)^{\mathrm{T}}$ denotes transpose operator. The observation vector $\mathbf{x}(t) = [x_1(t), \ldots, x_Q(t)]^{\mathrm{T}}$ is obtained from an array of Q sensors and contaminated by the noise vector $\mathbf{w}(t) = [w_1(t), \ldots, w_Q(t)]^{\mathrm{T}}$ which is assumed to be Gaussian white with zero mean and unknown covariance matrix $\mathbf{R}_w = \mathrm{diag}([\sigma_1^2, \ldots, \sigma_Q^2])$ and independent of $\mathbf{s}(t)$, where $\mathrm{diag}(\cdot)$ is an operator for forming a diagonal matrix with the elements of the specified vector on the main diagonal.

Different from the determined and/or noiseless models investigated in many existing contributions, here, we consider the more practical situations where the mixtures may be corrupted by noise or the mixing system is underdetermined. Our objective in this paper is therefore to develop an algorithm for recovering the source signals from noisy and/or underdetermined mixtures. To this aim, we propose to reconstruct the source signals $\{\mathbf{s}(t)\}_{t=1,\ldots,T}$ and the mixing matrix $\mathbf{A}$ in a joint manner under the Bayesian framework on the basis of the observed signals $\{\mathbf{x}(t)\}_{t=1,\ldots,T}$ and the assignment of some prior information.

3 Source Distribution Model

This section describes the source model based on GMM. The PDF of the ith source signal at time instance t is modeled by the GMM as follows

$$f_s\left(s_i(t), \boldsymbol{\theta}_i\right) = \sum_{l_i=1}^{N_i} \alpha_{i,l_i} \mathscr{N}\left(s_i(t); \mu_{i,l_i}, \sigma^2_{i,l_i}\right), \quad i = 1, \ldots, P \tag{2}$$

where $\mathscr{N}(\cdot; \cdot, \cdot)$ denotes a Gaussian density function and N_i denotes the number of Gaussians. The mixing weights are denoted by $\{\alpha_{i,l_i}\}_{l_i}^{N_i}$, such that $\sum_{l_i=1}^{N_i} \alpha_{i,l_i} = 1$. The means and variances of the Gaussians are denoted by $\{\mu_{i,l_i}\}_{l_i}^{N_i}$ and $\{\sigma^2_{i,l_i}\}_{l_i}^{N_i}$, respectively. Assuming that the source signals are statistically independent, the joint PDF of the sources can be formulated as follows [17]

$$\begin{aligned}
f_{\mathbf{s}}\left(\mathbf{s}(t); \Theta\right) &= \prod_{i=1}^{P} f_s\left(s_i(t); \boldsymbol{\theta}_i\right) \\
&= \sum_{l_1}^{N_1} \alpha_{1,l_1} \mathscr{N}\left(s_1(t); \mu_{1,l_1}, \sigma^2_{1,l_1}\right) \\
&\quad \sum_{l_2}^{N_2} \alpha_{2,l_2} \mathscr{N}\left(s_2(t); \mu_{2,l_2}, \sigma^2_{2,l_2}\right) \\
&\quad \ldots \sum_{l_P}^{N_P} \alpha_{P,l_P} \mathscr{N}\left(s_P(t); \mu_{P,l_P}, \sigma^2_{P,l_P}\right) \\
&= \sum_{l_1}^{N_1} \sum_{l_2}^{N_2} \cdots \sum_{l_P}^{N_P} \omega_{l_1,l_2,\ldots,l_P} \\
&\quad \mathscr{N}\left(\left[s_1(t), s_2(t) \ldots, s_P(t)\right]^{\mathrm{T}};\right. \\
&\quad \left[\mu_{1,l_1}, \mu_{2,l_2}, \ldots, \mu_{P,l_P}\right]^{\mathrm{T}}, \\
&\quad \left.\operatorname{diag}\left(\sigma^2_{1,l_1}, \sigma^2_{2,l_2}, \ldots, \sigma^2_{P,l_P}\right)\right) \\
&= \sum_{m=1}^{M} \omega_m \mathscr{N}\left(\mathbf{s}(t); \boldsymbol{\mu}_m, \mathbf{C}_m\right)
\end{aligned} \tag{3}$$

where $M = \prod_{i=1}^{P} N_i$ is the total number of Gaussians in the joint PDF and $\omega_m = \prod_{i=1}^{P} \alpha_{i,l_i}; m = 1, \ldots, M$ are the mixing weights of each Gaussian component such that $\sum_{m=1}^{M} \omega_m = 1$. The index denotes a unique combination of the Gaussian components from each source, i.e., $l_1, \ldots, l_P \rightarrow m$, where $l_i \in \{1, \ldots, N_i\}$ denotes a Gaussian index of the source. The mean vector and covariance matrix of the Gaussian are denoted by $\boldsymbol{\mu}_m = [\mu_{1,l_1}, \mu_{2,l_2}, \ldots, \mu_{P,l_P}]^{\mathrm{T}}$ and $\mathbf{C}_m = \mathrm{diag}(\sigma^2_{1,l_1}, \sigma^2_{2,l_2}, \ldots, \sigma^2_{P,l_P})$, respectively. It can be observed from (3) that the joint PDF of the sources is a multivariate GMM parameterized by a diagonal covariance matrix [11].

4 Choices of Prior Densities

In this section, we discuss how to choose the prior distributions to incorporate prior information for improving the performance of blind separation. The prior distribution is used to attribute uncertainty rather than randomness to the unknown parameter or latent variable. The hyper-parameters of the priors are chosen to reflect any existing information. In the Bayesian framework, the aim for solving BSS problem is to obtain the posterior distribution of the relevant parameters. Generally, let z be a random variable, and ϑ be the relevant parameter. According to the Bayesian theorem, the posterior distribution can be represented as the product of the likelihood function $f(z|\vartheta)$ and prior $f(\vartheta)$, normalized by the probability of the data $f(z)$

$$f(\vartheta|z) = \frac{f(z|\vartheta)f(\vartheta)}{\int f(z|\vartheta)f(\vartheta)\mathrm{d}\vartheta} \tag{4}$$

The likelihood function is usually well determined from the data-generating process and can be considered fixed. Therefore, the difficulty in calculating the integral in the denominator in the right-hand side (RHS) of the above equation will depend on the choice of the prior distributions. With conjugate priors,[1] the posterior distribution will have the same algebraic form as the prior distribution (but with different parameter values and also depending on the likelihood function if the form of the likelihood function is varied). This essentially reduces the difficulty involved in the calculation of the numerical integrations as described above, as the conjugate priors [7,30,35] can provide a closed-form expression for the posterior distribution. Moreover, the use of conjugate priors does not prevent the proposed EM algorithm from choosing flexible forms of the density functions, such as Gaussian, Laplacian, Gamma or other members in the exponential family, which covers a wide range of distributions for the mixing matrix and the noise covariance. The choice of the hyper-parameters of the priors will be discussed later in Sect. 6.2. Next, we discuss the choice of the prior densities for the mixing matrix and noise covariance matrix, as in [11], and for the source models, as in [40].

[1] For a likelihood function, a conjugate prior is defined as the prior for which the posteriori and the priori are of the same type of distributions.

4.1 Prior Density for Mixing Matrix

To account for some model uncertainty, we assign a Gaussian prior law to each element of the mixing matrix $\mathbf{A}$

$$g\left(a_{ij}|\mu_{ij},\sigma_{ij}^2\right)=\mathscr{N}\left(\mu_{ij},\sigma_{ij}^2\right) \tag{5}$$

With (5), some constraints can be imposed on the elements of the mixing matrix, i.e., by assigning some known values to the means μ_{ij} with σ_{ij} chosen for small values to reflect the degree of the uncertainty [12]. Assuming that the elements of the mixing matrix are independent from each other, it is straightforward to derive that $g(\text{vec}(\mathbf{A}))=\prod_{i=1}^{Q}\prod_{j=1}^{P}g(a_{ij})$, where vec$(\cdot)$ is an operator for obtaining a vector by stacking the columns of a matrix one beneath the other. It is straightforward to get

$$g(\text{vec}(\mathbf{A}))=\mathscr{N}\left(\boldsymbol{\mu}_A,\mathbf{\Lambda}\right) \tag{6}$$

where $\boldsymbol{\mu}_A=[\mu_{11},\ldots\mu_{1P},\ldots,\mu_{Q1},\ldots,\mu_{QP}]^{\text{T}}$ and $\mathbf{\Lambda}$ is a diagonal matrix whose elements are σ_{ij}^2.

4.2 Prior Density for Noise Covariance Matrix

Covariance matrices are symmetric positive semi-definite matrices. To model the prior knowledge about them, Wishart distribution, which is a generalization of the univariate Chi-square distribution, is often used. The Wishart distribution is a conjugate density which therefore has another advantage in simplifying the GMM-EM process as described in Sect. 5. Therefore, as in [11], the Wishart density is assigned as the prior density of the noise covariance matrix $\mathbf{R}_w$, defined as

$$g(\mathbf{R}_w^{-1}|\Sigma_w^{-1},v_R)\propto\left|\mathbf{R}_w^{-1}\right|^{(v_R-Q-1)/2}\exp\left[-\frac{1}{2}\text{tr}\left(\Sigma_w\mathbf{R}_w^{-1}\right)\right] \tag{7}$$

where Σ_w is a $Q\times Q$ positively defined symmetric matrix, v_R is a scalar greater than $Q-1$, tr$(\cdot)$ denotes the trace of a squared matrix, $|\cdot|$ indicates the determinant of a squared matrix, and exp$(\cdot)$ represents exponential operator.

4.3 Prior Density for Parameters of the Source Model

The conjugate prior density assignment for the parameters $\Theta=\{\omega_m,\boldsymbol{\mu}_m,\mathbf{C}_m\}_{m=1}^{M}$ in (3) is more complicated. According to [34], however, we can interpret a finite mixture density as a density associated with a statistical population, denoted as a mixture of M component populations weighted by the coefficients $(\omega_1,\ldots,\omega_M)$. Therefore, we can regard $f(\mathbf{s}(t);\Theta)$ as a marginal PDF of the joint PDF of the parameters Θ. More specifically, it can be computed as the product of a multinomial density and multivariate Gaussian densities, which denote the sizes of the populations and the densities of individual components, respectively [34]. If the joint density of the weighting parameters is a multinomial distribution, then a practical candidate for modeling the prior knowledge of these parameters is a conjugate density such as the Dirichlet density

$$g\left(\omega_1, \ldots, \omega_M | \eta_1, \ldots, \eta_M\right) \propto \prod_{m=1}^{M} \omega_m^{\eta_m - 1} \tag{8}$$

where $\eta_m > 0$ are the hyper-parameters for the Dirichlet density. As for the parameters $(\boldsymbol{\mu}_m, \mathbf{C}_m)$ of the individual Gaussian mixture component, the following conjugate density is chosen

$$\begin{aligned} g\left(\boldsymbol{\mu}_m, \mathbf{C}_m^{-1} | \tau_m, \mathbf{u}_m, \upsilon_m, \Sigma_m^{-1}\right) \propto & \left|\mathbf{C}_m^{-1}\right|^{(\upsilon_m - P)/2} \\ & \exp\left[-\frac{\tau_m}{2}\left(\boldsymbol{\mu}_m - \mathbf{u}_m\right)^{\mathrm{T}} \mathbf{C}_m^{-1}\left(\boldsymbol{\mu}_m - \mathbf{u}_m\right)\right] \\ & \exp\left[-\frac{1}{2}\mathrm{tr}\left(\Sigma_m \mathbf{C}_m^{-1}\right)\right] \end{aligned} \tag{9}$$

where $(\tau_m, \mathbf{u}_m, \upsilon_m, \Sigma_m)$ are the prior density hyper-parameters such that $\upsilon_m > P - 1$, $\tau_m > 0$, $\mathbf{u}_m$ is a vector of dimension P, and Σ_m is a $P \times P$ positive definite matrix.

Assuming that the parameters of the individual mixture components and the mixture weights are independent [34], then, the joint prior density $g(\Theta)$ can be computed as the product of the prior densities defined in (8) and (9), respectively, given as follows,

$$g(\Theta) = g\left(\omega_1, \ldots, \omega_M\right) \prod_{m=1}^{M} g\left(\boldsymbol{\mu}_m, \mathbf{C}_m^{-1}\right) \tag{10}$$

5 Bayesian Blind Separation

In this section, equipped with the source model discussed in Sect. 3 and prior densities defined in Sect. 4, we develop a new EM algorithm under the Bayesian framework for the BSS problem as described in Sect. 2. For the convenience of analysis, we employ a probability-generative model, as depicted in Fig. 1, where a graphical model is used to show the process of generating an observation signal at time instant t based on the mixture model. Apparently, there are two levels of hidden variables in this graphical model, with the first level being represented by the Gaussian component labels $\{y(t)\}_{t=1,\ldots,T}$ of the density mixture, and the second level by the source signals $\{\mathbf{s}(t)\}_{t=1,\ldots,T}$.

As a result, the BSS problem in essence can be treated as a problem of estimating parameters from incomplete data. The incomplete data are the observations $\mathbf{X} = \{\mathbf{x}(t)\}_{t=1,\ldots,T}$, while the missing data are the sources $\mathbf{S} = \{\mathbf{s}(t)\}_{t=1,\ldots,T}$ and the unobserved Gaussian component labels of the density mixture $Y = \{y(t)\}_{t=1,\ldots,T}$. The parameters that need to be estimated are $\mathbf{A}$, $\mathbf{R}_w$ and Θ. The EM algorithm is

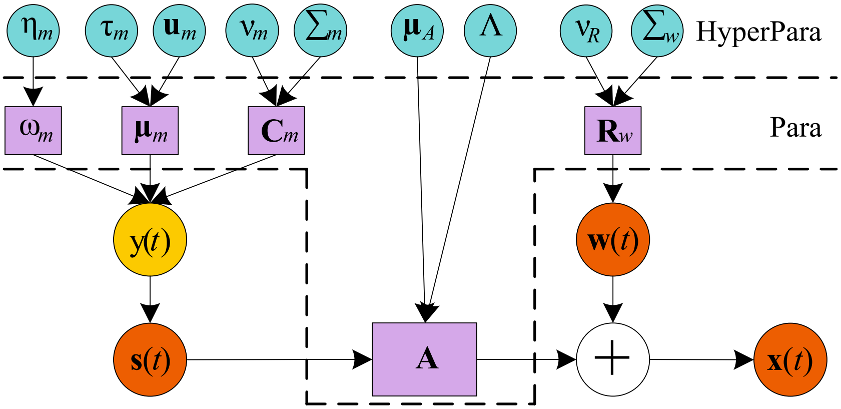

Fig. 1 Probability-generative model of observed signals at the discrete time instance t

a commonly used method for inferring the parameters of an underlying distribution from incomplete data based on the ML/MAP scheme [25]. Therefore, similar to the method we adopted in [11], we derive an GMM-EM algorithm in this work to obtain the MAP estimates of the unknowns including the mixing matrix, noise covariance matrix and the parameters of the source model, as detailed below.

5.1 The E-Step

Given the observed data $\mathbf{X}$ and the current parameter estimates, the E-step of the GMM-EM algorithm aims to obtain the expected value of the complete-data log-likelihood $\log f(\mathbf{X}, \mathbf{S}, Y|\mathbf{A}, \mathbf{R}_w, \Theta)$ with respect to the unknown data $\mathbf{S}$ and Y. The evaluation of this expectation is called the E-step of the algorithm. To this end, we define an auxiliary function as

$$\begin{aligned} & J\left(\mathbf{A}, \mathbf{R}_w, \Theta, \mathbf{A}^g, \mathbf{R}_w^g, \Theta^g\right) \\ & \quad = \mathrm{E}\left[\log f(\mathbf{X}, \mathbf{S}, Y|\mathbf{A}, \mathbf{R}_w, \Theta)|\mathbf{X}, \mathbf{A}^g, \mathbf{R}_w^g, \Theta^g\right] \end{aligned} \tag{11}$$

where $\mathbf{A}^g$, $\mathbf{R}_w^g$, and Θ^g are the current estimates of the parameters that we use to evaluate the expectation and $\mathbf{A}$, $\mathbf{R}_w$ and Θ are the new parameters that we optimize to increase J. E[$\cdot$] is an expectation operator.

Since $\mathbf{X}$ and $\mathbf{A}^g, \mathbf{R}_w^g, \Theta^g$ are constants, $\mathbf{A}, \mathbf{R}_w, \Theta$ are variables that we wish to adjust, and $\mathbf{s}$, Y are random variables governed by the distribution $f(\mathbf{S}, Y|\mathbf{X}, \mathbf{A}^g, \mathbf{R}_w^g, \Theta^g)$, the RHS of (11) can be rewritten as

$$\begin{aligned} & \mathrm{E}\left[\log f\left(\mathbf{X}, \mathbf{S}, Y|\mathbf{A}, \mathbf{R}_w, \Theta\right)|\mathbf{X}, \mathbf{A}^g, \mathbf{R}_w^g, \Theta^g\right] \\ & \quad = \mathrm{E}\left[f\left(\mathbf{S}, Y|\mathbf{X}, \mathbf{A}^g, \mathbf{R}_w^g, \Theta^g\right)\log f\left(\mathbf{X}, \mathbf{S}, Y|\mathbf{A}, \mathbf{R}_w, \Theta\right)\right] \end{aligned} \tag{12}$$

After a series of derivations (more details can be found in “Appendix 1”), we can get.

$$\begin{aligned} J = \sum_{t=1}^{T}\sum_{m=1}^{M}\int_{\mathbf{s}} & f\left(y(t)=m|\mathbf{s}(t),\mathbf{X},\mathbf{A}^{g},\mathbf{R}_{w}^{g},\Theta^{g}\right) \\ & \log\omega_{m} f(\mathbf{s}(t)|y(t)=m,\Theta)\mathrm{d}\mathbf{s} \\ & +\sum_{t=1}^{T}\int_{\mathrm{s}} f\left(\mathbf{s}(t)|\mathbf{X},\mathbf{A}^{g},\mathbf{R}_{w}^{g},\Theta^{g}\right)\log f\left(\mathbf{x}(t)|\mathbf{s}(t),\mathbf{A},\mathbf{R}_{w}\right)\mathrm{d}\mathbf{s} \end{aligned} \tag{13}$$

It is clear that the posterior distribution $f\left(\mathbf{s}(t)|\mathbf{X},\mathbf{A}^{g},\mathbf{R}_{w}^{g},\Theta^{g}\right)$ is indispensable for the evaluation of the expectation in (13). In practice, it can be proved that (more details can be found in “Appendix 2”)

$$f\left(\mathbf{s}(t)|\mathbf{X},\mathbf{A}^{g},\mathbf{R}_{w}^{g},\Theta^{g}\right)=\sum_{m=1}^{M}\tilde{\omega}_{mt}^{g}\mathcal{N}\left(\mathbf{s}(t);\tilde{\boldsymbol{\mu}}_{mt}^{g},\tilde{\mathbf{C}}_{mt}^{g}\right) \tag{14}$$

where

$$\begin{cases} \tilde{\mathbf{C}}_{mt}^{g} = \left((\mathbf{A}^{g})^{\mathrm{T}}(\mathbf{R}_{w}^{g})^{-1}\mathbf{A}^{g}+(\mathbf{C}_{m}^{g})^{-1}\right)^{-1} \\ \tilde{\boldsymbol{\mu}}_{mt}^{g} = \left(\tilde{\mathbf{C}}_{mt}^{g}\right)\left((\mathbf{A}^{g})^{\mathrm{T}}(\mathbf{R}_{w}^{g})^{-1}\mathbf{x}(t)+(\mathbf{C}_{m}^{g})^{-1}\boldsymbol{\mu}_{m}^{g}\right) \\ \tilde{\omega}_{mt}^{g} = \omega_{m}^{g}\left(\left|(\tilde{\mathbf{C}}_{mt}^{g})\right|^{1/2}\Big/\left|2\pi\mathbf{R}_{w}^{g}\right|^{1/2}\left|\mathbf{C}_{m}^{g}\right|^{1/2}\right) \\ \qquad \exp\left\{-\frac{1}{2}\left[\mathbf{x}^{\mathrm{T}}(t)(\mathbf{R}_{w}^{g})^{-1}\mathbf{x}(t)\right.\right. \\ \qquad \left.\left.+(\boldsymbol{\mu}_{m}^{g})^{\mathrm{T}}(\mathbf{C}_{m}^{g})^{-1}\boldsymbol{\mu}_{m}^{g}-(\tilde{\boldsymbol{\mu}}_{mt}^{g})^{\mathrm{T}}(\tilde{\mathbf{C}}_{mt}^{g})^{-1}\tilde{\boldsymbol{\mu}}_{mt}^{g}\right]\right\} \end{cases} \tag{15}$$

5.2 The M-Step

This step maximizes the expectation of the complete-data log-likelihood as shown in (13) with respect to $\mathbf{A}$, $\mathbf{R}_{w}$ and Θ, and the maximum point is then taken as the new parameters. It can be observed that the first term in the RHS of (13) is dependent on the parameters Θ of the GMM model, while the second term is determined by the mixing matrix $\mathbf{A}$ and the noise covariance matrix $\mathbf{R}_{w}$. For this reason, the auxiliary function in the RHS of (13) can be split into two parts J_1 and J_2, i.e.,

$$J=J_1+J_2 \tag{16}$$

where

$$\begin{cases} J_1=\sum\limits_{t=1}^{T}\int_{\mathrm{s}} f\left(\mathbf{s}(t)|\mathbf{x},\mathbf{A}^{g},\mathbf{R}_{\mathbf{w}}^{g},\Theta^{g}\right) \\ \qquad\qquad \log f\left(\mathbf{x}(t)|\mathbf{s}(t),\mathbf{A},\mathbf{R}_{\mathbf{w}}\right)\mathrm{d}\mathbf{s} \\ J_2=\sum\limits_{t=1}^{T}\sum\limits_{m=1}^{M}\int_{\mathbf{s}} f\left(y(t)=m|\mathbf{s}(t),\mathbf{x},\mathbf{A}^{g},\mathbf{R}_{\mathbf{w}}^{g},\Theta^{g}\right) \\ \qquad\qquad \log\omega_{m} f\left(\mathbf{s}(t)|y(t)=m,\Theta\right)\mathrm{d}\mathbf{s} \end{cases} \tag{17}$$

Accordingly, the parameters $\mathbf{A}$, $\mathbf{R}_w$ and Θ can be estimated by optimizing the auxiliary functions J_1 and J_2, respectively, as explained in the next two subsections.

5.2.1 Estimation Formula for the Mixing Matrix $\mathbf{A}$ *and the Noise Covariance Matrix* $\mathbf{R}_w$

First of all, the updating rules for the mixing matrix $\mathbf{A}$ and the noise covariance matrix $\mathbf{R}_w$ are discussed based on the auxiliary function J_1. Note that the auxiliary function J_1 in (17) can be converted into the following form as

$$\begin{aligned} J_1 &= -\frac{T}{2}\log|2\pi\mathbf{R}_w| \\ &\quad -\frac{T}{2}\mathrm{tr}\left[\mathbf{R}_w^{-1}\left(\mathbf{R}_{xx} - \mathbf{A}R_{sx} - \mathbf{R}_{sx}^{\mathrm{T}}\mathbf{A}^{\mathrm{T}} + \mathbf{A}R_{ss}\mathbf{A}^{\mathrm{T}}\right)\right] \end{aligned} \tag{18}$$

where

$$\begin{cases} \mathbf{R}_{xx} = \frac{1}{T}\sum\limits_{t=1}^{T}\mathbf{x}(t)\mathbf{x}^{\mathrm{T}}(t) \\ \mathbf{R}_{sx} = \frac{1}{T}\sum\limits_{t=1}^{T}\mathrm{E}\left[\mathbf{s}(t)|\mathbf{x},\mathbf{A}^{g},\mathbf{R}_w^{g},\Theta^{g})\right]\mathbf{x}^{\mathrm{T}}(t) \\ \mathbf{R}_{ss} = \frac{1}{T}\sum\limits_{t=1}^{T}\mathrm{E}\left[\mathbf{s}(t)\mathbf{s}^{\mathrm{T}}(t)|\mathbf{x},\mathbf{A}^{g},\mathbf{R}_w^{g},\Theta^{g}\right] \end{cases} \tag{19}$$

Note that the prior information for the mixing matrix $\mathbf{A}$ denoted by J_A^g is related to its conjugate prior as shown in (6). Therefore, we can obtain the MAP auxiliary function for the mixing matrix $\mathbf{A}$ by incorporating its prior information, which gives

$$\begin{aligned} \hat{J}_A &= J_1 + J_A^g \\ &= -\frac{T}{2}\log|2\pi\mathbf{R}_w| - \frac{1}{2}\log(|\boldsymbol{\Lambda}|) \\ &\quad -\frac{T}{2}\mathrm{tr}\left[\mathbf{R}_w^{-1}(\mathbf{R}_{xx} - \mathbf{A}R_{sx} - \mathbf{R}_{sx}^{\mathrm{T}}\mathbf{A}^{\mathrm{T}} + \mathbf{A}R_{ss}\mathbf{A}^{\mathrm{T}})\right] \\ &\quad -\frac{1}{2}\mathrm{tr}\left[\boldsymbol{\Lambda}^{-1}(\mathrm{vec}(\mathbf{A}) - \boldsymbol{\mu}_A)(\mathrm{vec}(\mathbf{A}) - \boldsymbol{\mu}_A)^{\mathrm{T}}\right] \end{aligned} \tag{20}$$

The updating rule for the mixing matrix $\mathbf{A}$ can therefore be obtained by taking the derivative of $\hat{J}_A$ with respect to $\mathbf{A}$, and setting it to zero, which gives

$$\begin{aligned} \mathrm{vec}(\hat{\mathbf{A}}) = &\left[T\mathbf{R}_{ss} \otimes \mathbf{R}_w^{-1} + \boldsymbol{\Lambda}^{-1}\right]^{-1} \\ &\left[\mathrm{vec}\left(T\mathbf{R}_w^{-1}\mathbf{R}_{xs}\right) + \boldsymbol{\Lambda}^{-1}\boldsymbol{\mu}_A\right] \end{aligned} \tag{21}$$

where $\otimes$ denotes the Kronecker product.

Similarly, by incorporating the prior information for the noise covariance matrix $\mathbf{R}_w$ denoted by $J^g_{R_w}$, which is related to its prior distribution as defined in (7), the MAP auxiliary function for the noise covariance $\mathbf{R}_w$ can be written as

$$\begin{aligned}\hat{J}_{R_w} &= J_1 + J^g_{R_w} \\ &= -\frac{T}{2}\log|\mathbf{R}_w| \\ &\quad -\frac{T}{2}\mathrm{tr}\left[\mathbf{R}_w^{-1}\left(\mathbf{R}_{xx} - \mathbf{A}R_{sx} - \mathbf{R}_{sx}^{\mathrm{T}}\mathbf{A}^{\mathrm{T}} + \mathbf{A}R_{ss}\mathbf{A}^{\mathrm{T}}\right)\right] \\ &\quad -\frac{v_R - Q - 1}{2}\log|\mathbf{R}_w| - \frac{1}{2}\mathrm{tr}\left(\Sigma_w\mathbf{R}_w^{-1}\right) \qquad (22)\end{aligned}$$

The updating rule for the noise covariance matrix $\mathbf{R}_w$ can be similarly obtained by taking the derivative of $\hat{J}_{R_w}$ with respect to $\mathbf{R}_w$, and setting it to zero, which gives

$$\begin{aligned}\hat{\mathbf{R}}_w = &\frac{1}{T + v_R - Q - 1} \\ &\left[T\left(\mathbf{R}_{xx} - \hat{\mathbf{A}}\mathbf{R}_{sx} - \mathbf{R}_{sx}^{\mathrm{T}}(\hat{\mathbf{A}})^{\mathrm{T}} + \hat{\mathbf{A}}\mathbf{R}_{ss}(\hat{\mathbf{A}})^{\mathrm{T}}\right) + \Sigma_w\right] \qquad (23)\end{aligned}$$

The updating rules for $\mathbf{A}$ and $\mathbf{R}_w$ involve the calculation of $\mathbf{R}_{ss}$ and $\mathbf{R}_{sx}$. Using the posterior distribution $f\left(\mathbf{s}(t)|\mathbf{x}, \mathbf{A}^g, \mathbf{R}^g_w, \Theta^g\right)$ as shown in (14) and (15), it is easy to obtain the conditional expectations $\mathrm{E}[\mathbf{s}(t)|\mathbf{x}, \mathbf{A}^g, \mathbf{R}^g_w, \Theta^g]$ and $\mathrm{E}[\mathbf{s}^{\mathrm{T}}(t)\mathbf{s}(t)|\mathbf{x}, \mathbf{A}^g, \mathbf{R}^g_w, \Theta^g]$.

5.2.2 Estimation Formula for the GMM Parameters

The parameters Θ of the GMM can be updated in a similar way to that for the mixing matrix and noise covariance matrix. Using (14) and (15), the auxiliary function J_2 in (17) can be rewritten as

$$\begin{aligned}J_2 = \sum_{t=1}^{T}\sum_{m=1}^{M}\int_{\mathbf{s}} &\tilde{\omega}^g_{mt}\mathcal{N}\left(\mathbf{s}(t); \tilde{\boldsymbol{\mu}}^g_{mt}, \tilde{\mathbf{C}}^g_{mt}\right) \\ &\log\omega_m f\left(\mathbf{s}(t)|y(t) = m, \Theta\right)\mathrm{d}\mathbf{s} \qquad (24)\end{aligned}$$

with the prior density as depicted in (8), (9) and (10), then the prior information for the source signals can be denoted as

$$\begin{aligned}J^g_S = \sum_{m=1}^{M}&(\eta_m - 1)\log\omega_m - \left((v_m - P)/2\right)\log|\mathbf{C}_m| \\ &-\frac{\tau_m}{2}\mathrm{tr}\left[\mathbf{C}_m^{-1}\left(\boldsymbol{\mu}_m - \mathbf{u}_m\right)\left(\boldsymbol{\mu}_m - \mathbf{u}_m\right)^{\mathrm{T}}\right] - \frac{1}{2}\mathrm{tr}\left(\Sigma_m\mathbf{C}_m^{-1}\right) \qquad (25)\end{aligned}$$

Hence, the MAP auxiliary function $\hat{J}_S$ for the GMM parameters can be written as $\hat{J}_S = J_2 + J^g_S$. To maximize this expression, we can maximize the terms containing the weighting coefficient parameter ω_m, and the term containing the mean vector $\boldsymbol{\mu}_m$, and the covariance matrix $\mathbf{C}_m$, $m = 1, \ldots, M$ separately since they are independent from each other.

Note that $\int_{\mathbf{s}} \mathscr{N}(\mathbf{s}(t); \tilde{\boldsymbol{\mu}}_m^g, \tilde{\mathbf{C}}_m^g)\mathrm{d}\mathbf{s} = 1$, and hence, the part of the auxiliary function $\hat{J}_S$ related to parameter ω_m can be simplified as

$$\hat{J}_S^{(1)} = \sum_{t=1}^{T}\sum_{m=1}^{M} \tilde{\omega}_{mt}^g \log \omega_m + \sum_{m=1}^{M} (\eta_m - 1) \log \omega_m \tag{26}$$

Adding the Lagrange multiplier λ, using the constraints that $\sum_{m=1}^{M} \omega_m = 1$, and setting the derivative of $\hat{J}_S^{(1)}$ with respect to ω_m equal to zero, one obtains

$$\begin{aligned} &\frac{\partial}{\partial \omega_m}\left[\left(\sum_{m=1}^{M}\sum_{t=1}^{T} \tilde{\omega}_{mt}^g \log \omega_m\right)\right. \\ &\quad \left. + \sum_{m=1}^{M} (\eta_m - 1) \log \omega_m + \lambda \left(\sum_{m=1}^{M} \omega_m - 1\right)\right] = 0 \end{aligned} \tag{27}$$

Summing both sides over m, we can get $\lambda = -\left(\sum_{m=1}^{M} \eta_m - M + T\right)$ resulting in

$$\hat{\omega}_m = \frac{\eta_m - 1 + \sum_{t=1}^{T} \tilde{\omega}_{mt}^g}{\sum_{m=1}^{M} \eta_m - M + T} \tag{28}$$

On the other hand, the part of the auxiliary function $\hat{J}_S$ related to parameters $\boldsymbol{\mu}_m$ and $\mathbf{C}_m$ can be written as

$$\begin{aligned} \hat{J}_S^{(2)} = &\sum_{t=1}^{T} \int_{\mathbf{s}} \tilde{\omega}_m^g \mathscr{N}\left(\mathbf{s}(t); \tilde{\boldsymbol{\mu}}_m^g, \tilde{\mathbf{C}}_m^g\right) \log f\left(\mathbf{s}(t)|y(t) = m, \Theta\right) \mathrm{d}\mathbf{s} \\ &+ \left((\upsilon_m - P)/2\right) \log \left|\mathbf{C}_m^{-1}\right| \\ &- \frac{\tau_m}{2} \mathrm{tr}\left[\mathbf{C}_m^{-1}(\boldsymbol{\mu}_m - \mathbf{u}_m)(\boldsymbol{\mu}_m - \mathbf{u}_m)^{\mathrm{T}}\right] - \frac{1}{2}\mathrm{tr}\left[\Sigma_m \mathbf{C}_m^{-1}\right] \end{aligned} \tag{29}$$

The updating rule for $\boldsymbol{\mu}_m$ and $\mathbf{C}_m$ can therefore be obtained by taking the derivative of $\hat{J}_S^{(2)}$ with respect to $\boldsymbol{\mu}_m$ and $\mathbf{C}_m$, respectively, and setting them to zero. That is $\partial \hat{J}_S^{(2)} \big/ \partial \boldsymbol{\mu}_m = 0$ and $\partial \hat{J}_S^{(2)} \big/ \partial \mathbf{C}_m = 0$.

For notational simplicity, define

$$\begin{aligned} \boldsymbol{\Gamma}_m &= \mathrm{E}\left[\mathbf{s}(t)\mathbf{s}^{\mathrm{T}}(t)|y(t) = m, \mathbf{x}, \mathbf{A}^g, \mathbf{R}_w^g, \Theta^g\right] \\ &= \tilde{\mathbf{C}}_{mt}^g + \tilde{\boldsymbol{\mu}}_{mt}^g \left(\tilde{\boldsymbol{\mu}}_{mt}^g\right)^{\mathrm{T}} \end{aligned} \tag{30}$$

Then, one obtains

$$\frac{\partial}{\partial \boldsymbol{\mu}_m}\left(\sum_{t=1}^{T}\int_{\mathbf{s}} \tilde{\omega}_{mt}^{g}\mathcal{N}(\mathbf{s}(t);\tilde{\boldsymbol{\mu}}_{mt}^{g},\tilde{\mathbf{C}}_{mt}^{g})\log f(\mathbf{s}(t)|y(t)=m,\Theta)\mathrm{d}\mathbf{s}\right)$$
$$=\sum_{t=1}^{T}\tilde{\omega}_{mt}^{g}\mathbf{C}_m^{-1}\left(\tilde{\boldsymbol{\mu}}_{mt}^{g}-\boldsymbol{\mu}_m\right) \tag{31}$$

and

$$\frac{\partial}{\partial \mathbf{C}_m}\left(\sum_{t=1}^{T}\int_{\mathbf{s}} \tilde{\omega}_{mt}^{g}\mathcal{N}(\mathbf{s}(t);\tilde{\boldsymbol{\mu}}_{mt}^{g},\tilde{\mathbf{C}}_{mt}^{g})\right.$$
$$\left.\log f(\mathbf{s}(t)|y(t)=m,\Theta)\mathrm{d}\mathbf{s}\right)$$
$$=\frac{1}{2}\sum_{t=1}^{T}\tilde{\omega}_{mt}^{g}\left[-\mathbf{C}_m+\left(\boldsymbol{\Gamma}_m-\tilde{\boldsymbol{\mu}}_{mt}^{g}\boldsymbol{\mu}_m^{\mathrm{T}}-\boldsymbol{\mu}_m(\tilde{\boldsymbol{\mu}}_{mt}^{g})^{\mathrm{T}}+\boldsymbol{\mu}_m\boldsymbol{\mu}_m^{\mathrm{T}}\right)\right] \tag{32}$$

On the other hand, notice that

$$\frac{\partial}{\partial \boldsymbol{\mu}_m}\left(((\upsilon_m-P)/2)\log\left|\mathbf{C}_m^{-1}\right|\right.$$
$$\left.-\frac{\tau_m}{2}\mathrm{tr}\left[\mathbf{C}_m^{-1}(\boldsymbol{\mu}_m-\mathbf{u}_m)(\boldsymbol{\mu}_m-\mathbf{u}_m)^{\mathrm{T}}\right]-\frac{1}{2}\mathrm{tr}(\Sigma_m\mathbf{C}_m^{-1})\right)$$
$$=-\tau_m\mathbf{C}_m^{-1}(\boldsymbol{\mu}_m-\mathbf{u}_m) \tag{33}$$

$$\frac{\partial}{\partial \mathbf{C}_m}\left(((\upsilon_m-P)/2)\log\left|\mathbf{C}_m^{-1}\right|\right.$$
$$\left.-\frac{\tau_m}{2}\mathrm{tr}\left[\mathbf{C}_m^{-1}(\boldsymbol{\mu}_m-\mathbf{u}_m)(\boldsymbol{\mu}_m-\mathbf{u}_m)^{\mathrm{T}}\right]-\frac{1}{2}\mathrm{tr}(\Sigma_m\mathbf{C}_m^{-1})\right)$$
$$=-\frac{(\upsilon_m-P)}{2}\mathbf{C}_m+\frac{\tau_m}{2}(\boldsymbol{\mu}_m-\mathbf{u}_m)(\boldsymbol{\mu}_m-\mathbf{u}_m)^{\mathrm{T}}+\frac{1}{2}\Sigma_m \tag{34}$$

Combining the terms (31) and (33), the updating formula for $\boldsymbol{\mu}_m$ can be easily obtained as

$$\hat{\boldsymbol{\mu}}_m=\frac{\tau_m\mathbf{u}_m+\sum_{t=1}^{T}\tilde{\omega}_{mt}^{g}\tilde{\boldsymbol{\mu}}_{mt}^{g}}{\tau_m+\sum_{t=1}^{T}\tilde{\omega}_{mt}^{g}} \tag{35}$$

and the updating formula for $\mathbf{C}_m$ can be similarly obtained by combining the terms (32) and (34)

$$\hat{\mathbf{C}}_m=\frac{\Sigma_m+\tau_m(\hat{\boldsymbol{\mu}}_m-\mathbf{u}_m)(\hat{\boldsymbol{\mu}}_m-\mathbf{u}_m)^{\mathrm{T}}}{\upsilon_m-P+\sum_{t=1}^{T}\tilde{\omega}_{mt}^{g}}$$
$$+\frac{\sum_{t=1}^{T}\tilde{\omega}_{mt}^{g}\left(\boldsymbol{\Gamma}_m-\tilde{\boldsymbol{\mu}}_{mt}^{g}(\hat{\boldsymbol{\mu}}_m)^{\mathrm{T}}-\hat{\boldsymbol{\mu}}_m(\tilde{\boldsymbol{\mu}}_{mt}^{g})^{\mathrm{T}}+\hat{\boldsymbol{\mu}}_m(\hat{\boldsymbol{\mu}}_m)^{\mathrm{T}}\right)}{\upsilon_m-P+\sum_{t=1}^{T}\tilde{\omega}_{mt}^{g}} \tag{36}$$

6 Practical Implementation and Algorithm Analysis

In this section, we discuss some practical implementation issues of the proposed algorithm and also offer an empirical analysis of its convergence and computational complexity.

6.1 Summary of the Algorithm

The proposed GMM-EM algorithm can be implemented as follows:

1. ***Initialization*** Initialize the mixing matrix $\mathbf{A}^{(0)}$, noise variance matrix $\mathbf{R}_w^{(0)}$ and model parameters $\Theta^{(0)}$ according to the initialization scheme described in Sect. 6.2, and set the EM iteration index $i = 0$.
2. ***EM iterations*** Repeat the E-step and M-step until convergence.
   (a) ***E-step*** Calculate $f(\mathbf{s}(t)|\mathbf{x}(t), \mathbf{A}^{(i)}, \mathbf{R}_w^{(i)}, \Theta^{(i)})$ according to (14)–(15), and calculate $\mathbf{R}_{ss}$ and $\mathbf{R}_{sx}$ according to (19).
   (b) ***M-step*** Calculate the mixing matrix $\mathbf{A}^{(i+1)}$ and noise covariance matrix $\mathbf{R}_w^{(i+1)}$ according to (21) and (23), respectively. Calculate the weight $\omega_m^{(i+1)}$, mean vector $\boldsymbol{\mu}_m^{(i+1)}$ and covariance matrix $\mathbf{C}_m^{(i+1)}$ according to (28), (35) and (36), respectively.
3. ***MAP source estimation*** Let i_0 be the number of iterations required before the convergence of the algorithm, the posterior mean estimate $\hat{\mathbf{s}}_{MAP}$ is approximated by the empirical mean of the sequence $\mathbf{s}_{i>i_0}$.

6.2 Parameter Initialization

It is well known that the EM optimization strategy is sensitive to the initial setups of the parameters. The likelihood function may converge to a local maximum instead of the global maximum due to the use of the bootstrap process in the iterations. The initial values for the parameters therefore become important for the convergence of the EM algorithm, not only in terms of minimizing the number of iterations required for the algorithm to converge to a local maximum, but also for it to find a "good" solution [35]. Therefore, the parameters are given proper prior densities, i.e., conjugate prior densities, in order to incorporate the prior information as discussed in Sect. 4. A reasonable choice for the initial estimates is the mode of the prior density. For the mixing matrix and the noise covariance matrix, the initial values can be set as

$$\begin{cases} \mathrm{vec}(\mathbf{A})^{(0)} = \boldsymbol{\mu}_A \\ \mathbf{R}_w^{(0)} = (v_R - Q - 1)\Sigma_w^{-1} \end{cases} \tag{37}$$

The mean value of the mixing matrix $\boldsymbol{\mu}_A$ can be estimated by the blind identification methods such as those presented in [10,15,22,24,26,27,36]. Although, in such cases, the source signals can not be recovered by multiplying the observed signals with the inverse/pseudo inverse of the mixing matrix, the hyper-parameter $\boldsymbol{\mu}_A$ can be determined by the estimate of the mixing matrix. In our implementation in Sect. 7, the LEMACAF-4 method[2] in [15] has been used for estimating the initial value of the mixing matrix.

Similarly, the initial estimates for the GMM model parameters of the sources are taken as

$$\begin{cases} \omega_m^{(0)} = (\eta_m - 1) \Big/ \left(\sum_{m=1}^{M} \eta_m - M \right) \\ \boldsymbol{\mu}_m^{(0)} = \mathbf{u}_m \\ \mathbf{C}_m^{(0)} = (v_m - P) \Sigma_m^{-1} \end{cases} \tag{38}$$

It has been shown that the joint PDF of the observed signals can also be modeled by GMM when the joint PDF of the source signals is modeled by GMM [40] (due to space limitation, the detail is omitted here). As a result, the weighting coefficients, the mean vectors and covariance matrices of the observation-based GMM model can be learned from the observed signals. Therefore, according to the relationship between the weighting coefficients, the mean vectors and covariance matrices of the observation-based GMM model and their counterparts of the source-based GMM model, the hyper-parameters η_m, $\mathbf{u}_m$ and Σ_m can be determined by the estimate of the mixing matrix and the estimates of the observation-based GMM model parameters jointly.

6.3 Convergence Analysis

In essence, the proposed GMM-EM separating algorithm is a gradient-based bootstrap method for optimizing the log-likelihood function. There are already some works that have investigated this issue [37,39]. For example, Xu et al. [39] have established the linkage of the EM algorithm with the gradient-based approaches for the ML learning based on GMM, and shown that the EM parameters are iterated in terms of the gradient obtained by a positive definite projection matrix. This result was extended to the more general block coordinate descent (BCD) method [12,28], in which a single block of variables is optimized at each iteration.

In Section V, to update each variable of $\mathbf{A}$, $\mathbf{R}_w$, ω_m, $\boldsymbol{\mu}_m$ and $\mathbf{C}_m$, the employed method is to simply set the first derivative of the complete-data log-likelihood with respect to each variable to be zero and solve the corresponding equation sequentially. Hence, if the Hessian matrices of complete-data log-likelihood with respect to these variables are non-positive, then it is safe to state that the subproblem with respect to each variable is convex. Take the mixing matrix as an example. By converting the matrix $\mathbf{A}$ into vector form $vec(\mathbf{A})$, the Hessian matrix of $\hat{J}_A$ with respect to $vec(\mathbf{A})$ can be easily obtained, and written as $\mathbf{H}_A = -(T\mathbf{R}_{ss} \otimes \mathbf{R}_w^{-1} + \boldsymbol{\Lambda}^{-1})$. Since $\mathbf{R}_w$ and $\boldsymbol{\Lambda}$ are positively defined, it is obvious that $\mathbf{H}_A$ is non-positive. The Hessian matrix of $\hat{J}_{R_w}$ with respect to $\mathbf{R}_w$ can be obtained in a similar way, and it is also non-positive. That is, for each iteration of the GMM-EM algorithm given in (21, 23), the search direction of the parameters has a positive projection on the gradient of its corresponding MAP auxiliary function.

[2] Matlab codes can be found at: http://www.i3s.unice.fr/pcomon/TensorPackage.html.

We further discuss the GMM parameters ω_m, $\boldsymbol{\mu}_m$ and $\mathbf{C}_m$. If each mixture component is assumed to be non-degenerate [39], i.e., $\hat{\omega}_m > 0$, then $\tilde{\omega}^g_{m1}, \ldots, \tilde{\omega}^g_{mT}$ is a sequence of T i.i.d. random variables with a non-degenerate distribution and $\lim_{T\to\infty} \sum_{t=1}^{T} \tilde{\omega}_{mt} = \infty$ with probability one. It follows that the Hessian matrix $\mathbf{H}_{\omega_m} = -(\eta_m - 1 + \sum_{t=1}^{T} \tilde{\omega}^g_{mt})$ is non-positive with probability one when $T \to \infty$. Applying the same reasoning, we can see that the GMM-EM estimation formulas for $\hat{\boldsymbol{\mu}}_m$ and $\hat{\mathbf{C}}_m$ are asymptotically similar in terms of the MAP approach [29,34]. Therefore, as long as the initial estimates of $\mathbf{A}^{(0)}$, $\mathbf{R}_w^{(0)}$ and $\Theta^{(0)}$ remain unchanged, the EM algorithm will converge to the same estimates with probability one when $T \to \infty$.

Finally, it should be pointed out that the EM algorithm may converge to a local maximum instead of the global maximum when the number of parameters is large and/or the parameters of the algorithm are inappropriately initialized. This is a general limitation associated with the gradient-based bootstrap-like optimization algorithms. The reason is that it is often trapped to the neighborhood of a local optimizer if the number of parameters is large and/or the parameters of the EM algorithm are initialized such that the solution is far from the global optimizer. Hence, the initialization scheme discussed in Sect. 6.2 is vital to ensure the convergence of the proposed GMM-EM algorithm.

6.4 Computational Complexity Analysis

The computational load of the proposed GMM-EM algorithm is dominated by the E-step and M-step. In each iteration of the E-step, it is required to:

– calculate the posterior probability $f\left(\mathbf{s}(t)|\mathbf{x}, \mathbf{A}^{(i)}, \mathbf{R}_w^{(i)}, \Theta^{(i)}\right)$ with (14) and (15) which requires $\mathscr{O}(Q(P+Q)(P^2+Q^2))$ multiplications per observation vector.
– calculate the statistics $\mathbf{R}_{xx}$, $\mathbf{R}_{sx}$ and $\mathbf{R}_{ss}$ with (19) which requires $\mathscr{O}((P+Q)^2)$ multiplications per observation vector.

In each iteration of the M-step, it is required to:

– update the mixing matrix $\mathbf{A}$ using (21) and the noise covariance matrix $\mathbf{R}_w$ using (23) which require $\mathscr{O}((PQ)^3 + (PQ)^2)$ and $\mathscr{O}(PQ(P+Q))$ multiplications, respectively.
– update the weighting coefficients $\omega_m^{(i+1)}$, mean vector $\boldsymbol{\mu}_m^{(i+1)}$ and covariance matrix $\mathbf{C}_m^{(i+1)}$ according to (28), (35) and (36) which amount to $\mathscr{O}(MP^2T)$ multiplications.

From the above analysis, we can see that the computational complexity of the proposed GMM-EM algorithm depends closely on the number of sources and sample size considered in the model. Theoretically, the proposed GMM-EM algorithm for GMM parameter estimation of source signals would become increasingly intractable and computationally unaffordable as the number of sources increases. This is because the number of Gaussians for modeling the source vector grows exponentially with the number of sources. For example, assuming that the number of sources is $P = 10$, and the PDF of each source is modeled by the GMM with $l_p = 3$ Gaussians, then it is straightforward to derive that $M = 3^{10}$ Gaussians are required to model the source vector. However, it has been shown that the determined GMM order in high dimensions is always much smaller than the theoretical number of Gaussians [40]. The main reason is that the distribution of the sensors becomes more Gaussian while the number of sources increases. Hence, it enables the applicability of the proposed GMM-EM method also for a large number of sources.

Note that the computational complexity of an iteration of the NCA algorithm [21] is $\mathscr{O}(T^3)$ when the size of the signal is assumed to be much larger than other parameters. In contrast to the NCA algorithm, the proposed GMM-EM algorithm apparently has advantage in terms of computational complexity. This is because the computational complexity of an iteration of the proposed GMM-EM algorithm is $\mathscr{O}(T)$ under the same situation.

7 Simulations and Analysis

In this section, the separation performance of the proposed GMM-EM algorithm is evaluated in terms of similarity score, and compared with that of the NCA [21] and FastICA [41] algorithms. Calculation of the similarity score is detailed in “Appendix 3.” Note that the FastICA algorithm cannot be implemented in the underdetermined case, and hence, we only compare the proposed GMM-EM with the NCA in such case.

The section is organized as follows. First, the separation performances of the compared algorithms are evaluated based on synthetic data in terms of similarity score versus the signal-to-noise (SNR) level within the mixtures, sample size, and the number of sources in determined mixtures. Second, these performance aspects are also investigated for underdetermined mixtures. Finally, the performances of the compared algorithms are evaluated for separating mixtures of real speech signals.

The compared algorithms were operated under the following overall settings: (1) The number of EM iterations used in the proposed GMM-EM algorithm was set to 100; (2) the separation performance of the NCA algorithm was evaluated with 100 iterations.

7.1 Synthetic Data

7.1.1 Separation Performance as a Function of SNR in Determined Mixtures

The following experiment compares the separation performances of the tested algorithms for determined mixtures in the presence of additive Gaussian noise. Each source signal is synthesized by the following GMM PDF $f_s = 0.5\mathscr{N}(s; 1, 0.4) + 0.5\mathscr{N}(s; -1, 0.4)$. For each SNR, 100 sets of two-dimensional independent source signals, containing $T = 1000$ samples, are synthesized and mixed by a random 2×2 mixing matrix, with its elements randomly drawn in the range $[-1, 1]$. Additive Gaussian noise is added in the mixing process, and the SNR of the observations ranges from 0 to 30 dB.

Figure 2 depicts the average similarity score between the original sources and recovered sources of the tested algorithms versus the SNR. For the low SNRs (e.g., lower than 20 dB), one can observe that the proposed GMM-EM algorithm offers the

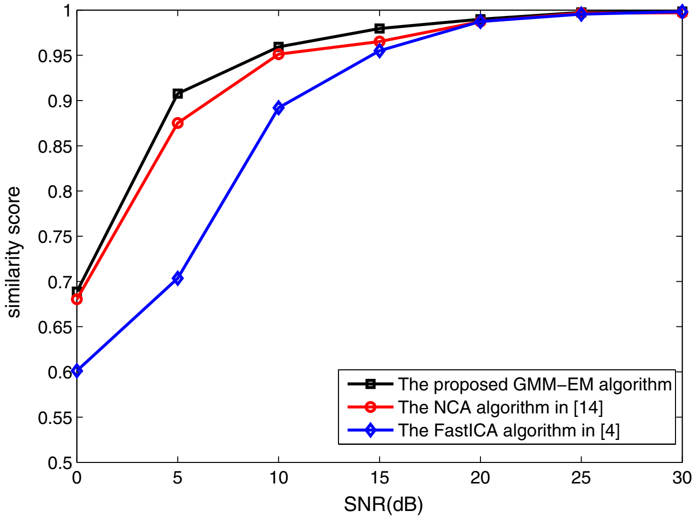

Fig. 2 Average similarity score of the tested algorithms versus the SNR in the determined case

best performance, followed by the NCA and FastICA algorithms, respectively. The advantage of the proposed algorithm tends to disappear when the SNR is greater than 20 dB, and in this case, the level of noise is pretty low and hence could be ignored in practice. Furthermore, the performance of the NCA algorithm is close to that of the proposed GMM-EM algorithm. This is because the noise component is also considered in the NCA algorithm. For the NCA algorithm, it mainly depends on whether the null space of different signals is orthogonal. For the proposed GMM-EM algorithm, it is robust to noise due to the fact that the noise component has been taken into account in the model with its covariance jointly estimated in the EM process.

7.1.2 Separation Performance as a Function of Sample Size in Determined Mixtures

The following experiment compares the separation performances of the tested algorithms as a function of the sample size T for determined mixtures. Each signal is synthesized by the same GMM used in the first experiment shown above. For each $T \in \{100, 200, 400, 600, 1000, 2000, 4000\}$, 100 sets of two-dimensional independent sources are synthesized and mixed by a random 2×2 mixing matrix, with its elements randomly drawn in the range $[-1, 1]$. Additive Gaussian noise is added in the mixing process, and the SNR of the observations is 10 dB.

Figure 3 depicts the average similarity score of the tested algorithm versus the sample size T. One can observe that as T increases from 100 to 4000, the separation performances of the tested algorithms improve, and the proposed GMM-EM and NCA algorithms outperform the FastICA algorithm. Moreover, the performance of the proposed GMM-EM algorithm will achieve a steady state when the sample size is larger than 2000. The reason is that the statistics can be estimated with a high precision by using a large number of samples, and the estimation precision will converge to a

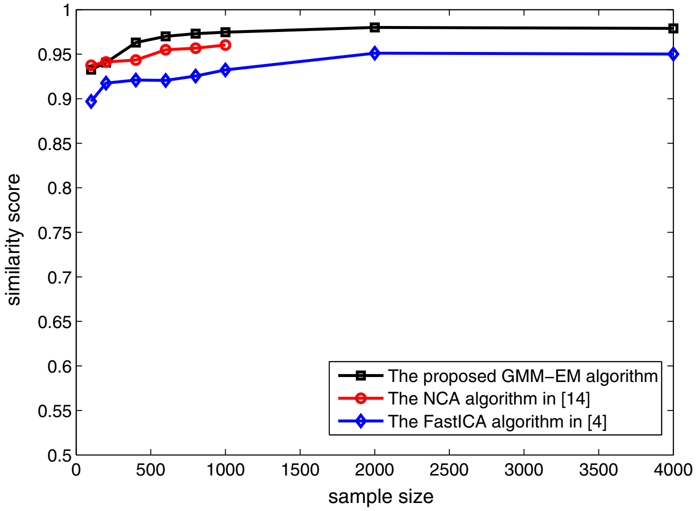

Fig. 3 Average similarity score of the tested algorithms versus the sample size in the determined case

steady state. However, as pointed out in [21], the computational complexity of the NCA algorithm is proportional to $\mathcal{O}(T^3)$, and hence, it becomes computationally prohibitive when the sample size is large than 1000, and no results are given beyond this point.

7.1.3 Separation Performance as a Function of Dimension in Determined Mixtures

The following experiment compares the separation performances of the proposed algorithm as a function of SNR for different number of sources in determined mixtures. Each signal is synthesized by the same GMM as used in the first experiment. For each $P \in \{2, 3, 4, 5\}$, 100 sets of P independent source signals, containing $T = 1000$ samples, are synthesized and mixed by a random $P \times P$ mixing matrix, with its elements randomly drawn in the range $[-1, 1]$. Additive Gaussian noise is added in the mixing process, and the SNR of the observations ranges from 0 to 30 dB.

Figure 4 depicts the average similarity score of the proposed GMM-EM algorithm versus the SNR when the number of sources is varied from 2 to 5. One can observe that the performance of the proposed GMM-EM algorithm deteriorates as the number of sources increases.

7.1.4 Separation Performance as a Function of SNR in Underdetermined Mixture

The following experiment compares the separation performances of the tested algorithms for underdetermined mixtures in the presence of additive Gaussian noise. Each signal is synthesized by the same GMM used in the first experiment. For each SNR, 100 sets of three-dimensional independent source signals, containing $T = 1000$ samples, are synthesized and mixed by a random 2×3 mixing matrix, with its elements ran-

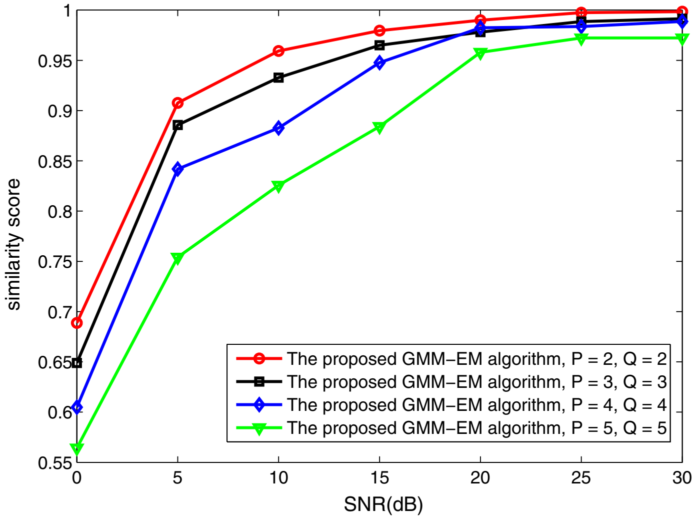

Fig. 4 Average similarity score of the proposed algorithm versus SNR for a varying number of sources in the determined case

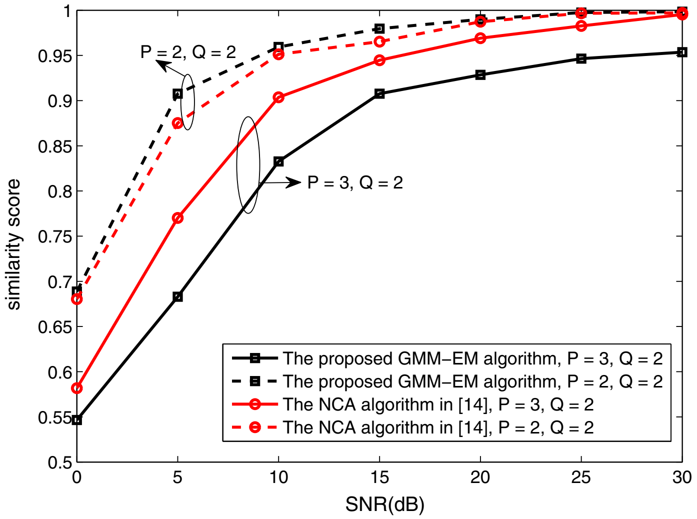

Fig. 5 Average similarity scores of the tested algorithms versus the SNR in the underdetermined case

domly drawn from the range [−1, 1]. Additive Gaussian noise is added in the mixing process, and the SNR of the observations ranges from 0 to 30 dB.

The average similarity scores of the tested algorithms versus the SNR in the underdetermined case are shown in Fig. 5. Due to the mixing matrix and sources are estimated jointly, rather than separately (e.g., based on the inverse of the mixing matrix), the proposed GMM-EM algorithm can also work in the underdetermined case. From Fig. 5, one can also observe that the separation performances of the tested algorithms

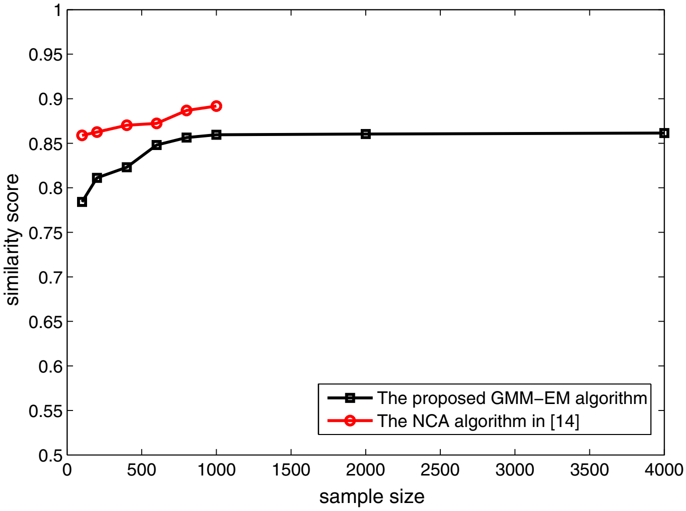

Fig. 6 Average similarity score of the tested algorithms versus the sample size in the underdetermined case

improve with the increase of SNR. However, it can also be observed that the separation performance in the underdetermined case deteriorates as compared with the performance in determined case. This is because the information loss caused by the lack of sensors in the underdetermined case. Furthermore, it should be pointed out that the NCA algorithm outperforms the proposed GMM-EM algorithm in such an underdetermined case. The main reason is that the NCA algorithm mainly depends on whether the null spaces of different sources are orthogonal, regardless whether the mixture is determined or underdetermined.

7.1.5 Separation Performance as a Function of Sample Size in Underdetermined Mixtures

The following experiment compares the separation performances of the tested algorithms as a function of sample size T for underdetermined mixtures. Each signal is synthesized by the same GMM as used in the first experiment. For each $T \in \{100, 200, 400, 600, 1000, 2000, 4000\}$, 100 sets of three-dimensional independent source signals are synthesized and mixed by a 2×3 mixing matrix, with its elements randomly drawn in the range $[-1, 1]$. Additive Gaussian noise is added in the mixing process, and the SNR of the observations is 10 dB.

Figure 6 depicts the average similarity score of the tested algorithm versus the sample size T, where the performance of the NCA algorithm is again shown for up to 1000 samples. One can observe that as T increases from 100 to 4000, the separation performances of the tested algorithms improve.

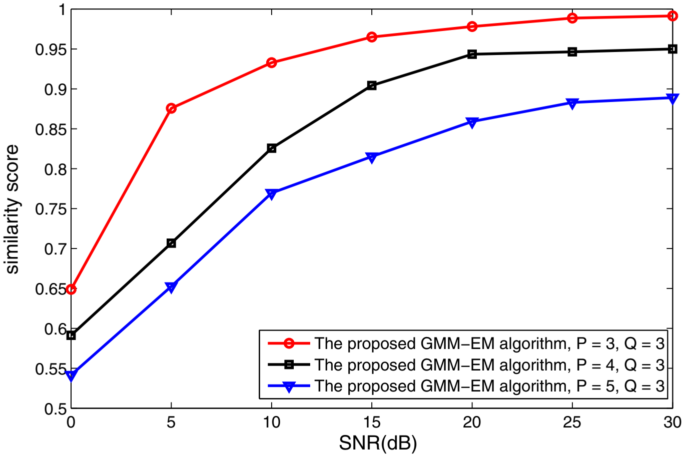

Fig. 7 Average similarity scores of the proposed algorithm versus the SNR for a varying number of sources in the underdetermined case

7.1.6 Separation Performance as a Function of Dimension in Underdetermined Mixtures

The following experiment compares the separation performances of the tested algorithms as a function of the number of sources in underdetermined mixtures. Each signal is synthesized by the same GMM used in the first experiment. For each $P \in \{3, 4, 5\}$, 100 sets of P independent source signals, containing $T = 1000$ samples, are synthesized and mixed by a random $3 \times P$ mixing matrix, with its elements randomly drawn in the range $[-1, 1]$. Additive Gaussian noise is added in the mixing process, and the SNR of the observations ranges from 0 to 30 dB.

Figure 7 depicts the average similarity score of the proposed GMM-EM algorithm versus the SNR when the number of observations is fixed and the number of sources is varied. One can observe that the performance of the proposed GMM-EM algorithm deteriorates as the number of sources increases.

7.2 Real Data

The following experiments compare the performances of the tested algorithms, in terms of similarity score, in separating different mixture combinations of three 0.5-s-long speech signals,[3] sampled at 8000 Hz and recorded with 8 bits per sample. Figure 8 shows the waveform of the sources and their histograms of the amplitude distributions. On the other hand, the PDF of each source signal is modeled by GMM of order 3, where the order of the GMM is determined according to the Bayesian information criterion (BIC) [16]. The BIC, based on the likelihood function and a penalty term introduced for the number of parameters in the model, is a well-known criterion for model selection

[3] Available at: http://www.kecl.ntt.co.jp/icl/signal/sawada/webdemo/bssdemo.html.

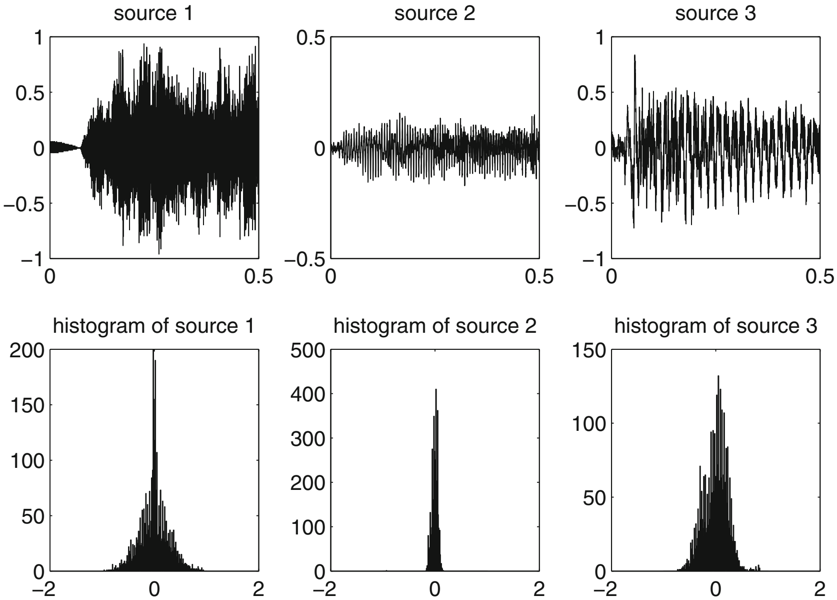

Fig. 8 Speech signals and their respective histograms of amplitude distributions

among a finite set of models. By calculating the BIC values for all possible models, the candidate model is chosen as the one corresponding to the minimum value of the BIC. The density estimations for the sources are shown in Fig. 9. We can observe that each distribution can be well approximated with 3 Gaussian components and similar to its counterpart in Fig. 8.

Two experiments are used to investigate the separation performance of the proposed GMM-EM algorithm when the sources are real speech signals.

- In the first experiment, two speech signals are artificially mixed by a 2×2 mixing matrix, whose elements are randomly generated from a uniform distribution over the interval $[-1, 1]$. Additive Gaussian noise is added in the mixing process, and the SNR of the observations ranges from 0 to 30 dB. In [40], it is shown that the joint PDF of the observed signals can also be modeled by GMM when the joint PDF of the source signals is modeled by GMM. Hence, the order of the GMM of the sources equals to that of the observed signals. As a result, the order of the GMM can be determined according to the BIC based on the observed signals. Here, the optimal GMM order determined by the BIC criterion for the GMM-EM separating algorithm is 9. The NCA and FastICA algorithms are implemented as baseline algorithms. 100 Monte Carlo experiments are run.
- In the second experiment, the underdetermined case of $P = 3$ sources and $Q = 2$ observations is considered. The source signals are the same speech signals used in the noisy determined case. The sources are artificially mixed by a 2×3 mixing matrix, whose elements are randomly generated from a uniform distribution over the interval $[-1, 1]$. The SNR of the observations is ranged from 0 to 30 dB. The optimal GMM order determined by the BIC criterion for the GMM-EM algorithm

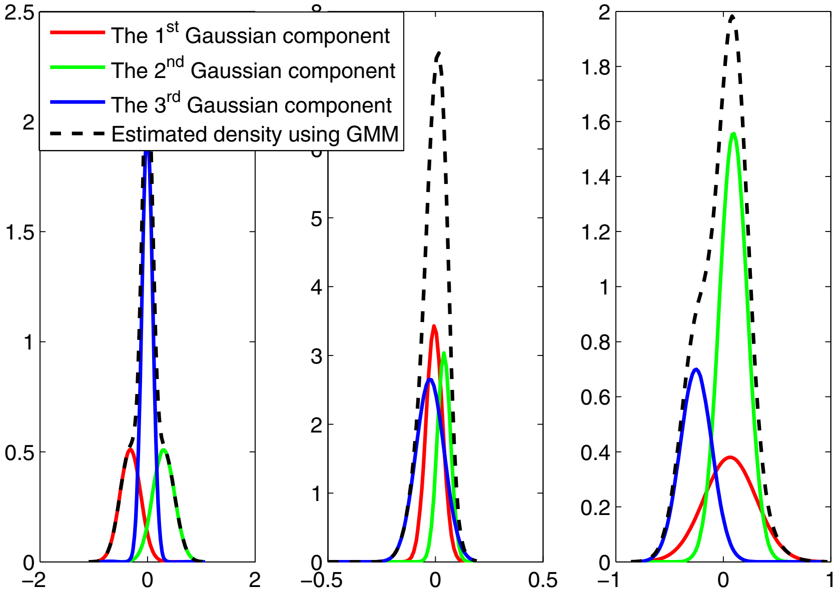

Fig. 9 GMM for probability density estimations of speech signals

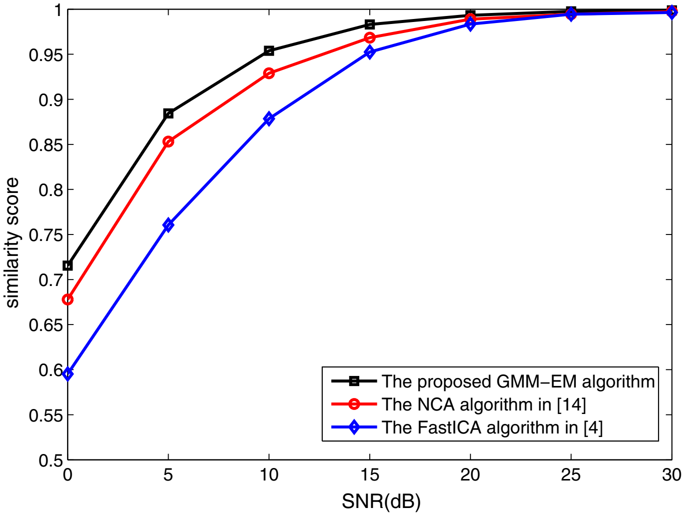

Fig. 10 Average similarity score of the tested algorithms versus the SNR in the determined mixtures with real speech signals

is 27. The NCA algorithm is implemented as the baseline algorithm. 100 Monte Carlo experiments are run.

The average separation performance of the tested algorithms in terms of the similarity score versus the SNR in determined mixtures is shown in Fig. 10. The separation performances of the tested algorithms for underdetermined mixtures are shown in

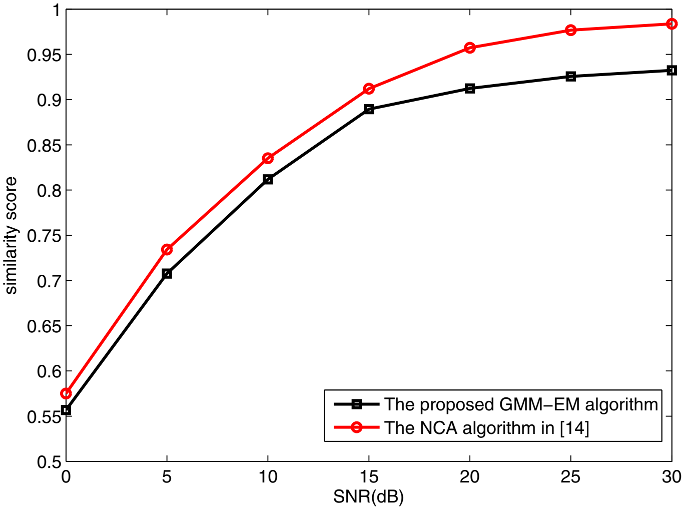

Fig. 11 Average similarity score of the tested algorithms versus the SNR in the underdetermined mixtures when the sources are speech signals

Fig. 11. One can observe similar performance patterns to those in Figs. 2 and 5. More specifically, the proposed GMM-EM algorithm seems to give almost identical similarity scores as for synthetic data.

8 Conclusion

In this paper, the challenging noisy and/or underdetermined BSS problem is considered. To address this issue, we have proposed a GMM-EM approach in which the non-Gaussianity of the sources is exploited by modeling their distributions using GMM. Then, the mixing coefficients, the GMM parameters and the noise covariance matrix are estimated by maximizing their posterior probabilities using an EM algorithm. Finally, issues regarding the practical implementation and performance of the proposed GMM-EM algorithm, such as the initialization scheme for the parameters, the convergence performance and computational complexity, are also discussed. Simulation results have shown that the proposed GMM-EM algorithm gives promising results with considerable computational complexity in two difficult cases: low SNR and underdetermined mixtures. Taking into account the noise in the model and jointly estimating its covariance are the main reasons for the robust performance achieved by the proposed GMM-EM method in noisy environments. The competitive separation performance achieved by the proposed algorithm in underdetermined cases is mainly due to the incorporation of prior information by conjugate priors which facilitates the recovery of the sources without taking the inverse of the mixing matrix.

Acknowledgements This work is supported in part by the National Natural Science Foundation of China under Grant 61601477 and the Engineering and Physical Sciences Research Council (EPSRC) Grant No. EP/K014307/1.

Appendix 1: Proof of Equation (13)

Since

$$\begin{aligned} & f\left(\mathbf{s}, Y|\mathbf{x}, \mathbf{A}^g, \mathbf{R}_w^g, \Theta^g\right) \\ & \quad = \prod_{t=1}^{T} f\left(\mathbf{s}(t)|\mathbf{x}, \mathbf{A}^g, \mathbf{R}_w^g, \Theta^g\right) f\left(y(t)|\mathbf{s}(t), \mathbf{x}, \mathbf{A}^g, \mathbf{R}_w^g, \Theta^g\right) \end{aligned} \tag{39}$$

On the other hand,

$$\begin{aligned} & f(\mathbf{x}, \mathbf{s}, Y|\mathbf{A}, \mathbf{R}_w, \Theta) \\ & \quad = \prod_{t=1}^{T} f\left(\mathbf{x}(t)|\mathbf{s}(t), \mathbf{A}, \mathbf{R}_w\right) f\left(\mathbf{s}(t)|y(t), \Theta\right) \end{aligned} \tag{40}$$

Substituting (39) and (40) in (11), it is straightforward to derive that

$$\begin{aligned} J = & \sum_{t=1}^{T} \sum_{m=1}^{M} \int_{\mathbf{s}} f(y(t) = m|\mathbf{s}(t), \mathbf{x}, \mathbf{A}^g, \mathbf{R}_w^g, \Theta^g) \\ & \log \omega_m f(\mathbf{s}(t)|y(t) = m, \Theta) \mathrm{d}\mathbf{s} \\ & + \sum_{t=1}^{T} \int_{\mathrm{s}} f(\mathbf{s}(t)|\mathbf{x}, \mathbf{A}^g, \mathbf{R}_w^g, \Theta^g) \log f(\mathbf{x}(t)|\mathbf{s}(t), \mathbf{A}, \mathbf{R}_w) \mathrm{d}\mathbf{s} \end{aligned}$$

Appendix 2: Proof of Equation (14)

Based on the Bayesian theory, it is easy to obtain

$$\begin{aligned} & f\left(\mathbf{s}(t)|\mathbf{x}(t), \mathbf{A}^g, \mathbf{R}_w^g, \Theta^g\right) \\ & \quad = \sum_{y(t)=1}^{M} f\left(\mathbf{x}(t)|\mathbf{s}(t), \mathbf{A}^g, \mathbf{R}_w^g, \Theta^g\right) \\ & \qquad f\left(\mathbf{s}(t)|y(t) = m, \mathbf{A}^g, \mathbf{R}_w^g, \Theta^g\right) \end{aligned} \tag{41}$$

Hence,

$$\begin{aligned} & f(\mathbf{x}(t), \mathbf{s}(t), \mathbf{A}^g, \mathbf{R}_w^g, \Theta^g) \\ & \quad = \frac{1}{\left|2\pi \mathbf{R}_w^g\right|^{1/2}} \\ & \qquad \exp\left\{-\frac{1}{2}(\mathbf{x}(t) - \mathbf{A}^g\mathbf{s}(t))^{\mathrm{T}}(\mathbf{R}_w^g)^{-1}(\mathbf{x}(t) - \mathbf{A}^g\mathbf{s}(t))\right\} \end{aligned}$$

$$\begin{aligned}
&\sum_{m=1}^{M} \omega_m^g \frac{1}{\left|2\pi \mathbf{C}_m^g\right|^{1/2}} \\
&\quad \exp\left\{-\frac{1}{2}(\mathbf{s}(t)-\boldsymbol{\mu}_m^g)^{\mathrm{T}}(\mathbf{C}_m^g)^{-1}(\mathbf{s}(t)-\boldsymbol{\mu}_m^g)\right\} \\
&= \sum_{m=1}^{M} \omega_m^g \frac{1}{\left|2\pi \mathbf{R}_w^g\right|^{1/2}} \frac{1}{\left|2\pi \mathbf{C}_m^g\right|^{1/2}} \\
&\quad \exp\left\{-\frac{1}{2}(\mathbf{x}(t)-\mathbf{A}^g\mathbf{s}(t))^{\mathrm{T}}(\mathbf{R}_w^g)^{-1}(\mathbf{x}(t)-\mathbf{A}^g\mathbf{s}(t))\right\} \\
&\quad \exp\left\{-\frac{1}{2}(\mathbf{s}(t)-\boldsymbol{\mu}_m^g)^{\mathrm{T}}(\mathbf{C}_m^g)^{-1}(\mathbf{s}(t)-\boldsymbol{\mu}_m^g)\right\}
\end{aligned} \tag{42}$$

After a series of derivations, (42) can be simplified as

$$f(\mathbf{s}(t)|\mathbf{x}(t), \mathbf{A}^g, \mathbf{R}_w^g, \Theta^g) = \sum_{m=1}^{M} \tilde{\omega}_{mt}^g \mathcal{N}\left[\mathbf{s}(t); \tilde{\boldsymbol{\mu}}_{mt}^g, \tilde{\mathbf{C}}_{mt}^g\right]$$

where

$$\begin{cases}
\tilde{\mathbf{C}}_{mt}^g = \left((\mathbf{A}^g)^{\mathrm{T}}(\mathbf{R}_w^g)^{-1}\mathbf{A}^g + (\mathbf{C}_m^g)^{-1}\right)^{-1} \\
\tilde{\boldsymbol{\mu}}_{mt}^g = \left(\tilde{\mathbf{C}}_{mt}^g\right)\left((\mathbf{A}^g)^{\mathrm{T}}(\mathbf{R}_w^g)^{-1}\mathbf{x}(t) + (\mathbf{C}_m^g)^{-1}\boldsymbol{\mu}_m^g\right) \\
\tilde{\omega}_{mt}^g = \omega_m^g \left(\left|(\tilde{\mathbf{C}}_{mt}^g)\right|^{1/2} \Big/ \left|2\pi \mathbf{R}_w^g\right|^{1/2}\left|\mathbf{C}_m^g\right|^{1/2}\right) \\
\qquad \exp\left\{-\frac{1}{2}\left[\mathbf{x}^{\mathrm{T}}(t)(\mathbf{R}_w^g)^{-1}\mathbf{x}(t)\right.\right. \\
\qquad\qquad \left.\left. + (\boldsymbol{\mu}_m^g)^{\mathrm{T}}(\mathbf{C}_m^g)^{-1}\boldsymbol{\mu}_m^g - (\tilde{\boldsymbol{\mu}}_{mt}^g)^{\mathrm{T}}(\tilde{\mathbf{C}}_{mt}^g)^{-1}\tilde{\boldsymbol{\mu}}_{mt}^g\right]\right\}
\end{cases}$$

Appendix 3: Definition of Similarity Score

In order to measure the separation performance, the similarity score is introduced to evaluate the separation performance of the proposed algorithm

$$\rho_{ii} = \sum_{t=1}^{T} s_i(t)\hat{s}_i(t) \Bigg/ \sqrt{\sum_{t=1}^{T}(s_i(t))^2 \sum_{t=1}^{T}(\hat{s}_i(t))^2} \tag{43}$$

where $\hat{s}_i(t)$ is the ith recovered source signal. ρ_{ii} depicts the similarity between the ith original source signal and the corresponding recovered source signal. It is clear that the larger the value of ρ_{ii}, the higher the degree of similarity between the original sources and the recovered sources.

References

1. S. Amari, A. Cichocki, Adaptive blind signal processing-neural network approaches. Proc. IEEE **86**(10), 2026–2048 (1998)
2. A.J. Bell, T.J. Sejnowski, An information-maximization approach to blind separation and blind deconvolution. Neural Comput. **7**, 1129–1159 (1995)
3. A. Belouchrani, J.F. Cardoso, Maximum likelihood source separation for discrete sources, in *Elsevier EUSIPCO'94* (Edinburgh, 1994)
4. J. Bilmes, *A Gentle Tutorial on the EM Algorithm and Its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models*, Technical Report ICSI-TR-97-021 (University of Berkelym 1997). http://www.citeseer.nj.nec.com/blimes98gentle.html
5. J.F. Cardoso, B.H. Laheld, Equivariant adaptive source separation. IEEE Trans. Signal Process. **44**(12), 3017–3030 (1996)
6. J.F. Cardoso, A. Souloumiac, Blind beamforming for non-Gaussian signals, in *IEE Proceedings F on Radar and Signal Processing*, vol. 140(6) (1993), pp. 362–370
7. P. Comon, M. Rajih, Blind identification of underdetermined mixtures based on the characteristic function. Signal Process. **86**(9), 2671–2681 (2006)
8. A.P. Dempster, N.M. Laird, D.B. Rubin, Maximum likelihood from incomplete data via the EM algorithm. J. R. Stat. Soc. Ser. B **39**(1), 1–38 (1977)
9. J.L. Gauvain, C.H. Lee, Maximum a posteriori estimation for multivariate Gaussian mixture observations of chains. IEEE Trans. Speech Audio Process. **2**(2), 291–298 (1994)
10. F. Gu, H. Zhang, D. Zhu, Blind separation of complex sources using generalised generating function. IEEE Signal Process. Lett. **20**(1), 71–74 (2013)
11. F. Gu, H. Zhang, W. Wang, D. Zhu, Generalized generating function with tucker decomposition and alternating least squares for underdetermined blind identification. EURASIP J. Adv. Signal Process. (2013). doi:10.1186/1687-6180-2013-124
12. F. Gu, H. Zhang, D. Zhu, Blind separation of non-stationary sources using continuous density Markov models. Digit. Signal Process. **23**(5), 1549–1564 (2013)
13. F. Gu, H. Zhang, Y. Xiao, A Bayesian approach to blind separation of mixed discrete sources by Gibbs sampleing, in *Lecture Notes on Computer Science*, vol. 6905 (2011), pp. 463–475
14. http://ai.korea.ac.kr/classes/2004/cse827/doc/map.1994.ieee.291, Accessed in 2013
15. Q. Huang, J. Yang, Y. Xue, Y. Zhou, Temporally correlated source separation based on variational Kalman smoother. Digit. Signal Process. **18**(3), 422–433 (2008)
16. Q. Huo, C. Chan, *Bayesian Adaptive Learning of the Parameters of the Hidden Markov Model for Speech Recognition*, HKU CSIS Technical Report TR-92-08 (1992). http://www.csis.hku.hk/research/techreps/document/TR-92-08
17. W. Hwang, J. Ho, Y. Lin, *Null Space Component Analysis for Noisy Blind Source Separation*, Technical Report TR-IIS-13-001 (2014). http://www.iis.sinica.edu.tw/page/library/TechReport/tr2013/tr13.html
18. A. Hyvärinen, Fast and robust fixed-point algorithms for independent component analysis. IEEE Trans. Neural Netw. **10**(3), 626–634 (1999)
19. A. Karfoul, L. Albera, G. Birot, Blind underdetermined mixture identification by joint canonical decomposition of HO cumulants. IEEE Trans. Signal Process. **58**(2), 638–649 (2010)
20. S. Kim, C.D. Yoo, Underdetermined blind source separation based on generalized Gaussian distribution, in *Proceedings of the 16th IEEE Signal Processing Society Workshop on Machine Learning for Signal Processing* (Arlington, VA, 2006), pp. 103–108
21. K.H. Knuth, Informed source separation: a Bayesian tutorial, in *Proceedings of the 13th European Signal Processing Conference (EUSIPCO 2005)* (Antalya, 2005)
22. S. Kullback, R.A. Leibler, On information and sufficiency. Ann. Math. Stat. **22**(1), 79–86 (1951)
23. L.D. Lathauwer, J. Castaing, J.F. Cardoso, Fourth-order cumulant-based blind identification of underdetermined mixtures. IEEE Trans. Signal Process. **55**(6), 2965–2973 (2007)
24. L.D. Lathauwer, J. Castaing, Blind identification of underdetermined mixtures by simultaneous matrix diagonalization. IEEE Trans. Signal Process. **56**(3), 1096–1105 (2008)
25. C.-H. Lee, J.-L. Gauvain, Speaker adaptation based on MAP estimation of HMM parameters, in *Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing* (1993), pp. 558–561

26. J.Q. Li, A.R. Barron, Mixture density estimation, in *Advances in Neural Information Processing Systems*, vol. 12 (MIT Press, Cambridge, 2000), pp. 279–285
27. X. Luciani, A.L.F. de Almeida, P. Comon, Blind identification of underdetermined mixtures based on the characteristic function: the complex case. IEEE Trans. Signal Process. **59**(2), 540–553 (2011)
28. J. Ma, L. Xu, M.I. Jordan, Asymptotic convergence rate of the EM algorithm for Gaussian mixtures. Neural Comput. **12**, 2881–2907 (2000)
29. E. Moulines, J.F. Cardoso, E. Gassiat, Maximum likelihood for blind separation and deconvolution of noisy signals using mixture models, in *International Conference on Acoustics, Speech, and Signal Processing* (Munich, 1997), pp. 3617–3620
30. S. Peng, W. Hwang, Null space pursuit: an operator-based approach to adaptive signal processing. IEEE Trans. Signal Process. **58**(5), 2475–2483 (2010)
31. M. Razaviyayn, M. Hong, Z. Luo, *A Unified Convergence Analysis of Block Successive Minimization Methods for Nonsmooth Optimization*, arXiv:1209.2385 [math.OC] (2012)
32. T. Routtenberg, J. Tabrikian, MIMO-AR system identification and blind source separation for GMM-distributed sources. IEEE Trans. Signal Process. **57**(5), 1717–1730 (2009)
33. T. Rydn, EM versus chain Monte Carlo for estimation of hidden Markov models: a computational perspective. Bayesian Anal. **3**, 659–688 (2008)
34. G. Schwarz, Estimation the dimension of a model. Annu. Stat. **6**(2), 461–464 (1978)
35. H. Snoussi, A.M. Djafari, Unsupervised learning for source separation with mixture of Gaussians prior for sources and Gaussian prior for mixture coefficients, in *Proceedings of the 2001 IEEE Signal Processing Society Workshop on Neural Networks for Signal Processing XI* (2001), pp. 293-302
36. H. Snoussi, J. Idier, Bayesian blind separation of generalized hyperbolic processes in noisy and underdetermined mixtures. IEEE Trans. Signal Process. **54**(9), 3257–3269 (2006)
37. S. Sun, C. Peng, W. Hou, J. Zheng, Y. Jiang, X. Zheng, Blind source separation with time series variational Bayes expectation maximization algorithm. Digit. Signal Process. **12**(1), 17–33 (2012)
38. K. Todros, J. Tabrikian, Blind separation of independent sources using Gaussian mixture model. IEEE Trans. Signal Process. **55**(7), 3645–3658 (2007)
39. P. Tseng, Convergence of a block coordinate descent method for nondifferentiable minimization. J. Optim. Theory Appl. **109**, 475–494 (2001)
40. L. Xu, M.I. Jordan, On convergence properties of the EM algorithm for Gaussian mixtures. Neural Comput. **8**, 129–151 (1996)
41. Y. Zhang, X. Shi, C.H. Chen, A Gaussian mixture model for underdetermined independent component analysis. Signal Process. **86**(6), 1538–1549 (2006)
42. Y. Zhao, Image segmentation using temporal-spatial information in dynamic scenes, in *Proceedings of the IEEE International Conference on Machine Learning and Cybernetics* (2003)