BEN-GURION UNIVERSITY OF THE NEGEV
FACULTY OF ENGINEERING SCIENCE
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

MAXIMUM LIKELIHOOD BASED TECHNIQUES FOR BLIND SOURCE SEPARATION AND APPROXIMATE JOINT DIAGONALIZATION

Thesis submitted in partial fulfillment of the requirements towards the M.Sc. degree

By Koby Todros

October 2006

Abstract

In this work, two novel algorithms for blind separation of noiseless instantaneous linear mixture of independent sources are presented. The proposed algorithms exploit non-Gaussianity of the independent sources by modeling their distribution using the Gaussian mixture model (GMM). The first proposed method is based on the maximum likelihood (ML) estimator. According to this method, the sensors distribution parameters are estimated via the expectation-maximization (EM) algorithm for GMM parameter estimation and the separation matrix is estimated by applying nonorthogonal joint diagonalization of the estimated GMM covariance matrices. The second proposed method is also ML-based approach. According to this method, the distribution parameters of the pre-whitened sensors are estimated via the EM algorithm for GMM parameter estimation and a unitary separation matrix is estimated by applying orthogonal joint diagonalization of the estimated GMM covariance matrices. It is shown that estimation of the sensors distribution parameters via the EM algorithm for GMM parameter estimation amounts to obtaining a tight lower bound on the log-likelihood of the separation matrix. It is also shown that joint diagonalization of the estimated GMM covariance matrices amounts to maximization of the obtained tight lower bound w.r.t. the separation matrix. The performances of the two proposed methods are evaluated and compared to existing blind source separation techniques. The results show superior performances of the proposed methods in terms of interference-to-signal ratio.

In addition, a new efficient iterative algorithm for approximate joint diagonalization of positive-definite Hermitian matrices is presented. According to the proposed algorithm, the joint diagonalization matrix is not constrained to be orthogonal and it is estimated by iterative optimization of a ML-based objective function. The columns of the joint diagonalization matrix are estimated separately using iterative singular value decompositions of a weighted sum of the matrices to be diagonalized. This property enables low computational load of the proposed joint diagonalization algorithm, which is most useful in cases of large amount of matrices. The performance of the proposed algorithm is evaluated and compared to other state-of-the-art algorithms for approximate joint diagonalization. The results imply that the proposed algorithm is computationally efficient with performance similar to state-of-the-art algorithms for approximate joint diagonalization.

Acknowledgment

I would like to dedicate this work to my sister, Efrat Todros, and to my best friend, Zohar Levi, who supported and encouraged me through all this long and harsh period. I would also like to greatly thank my thesis supervisor, Dr. Joseph Tabrikian, for his devoted and great supervisory.

Contents

Abbreviations

AAMSE – Approximated Averaged Mean Square Error

AC/DC – Alternating Columns/Diagonal Centers

BSS – Blind Source Separation

EM – Expectation Maximization

FFDIAG – Fast Forbenius Diagonalization

FG – Flurry Gautschi

GMM – Gaussian Mixture Model

ICA – Independent Component Analysis

IFA – Independent Factor Analysis

ISR – Interference-to-Signal Ratio

JADE – Joint Approximate Diagonalization of Eigenmatrices

KL – Kullback Leibler

LRF – Learning Rate Factor

ML – Maximum Likelihood

NIFA – Noiseless Independent Factor Analysis

PDF – Probability Density Function

SVD – Singular Value Decomposition

SVDJD – Singular Value Decomposition Joint Diagonalization

List of Figures

1. Introduction

Blind separation of an instantaneous linear mixture of independent sources can be achieved, up to scaling and permutation of the estimated sources, by exploiting their non-Gaussianity [1]. Some existing methods for BSS use restrictive assumptions on the sources distribution, which makes them inapplicable in some cases. For example, cumulant-based methods like JADE [3], assume that the sources have non-zero 4^{th} order cumulant and the PDF of each source is approximated by using only 2^{nd} and 4^{th} order cumulants.

In [14] it is shown that any density can be estimated to any desired degree of approximation, in terms of Kullback-Leibler (KL) divergence [20], using a finite order GMM. Therefore, in this work the probability density function (PDF) of the non-Gaussian sources is modeled by GMM.

Several researchers have utilized the GMM in solving the BSS problem. For example, Moulines et al. [4] developed an approximate maximum likelihood (ML) method for blind separation and deconvolution of noisy linear mixtures, where the density of each source was modeled by a univariate GMM. According to this approach, an expectation-maximization (EM) algorithm [10], which jointly estimates the mixing matrix, the source distribution parameters and the noise covariance matrix, was developed. A similar approach for blind separation of noisy linear mixtures, named as independent factor analysis (IFA), was offered by Attias [5]. In contrast to [4], the intractability of the EM algorithm when the number of sources increases and the sources reconstruction problem were handled. In [5], Attias extended this method for the case of noiseless linear mixtures and an algorithm, named noiseless IFA (NIFA), for joint estimation of the sources distribution parameters and the separation matrix was developed. This algorithm is strictly EM only for sufficiently small, empirically selected, learning rate factor (LRF), used for updating the estimation of the separation matrix in each step of the algorithm. In [6], Attias extended the IFA method for the case of temporally structured sources. In order to capture the temporal statistical properties of the observed data, each source was described by a hidden Markov model (HMM) and a family of EM algorithms that learn the structure of the underlying sources and their relation to the observed data were derived. In [7], a constraint EM algorithm for blind separation of linear mixture with isotropic noise was developed. The mixing matrix in this algorithm was subject to an orthogonality constraint. Under this constraint, exponential increase of complexity with the number of sources was avoided. In [8] a modified EM algorithm for blind separation of independent linear sparse over-complete noisy mixtures was derived. The sparsity assumption enabled a number of simplifying approximations to the observations density, which avoided exponential growth of the number of mixture components.

The algorithms mentioned above utilize an EM algorithm, which jointly estimates the unobserved source distribution parameters and the mixing matrix coefficients. This approach has the following disadvantages. First, accurate initialization and order selection of the distribution model of the unobserved source signals is difficult, so the EM algorithm may converge into undesired maxima. Second, implementation of this approach is cumbersome.

In [9], a method for BSS in nonstationary environments using a non-Gaussian mixture model was developed. According to this method, the observed data were modeled as a mixture of several mutually exclusive classes, where each class was described by a different noisy instantaneous linear mixture of independent non-Gaussian densities. The density of the sources in each class was modeled by the generalized Gaussian density function [21] and a gradient descent algorithm for ML estimation of the model parameters was derived. However, in this contribution we focus our interest in blind separation of sources related to a unique class, where the density of each source can be modeled by GMM.

In this work, the BSS problem is solved in two separate steps. In the first step, the sensors distribution parameters are estimated via the EM algorithm for GMM parameter estimation [11]. It is shown that this operation, amounts to obtaining a tight lower bound on the log-likelihood of a function of the separation matrix. In the second step, the separation matrix is estimated by maximizing the tight lower bound, obtained in the first step, w.r.t. its entries. Based on this approach, two novel source separation techniques are proposed. The first proposed method is a ML-based technique, which comprises the following steps: 1) estimation of the sensors distribution parameters via the EM algorithm for GMM parameter estimation [11], 2) estimation of a separation matrix, which approximately diagonalizes the estimated GMM covariance matrices simultaneously. The joint diagonalization is performed according to an algorithm offered by Pham [2], [25] and according to a new joint diagonalization algorithm, offered in this work and denoted as the SVDJD algorithm. BSS using PHAM and SVDJD techniques is denoted as the GMMPHAM and GMMSVDJD algorithms, respectively.

The second proposed method is also a ML-based technique, which comprises the following steps: 1) estimation of the pre-whitened sensors distribution parameters via the EM algorithm for GMM parameter estimation [11], 2) estimation of a unitary separation matrix, which approximately diagonalizes the estimated GMM covariance matrices simultaneously, according to an algorithm offered by Flury and Gautschi [13]. BSS using the Flury Gautschi method is denoted as the GMMFG algorithm. The two methods are also presented in [30].

As mentioned above, a new efficient iterative algorithm for approximate joint diagonalization of positive-definite Hermitian matrices is proposed in this work. A variety of algorithms for approximate joint diagonalization have been proposed in the literature. To name a few, the FG algorithm for simultaneous orthogonal transformation of several positive-definite symmetric matrices to nearly diagonal form was proposed by Flury and Gautchi [13]; Cardoso and Souloumiac [24] proposed the extended Jacobi method for orthogonal joint diagonalization; An algorithm for nonorthogonal joint diagonalization of positive-definite Hermitian matrices was proposed by Pham [2], [25]; Joint diagonalization of certain algebraically derived matrices via subspace fitting techniques was proposed by van der Veen [26]; Yeredor [27] proposed the AC/DC algorithm for nonorthogonal joint diagonalization in the least-squares sense, using subspace fitting techniques; Joho and Rahbar [28] developed a method for approximate joint diagonalization of correlation matrices using Newton methods; Ziehe et. al [29] proposed a fast algorithm for nonorthogonal joint diagonalization, named as the FFDIAG algorithm.

Many techniques for blind source separation [1] utilize approximate joint diagonalization algorithms. According to these techniques, a set of unknown matrices, which obey exact joint diagonalization, is estimated from the observed data. A diagonalization matrix, which is usually the separation matrix, is estimated (up to scaling and permutation of rows) by minimizing an objective function that measures the deviation of the diagonalized matrices from diagonality. For example, joint diagonalization of fourth-order joint-cumulants matrices was performed via the extended Jacobi method [24] in the JADE algorithm, offered by Cardoso [3]; Pham and Cardoso [2] utilized Pham's algorithm [25] for joint diagonalization of an estimated set of covariance matrices in the context of blind separation of nonstationary Gaussian sources; Todros and Tabrikian [30] utilized Pham's algorithm [2], [25], and the FG method [13] for joint diagonalization in the context of blind separation of independent sources using GMM [14]; van der Veen and Paulraj proposed joint diagonalization of certain algebraically derived matrices, in the contexts of blind separation of constant modulus sources [32].

In this work, a new efficient iterative algorithm for approximate joint diagonalization of positive-definite Hermitian matrices, named as the SVDJD algorithm is proposed. The positive-definite assumption is motivated by the fact that in many applications [2], [30], the matrices to be diagonalized are covariance matrices of some random variables. According to the proposed algorithm, a diagonalization matrix, which is not constrained to be orthogonal, is estimated by optimization of a ML-based objective function, also used by Pham [25]. The columns of the diagonalization matrix are estimated separately using iterative singular value decompositions (SVD) of a weighted sum of the matrices to be diagonalized. This technique enables

low computational load, which is practical especially in cases of large amount of matrices. This method is also presented in [31].

The thesis is organized as follows. In Chapter 2, an application of the GMM for blind separation of independent sources is presented. In Chapter 3, a novel algorithm for approximate joint diagonalization of positive-definite Hermitian matrices is derived. Simulation results as well as discussion on each topic are given at the end of each chapter. Finally, Chapter 4 summarizes the main points of this contribution.

2. Application of Gaussian Mixture Model for Blind Separation of Independent Sources

Consider the following noiseless instantaneous linear mixture model:

$$\mathbf{x}_t = \mathbf{A}\mathbf{s}_t, \quad t = 1,2,...,T. \tag{2.1}$$

The random vector $\mathbf{s}_t = \left[s_{1,t},...,s_{K,t}\right]^{\mathrm{T}}$, representing K statistically independent sources at time instance t, is mixed by a fixed unknown $L \times K$ $\left(L \geq K\right)$ mixing matrix $\mathbf{A}$. The observation vector $\mathbf{x}_t = [x_{1,t},...,x_{L,T}]^{\mathrm{T}}$ is obtained from an array of L sensors. The problem of BSS addresses the reconstruction of the source vectors $\{\mathbf{s}_t\}_{t=1}^{T}$, by estimating a $K \times L$ separation matrix $\mathbf{B}$ for which

$$\hat{\mathbf{s}}_t = \mathbf{B}\mathbf{x}_t, \quad t = 1,2,...,T. \tag{2.2}$$

In this work, the BSS problem is solved in two separate steps. In the first step, a tight lower bound on the log-likelihood of a function of $\mathbf{B}$ is obtained by applying the EM algorithm [10] for GMM parameter estimation [11] of the *sensors* distribution parameters. In the second step, the obtained tight lower bound is maximized w.r.t. $\mathbf{B}$ by applying approximate joint diagonalization of the estimated GMM covariance matrices.

This chapter is organized as follows. In Section 2.1, mathematical models for the PDF's of the source and observation signals are derived. In Section 2.2, a novel technique for solving the BSS problem is presented under the assumption of equal number of sensors and sources $\left(L = K\right)$. In Section 2.3, the case of more sensors than sources is addressed, and in Section 2.4, the performance of the proposed method is evaluated and compared to other existing methods for BSS. Finally, Section 2.5 summarizes the main points of this chapter.

2.1 DERIVATION OF THE SOURCE AND SENSOR DISTRIBUTION MODELS

In this section, derivation of the source and sensor distribution models is carried out under the assumption of stationary and non-Gaussian source signals.

2.1.1 SOURCE DISTRIBUTION MODEL

The PDF of the k^{th} source signal at time instance *t* is modeled by GMM in the following manner:

$$f_s\left(s_{k,t};\boldsymbol{\theta}_k^{(s)}\right)=\sum_{l_k=1}^{n_k}\vartheta_{k,l_k}\Phi\left(s_{k,t};\mu_{k,l_k},\sigma_{k,l_k}^2\right),\ k=1,...,K\ , \tag{2.3}$$

where $\Phi(\cdot;\cdot,\cdot)$ denotes the proper complex Gaussian density function and n_k denotes the number of Gaussians. The mixing proportions are denoted by $\left\{\vartheta_{k,l_k}\right\}_{l_k=1}^{n_k}$, such that $\sum_{l_k=1}^{n_k}\vartheta_{k,l_k}=1$. The means and variances of the Gaussians are denoted by $\left\{\mu_{k,l_k}\right\}_{l_k=1}^{n_k}$ and $\left\{\sigma_{k,l_k}^2\right\}_{l_k=1}^{n_k}$, respectively. The vector of unknown distribution parameters of the k^{th} source signal is denoted by $\boldsymbol{\theta}_k^{(s)}=\{\vartheta_{k,l_k},\mu_{k,l_k},\sigma_{k,l_k}^2\}_{l_k=1}^{n_k}$. It is noted that $s_{k,t}$, $t=1,2,...,T$ are i.i.d $\forall\ k=1,...,K$. By applying the assumption of independent source signals, their joint PDF can be formulated as follows:

$$\begin{aligned}
f_{\mathbf{s}}\left(\mathbf{s}_t;\boldsymbol{\theta}^{(\mathbf{s})}\right)&=\prod_{k=1}^{K}f_s\left(s_{k,t};\boldsymbol{\theta}_k^{(s)}\right)\\
&=\sum_{l_1=1}^{n_1}\sum_{l_2=1}^{n_2}\cdots\sum_{l_K=1}^{n_K}\vartheta_{1,l_1}\vartheta_{2,l_2}\cdots\vartheta_{K,l_K}\Phi\left(s_{1,t};\mu_{1,l_1},\sigma_{1,l_1}^2\right)\Phi\left(s_{2,t};\mu_{2,l_2},\sigma_{2,l_2}^2\right)\cdots\Phi\left(s_{K,t};\mu_{K,l_K},\sigma_{K,l_K}^2\right)\\
&=\sum_{l_1=1}^{n_1}\sum_{l_2=1}^{n_2}\cdots\sum_{l_K=1}^{n_K}w_{l_1,l_2,...,l_K}\Phi\left(\left[s_{1,t},...,s_{K,t}\right]^{\mathrm{T}};\left[\mu_{1,l_1},...,\mu_{K,l_K}\right]^{\mathrm{T}},diag\left(\sigma_{1,l_1}^2,...,\sigma_{K,l_K}^2\right)\right)\\
&=\sum_{m=1}^{M}w_m\Phi\left(\mathbf{s}_t;\boldsymbol{\mu}_m,\mathbf{C}_m\right),
\end{aligned} \tag{2.4}$$

where $M=\prod_{k=1}^{K}n_k$ is the total number of Guassians in the joint PDF and $w_m=\prod_{k=1}^{K}\vartheta_{k,l_k}$; $m=1,...,M$ are the mixing proportions of each Gaussian such that $\sum_{m=1}^{M}w_m=1$. The index *m*, denotes a unique combination of Gaussians from each source, i.e. $l_1,...,l_K\rightarrow m$, where $l_k\in[1,...,n_k]$ denotes a Gaussian index of the k^{th} source. The mean vector and covariance matrix of the m^{th} Gaussian are denoted by $\boldsymbol{\mu}_m=[\mu_{1,l_1},...,\mu_{K,l_K}]^{\mathrm{T}}$

and $\mathbf{C}_m = diag\left(\sigma^2_{1,l_1},...,\sigma^2_{K,l_K}\right)$, respectively. The vector of unknown parameters of the joint PDF is denoted by $\boldsymbol{\theta}^{(\mathbf{s})} = \left\{w_m, \boldsymbol{\mu}_m, \mathbf{C}_m\right\}_{m=1}^{M}$. Equation (2.4) implies that the joint PDF of the sources is a multivariate GMM with diagonal covariance matrices.

2.1.2 SENSOR DISTRIBUTION MODEL

In this subsection, the generative model of the observation signals at time instance *t* is utilized in order to derive an expression for their joint PDF. Fig. 1 depicts a graphical model corresponding to the generation process of an observation at time instance *t*. According to this generative model, at every time instance a hidden indication vector, $\mathbf{y}_t = [y_{t,1},...,y_{t,M}]^{\mathrm{T}}$, which indicates the generating Gaussian of $\mathbf{s}_t$, is randomized by the following discrete PDF:

$$f_{\mathbf{y}}\left(\mathbf{y}_t\right) = \sum_{m=1}^{M} w_m \delta\left(y_{t,m} - 1\right), \tag{2.5}$$

where $\delta(\cdot)$ denotes the dirac's delta function and

$$y_{t,m} = \begin{cases} 1, & \text{if } \mathbf{s}_t \text{ is generated by the } m^{\text{th}} \text{ Gaussian} \\ 0, & \text{otherwise} \end{cases}. \tag{2.6}$$

Hidden values of $\mathbf{s}_t$ are then mixed by the mixing matrix $\mathbf{A}$ and an observation vector $\mathbf{x}_t$ is formed. According to the Bayes theorem,

$$f_{\mathbf{x}}\left(\mathbf{x}_t; \boldsymbol{\theta}^{(\mathbf{x})}\right) = E_{\mathbf{y}}\left[f_{\mathbf{x}|\mathbf{y}}(\mathbf{x}_t \mid \mathbf{y}_t; \boldsymbol{\theta}^{(\mathbf{x})})\right], \tag{2.7}$$

where $E_{\mathbf{y}}$ denotes the expectation operator w.r.t. the random vector $\mathbf{y}$ and the vector of unknown distribution parameters of the observation signals is denoted by

$$\boldsymbol{\theta}^{(\mathbf{x})} = \left\{w_m, \mathbf{A}\boldsymbol{\mu}_m, \mathbf{A}\mathbf{C}_m\mathbf{A}^H\right\}_{m=1}^{M}. \tag{2.8}$$

Therefore, the PDF of $\mathbf{x}_t$ is given by

$$f_{\mathbf{x}}\left(\mathbf{x}_t; \boldsymbol{\theta}^{(\mathbf{x})}\right) = \sum_{m=1}^{M} w_m f_{\mathbf{x}|\mathbf{y}}\left(\mathbf{x}_t \mid y_{t,m} = 1; \boldsymbol{\theta}^{(\mathbf{x})}\right), \tag{2.9}$$

where according to the generative model of the observation signals

$$f_{\mathbf{x}|\mathbf{y}}\left(\mathbf{x}_t \mid y_{t,m} = 1; \boldsymbol{\theta}^{(\mathbf{x})}\right) = \Phi\left(\mathbf{x}_t; \mathbf{A}\boldsymbol{\mu}_m, \mathbf{A}\mathbf{C}_m\mathbf{A}^H\right). \tag{2.10}$$

Thus, the joint PDF of the observation signals is also GMM with nondiagonal covariance matrices, as formulated in the following equation:

$$f_{\mathbf{x}}\left(\mathbf{x}_t;\boldsymbol{\theta}^{(\mathbf{x})}\right)=\sum_{m=1}^{M} w_m \Phi\left(\mathbf{x}_t;\boldsymbol{\eta}_m,\mathbf{R}_m\right), \tag{2.11}$$

where $\boldsymbol{\eta}_m = \mathbf{A}\boldsymbol{\mu}_m$ and $\mathbf{R}_m = \mathbf{A}\mathbf{C}_m\mathbf{A}^H$.

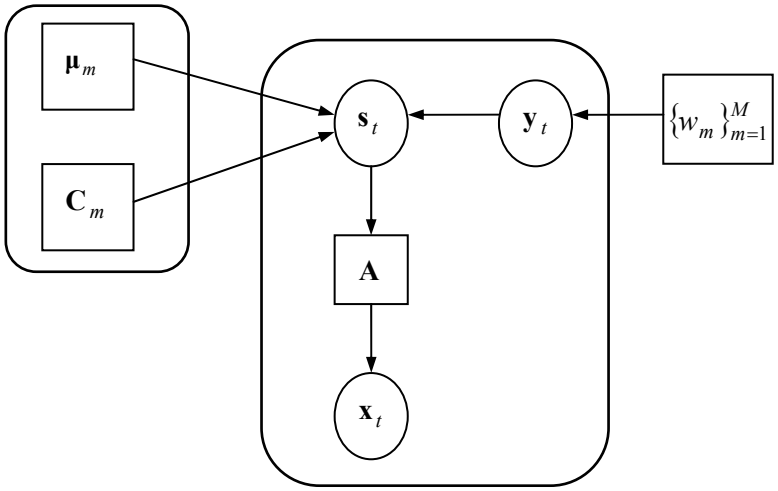

Fig. 1. The generative model of the observation signals at time instance *t*.

The generation model of the source signals implies that at every time instance the source vector, $\mathbf{s}_t$, may be generated by a different set of Gaussians. Therefore, $\mathbf{s}_t$ and $\mathbf{x}_t$ may be viewed as non-Gaussian and/or nonstationary multivariate random processes.

2.2 SOLUTION OF THE BSS PROBLEM

In this section, two novel techniques for solving the BSS problem, under the assumption of equal number of sensors and sources (*L*=*K*), are derived. Direct maximization of the log-likelihood of **B** is analytically cumbersome. Therefore, the log-likelihood function is maximized w.r.t. **B** in two separate steps. In the first step, a tight lower bound on the log-likelihood of a function of **B** and $\boldsymbol{\theta}^{(\mathbf{s})}=\left\{w_m,\boldsymbol{\mu}_m,\mathbf{C}_m\right\}_{m=1}^{M}$ with the observations is obtained by applying the EM algorithm for GMM parameter estimation [11]. In the second step, the obtained tight lower bound is maximized w.r.t. **B** and $\boldsymbol{\theta}^{(\mathbf{s})}$. The basic difference between the two

proposed algorithms stems from the manner in which the tight lower bound of the log-likelihood function is maximized w.r.t. to the entries of $\mathbf{B}$.

2.2.1 DERIVATION OF A ML-BASED OBJECTIVE FUNCTION

In this subsection, a tight lower bound on the log-likelihood of a function of $\mathbf{B}$ and $\boldsymbol{\theta}^{(\mathbf{s})}$ is derived by applying the EM algorithm for GMM parameter estimation [11]. A ML-based objective function is derived from the obtained tight lower bound.

In the case of equal number of sensors and sources and invertible mixing matrix $\mathbf{A}$, (2.1) and (2.2) imply $\mathbf{B} = \mathbf{A}^{-1}$. Therefore, it is implied by (2.8) that the vector of unknown distribution parameters of the observation signals can be represented as a function of the separation matrix $\mathbf{B}$ and $\boldsymbol{\theta}^{(\mathbf{s})}$ in the following manner:

$$\boldsymbol{\theta}^{(\mathbf{x})} = \left\{ w_m, \mathbf{B}^{-1}\boldsymbol{\mu}_m, \mathbf{B}^{-1}\mathbf{C}_m\mathbf{B}^{-H} \right\}_{m=1}^{M} = \boldsymbol{\rho}\left(\mathbf{B}, \boldsymbol{\theta}^{(\mathbf{s})}\right), \tag{2.12}$$

where $\mathbf{B}^{-1}\boldsymbol{\mu}_m = \boldsymbol{\eta}_m$ and $\mathbf{B}^{-1}\mathbf{C}_m\mathbf{B}^{-H} = \mathbf{R}_m$ denote the mean vector and covariance matrix of the m^{th} Gaussian component, respectively. Therefore,

$$\hat{\mathbf{B}} = \arg\max_{\mathbf{B}} \max_{\boldsymbol{\theta}^{(\mathbf{s})}} \log f_{\mathbf{X};\boldsymbol{\rho}\left(\mathbf{B},\boldsymbol{\theta}^{(\mathbf{s})}\right)}, \tag{2.13}$$

where $\mathbf{X} = \left[\mathbf{x}_1, ..., \mathbf{x}_T\right]$ denotes the matrix of observation vectors.

Direct maximization of $\log f_{\mathbf{X};\boldsymbol{\rho}\left(\mathbf{B},\boldsymbol{\theta}^{(\mathbf{s})}\right)}$ w.r.t. to $\mathbf{B}$ and $\boldsymbol{\theta}^{(\mathbf{s})}$ is analytically cumbersome. Hence, its lower bound is maximized instead. In Appendix A, it is shown that by utilizing the EM algorithm for GMM parameter estimation [11], a tight lower bound on $\log f_{\mathbf{X};\boldsymbol{\theta}^{(\mathbf{x})}}$ is obtained and a ML based estimate of $\mathbf{B}$ can be achieved. As proved in Appendix A, a tight lower bound on $\log f_{\mathbf{X};\boldsymbol{\theta}^{(\mathbf{x})}}$ is given by:

$$\log f_{\mathbf{X};\boldsymbol{\theta}^{(\mathbf{x})}} \geq D\left(\boldsymbol{\theta}^{(\mathbf{x})}, \hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right), \tag{2.14}$$

where

$$D\left(\boldsymbol{\theta}^{(\mathbf{x})}, \hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right) = \log f_{\mathbf{X};\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}} + E_{\mathbf{Y}|\mathbf{X};\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}}\left[\log f_{\mathbf{X},\mathbf{Y};\boldsymbol{\theta}^{(\mathbf{x})}}\right] - E_{\mathbf{Y}|\mathbf{X};\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}}\left[\log f_{\mathbf{X},\mathbf{Y};\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}}\right]. \tag{2.15}$$

The distribution parameters of the observation signals, obtained in the final step of the EM algorithm for GMM parameter estimation [11] are denoted by $\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}=\left\{\hat{w}_m,\hat{\boldsymbol{\eta}}_m,\hat{\mathbf{R}}_m\right\}_{m=1}^{M}$. The function $\log f_{\mathbf{X},\mathbf{Y};\boldsymbol{\theta}^{(\mathbf{x})}}$ is the joint log-likelihood of $\boldsymbol{\theta}^{(\mathbf{x})}$ with the matrices of observation vectors and of their corresponding hidden indication vectors, denoted by $\mathbf{X}=\left[\mathbf{x}_1,...,\mathbf{x}_T\right]$ and $\mathbf{Y}=\left[\mathbf{y}_1,...,\mathbf{y}_T\right]$, respectively. Hence, it is implied by (2.12) and (2.14) that

$$\log f_{\mathbf{X};\boldsymbol{\rho}\left(\mathbf{B},\boldsymbol{\theta}^{(\mathbf{s})}\right)}\geq D\left(\boldsymbol{\rho}\left(\mathbf{B},\boldsymbol{\theta}^{(\mathbf{s})}\right),\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right), \tag{2.16}$$

and therefore,

$$\hat{\mathbf{B}}=\arg\max_{\mathbf{B}}\max_{\boldsymbol{\theta}^{(\mathbf{s})}}D\left(\boldsymbol{\rho}\left(\mathbf{B},\boldsymbol{\theta}^{(\mathbf{s})}\right),\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right). \tag{2.17}$$

The following example is aimed to examine in a graphical manner the relation between $\log f_{\mathbf{X};\boldsymbol{\rho}\left(\mathbf{B},\boldsymbol{\theta}^{(\mathbf{s})}\right)}$ and $D\left(\boldsymbol{\rho}\left(\mathbf{B},\boldsymbol{\theta}^{(\mathbf{s})}\right),\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)$. Twenty five hundred samples of two source signals were synthesized by the following GMM PDF: $f_{\mathbf{s}}\left(\mathbf{s}_t;\boldsymbol{\theta}^{(\mathbf{s})}\right)=\sum_{m=1}^{4}w_m\Phi\left(\mathbf{s}_t;\boldsymbol{\mu}_m;\mathbf{C}_m\right)$, where the univariate GMM order of each source was 2. The values of the mixing proportions, mean vectors and covariance matrices were: $w_m=0.25\ \forall\ m=1,...,4$, $\boldsymbol{\mu}_1=\left[-5,5\right]^{\mathbf{T}}$, $\boldsymbol{\mu}_2=\left[-5,10\right]^{\mathbf{T}}$, $\boldsymbol{\mu}_3=\left[5,-5\right]^{\mathbf{T}}$, $\boldsymbol{\mu}_4=\left[5,10\right]^{\mathbf{T}}$, $\mathbf{C}_1=diag\left(10,5\right)$, $\mathbf{C}_2=diag\left(10,12\right)$, $\mathbf{C}_3=diag\left(5,5\right)$ and $\mathbf{C}_4=diag\left(5,12\right)$, respectively. The source signals were mixed by a unitary mixing matrix $\mathbf{A}=\begin{bmatrix}\cos\alpha & \sin\alpha\\ -\sin\alpha & \cos\alpha\end{bmatrix}$, where $\alpha=45^{\circ}$. The distribution parameters of the mixed source signals were estimated via the greedy EM algorithm for GMM parameter estimation [11]. Due to the fact that in this example **A** is unitary, the separation matrix **B** is of the form $\mathbf{B}\left(\beta\right)=\begin{bmatrix}\cos\beta & -\sin\beta\\ \sin\beta & \cos\beta\end{bmatrix}$. Hence, $\log f_{\mathbf{X};\boldsymbol{\rho}\left(\mathbf{B}(\beta)\right)}$ and $D\left(\boldsymbol{\rho}\left(\mathbf{B}\left(\beta\right),\boldsymbol{\theta}^{(\mathbf{s})}\right),\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)$ can be sketched as a function of β, as depicted in Fig. 2. According to this figure, one can observe that $D\left(\boldsymbol{\rho}\left(\mathbf{B}\left(\beta\right),\boldsymbol{\theta}^{(\mathbf{s})}\right),\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)$ is a tight lower bound on $\log f_{\mathbf{X};\boldsymbol{\rho}\left(\mathbf{B}(\beta),\boldsymbol{\theta}^{(\mathbf{s})}\right)}$ and both functions are maximized for $\beta=45^{\circ}$.

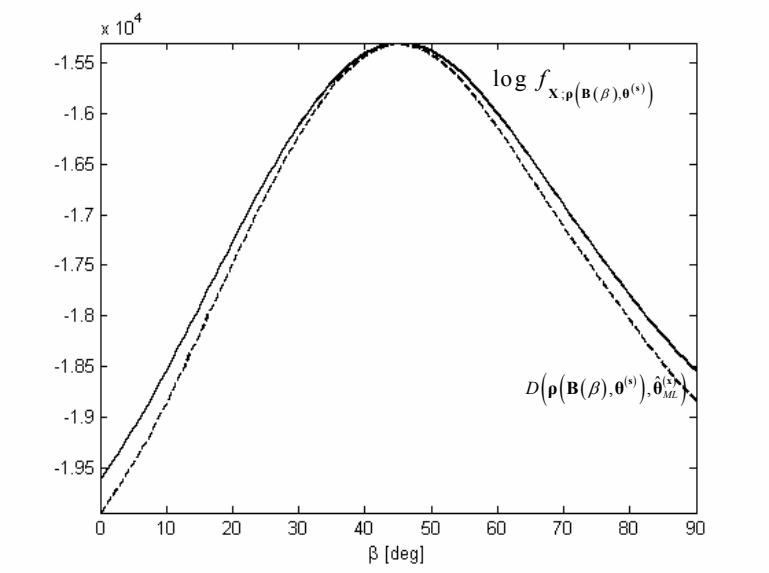

Fig. 2. The log-likelihood function of $\boldsymbol{\rho}\left(\mathbf{B}(\beta),\boldsymbol{\theta}^{(\mathbf{s})}\right)$ (solid curve) with the observations matrix, $\mathbf{X}$ and its tight lower bound (dashed curve) as a function of β.

The first and last terms in the r.h.s. of (2.15) are independent of $\boldsymbol{\theta}^{(\mathbf{x})}$ and therefore **B**-independent, while its middle term is $\boldsymbol{\theta}^{(\mathbf{x})}$-dependent and therefore **B**-dependent. Thus, by normalizing the middle term of (2.15) by a factor of $-\frac{1}{T}$, where T is the number of observation vectors, estimation of **B** can be performed in the following manner:

$$\hat{\mathbf{B}}_{ML} = \arg\min_{\mathbf{B}} \min_{\boldsymbol{\theta}^{(\mathbf{s})}} Q\left(\boldsymbol{\rho}\left(\mathbf{B},\boldsymbol{\theta}^{(\mathbf{s})}\right),\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right), \tag{2.18}$$

where the objective function

$$Q\left(\boldsymbol{\rho}\left(\mathbf{B},\boldsymbol{\theta}^{(\mathbf{s})}\right),\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right) = -\frac{1}{T} E_{\mathbf{Y}|\mathbf{X};\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}}\left[\log f_{\mathbf{X},\mathbf{Y};\boldsymbol{\rho}\left(\mathbf{B},\boldsymbol{\theta}^{(\mathbf{s})}\right)}\right] \tag{2.19}$$

is the normalized conditional expectation of $\log f_{\mathbf{X},\mathbf{Y};\boldsymbol{\rho}(\mathbf{B})}$.

In the following, a strict analytical expression of $Q\left(\boldsymbol{\rho}\left(\mathbf{B}\right),\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)$ is derived. According to (2.5), (2.6) and (2.9), the joint PDF of $\mathbf{x}_t$ and $\mathbf{y}_t$ is given by

$$f_{\mathbf{x},\mathbf{y};\boldsymbol{\rho}\left(\mathbf{B},\boldsymbol{\theta}^{(\mathbf{s})}\right)}\left(\mathbf{x}_t,\mathbf{y}_t;\boldsymbol{\rho}\left(\mathbf{B},\boldsymbol{\theta}^{(\mathbf{s})}\right)\right)=\prod_{m=1}^{M}\left[w_m\Phi\left(\mathbf{x}_t;\boldsymbol{\eta}_m,\mathbf{R}_m\right)\right]^{y_{t,m}}. \tag{2.20}$$

Assuming that $\mathbf{x}_1,...,\mathbf{x}_T$ and $\mathbf{y}_1,...,\mathbf{y}_T$ are temporally independent, the joint log-likelihood of $\boldsymbol{\rho}\left(\mathbf{B},\boldsymbol{\theta}^{(\mathbf{s})}\right)$ with $\mathbf{X}$ and $\mathbf{Y}$ is given by

$$\log f_{\mathbf{X},\mathbf{Y};\boldsymbol{\rho}\left(\mathbf{B},\boldsymbol{\theta}^{(\mathbf{s})}\right)}=\sum_{t=1}^{T}\sum_{m=1}^{M}\mathbf{y}_{t,m}\log\left(w_m\Phi\left(\mathbf{x}_t;\boldsymbol{\eta}_m,\mathbf{R}_m\right)\right). \tag{2.21}$$

The conditional expectation of (2.21) is

$$E_{\mathbf{Y}|\mathbf{X};\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}}\left[\log f_{\mathbf{X},\mathbf{Y};\boldsymbol{\rho}\left(\mathbf{B},\boldsymbol{\theta}^{(\mathbf{s})}\right)}\right]=\sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_{t,m}\log\left(w_m\Phi\left(\mathbf{x}_t,\boldsymbol{\eta}_m,\mathbf{R}_m\right)\right), \tag{2.22}$$

where

$$\gamma_{t,m}=E_{\mathbf{Y}|\mathbf{X};\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}}[y_{t,m}]. \tag{2.23}$$

Since $y_{t,m}$ can have only discrete values of 0 and 1, $\gamma_{t,m}$ can be calculated in the following manner:

$$\gamma_{t,m}=0\cdot P\left(y_{t,m}=0\,|\,\mathbf{x}_t;\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)+1\cdot P\left(y_{t,m}=1\,|\,\mathbf{x}_t;\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right). \tag{2.24}$$

Therefore, applying the Bayes theorem, $\gamma_{t,m}$ is given by

$$\gamma_{t,m}=P\left(y_{t,m}=1\,|\,\mathbf{x}_t;\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)=\frac{P(y_{t,m}=1)\cdot f\left(\mathbf{x}_t\,|\,\mathbf{y}_{t,m}=1;\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)}{f(\mathbf{x}_t;\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})})}=\frac{\hat{w}_m\Phi\left(\mathbf{x}_t;\hat{\boldsymbol{\eta}}_m,\hat{\mathbf{R}}_m\right)}{\sum_{m=1}^{M}\hat{w}_m\Phi\left(\mathbf{x}_t;\hat{\boldsymbol{\eta}}_m,\hat{\mathbf{R}}_m\right)}. \tag{2.25}$$

Using (2.12) and (2.22), (2.19) can be rewritten as

$$Q\left(\boldsymbol{\rho}\left(\mathbf{B},\boldsymbol{\theta}^{(\mathbf{s})}\right),\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)=-\frac{1}{T}\sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_{t,m}\log\left(w_m\Phi\left(\mathbf{x}_t;\mathbf{B}^{-1}\boldsymbol{\mu}_m,\mathbf{B}^{-1}\mathbf{C}_m\mathbf{B}^{-H}\right)\right). \tag{2.26}$$

According to (2.26), one can notice that in order to estimate $\mathbf{B}$ a structure on $\boldsymbol{\eta}_m=\mathbf{B}^{-1}\boldsymbol{\mu}_m$ and $\mathbf{R}_m=\mathbf{B}^{-1}\mathbf{C}_m\mathbf{B}^{-\mathrm{T}}$ is imposed. Since $\left\{\hat{\boldsymbol{\eta}}_m\right\}_{m=1}^{M}$ and $\left\{\hat{\mathbf{R}}_m\right\}_{m=1}^{M}$ in $\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}$ are estimated without this impose there are no $\left\{\mathbf{B},\left\{\boldsymbol{\mu}_m,\mathbf{C}_m\right\}_{m=1}^{M}\right\}$ such that $\hat{\boldsymbol{\eta}}_m=\mathbf{B}^{-1}\boldsymbol{\mu}_m$ and $\hat{\mathbf{R}}_m=\mathbf{B}^{-1}\mathbf{C}_m\mathbf{B}^{-H}$. Therefore, the minimum of

$Q\left(\boldsymbol{\rho}\left(\mathbf{B},\boldsymbol{\theta}^{(\mathbf{s})}\right),\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)$ w.r.t. $\left\{\mathbf{B},\left\{\boldsymbol{\mu}_m,\mathbf{C}_m\right\}_{m=1}^{M}\right\}$ is not strictly attained. It is noted that in the asymptotic case, where $\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}=\boldsymbol{\theta}^{(\mathbf{x})}$ this minimum is attained in a strict manner.

Since $Q\left(\boldsymbol{\rho}\left(\mathbf{B},\boldsymbol{\theta}^{(\mathbf{s})}\right),\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)$ is minimized for $\boldsymbol{\rho}\left(\mathbf{B},\boldsymbol{\theta}^{(\mathbf{s})}\right)=\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}$, it is obvious that

$$\begin{aligned}\min_{\{w_m\}_{m=1}^{M}} Q\left(\boldsymbol{\rho}\left(\mathbf{B},\boldsymbol{\theta}^{(\mathbf{s})}\right),\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)&=-\frac{1}{T}\sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_{t,m}\log\left(\hat{w}_m\Phi\left(\mathbf{x}_t;\mathbf{B}^{-1}\boldsymbol{\mu}_m,\mathbf{B}^{-1}\mathbf{C}_m\mathbf{B}^{-H}\right)\right)\\&=Q'\left(\boldsymbol{\rho}\left(\mathbf{B},\boldsymbol{\theta}^{(\mathbf{s})}\right),\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right).\end{aligned}\tag{2.27}$$

In Appendix B, it is shown that

$$Q'\left(\boldsymbol{\rho}\left(\mathbf{B},\boldsymbol{\theta}^{(\mathbf{s})}\right),\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)=\sum_{m=1}^{M}\hat{w}_m\left\{KL_{norm}\left[\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\mid\mathbf{C}_m\right]+\left[\left(\boldsymbol{\mu}_m-\mathbf{B}\hat{\boldsymbol{\eta}}_m\right)^H\mathbf{C}_m^{-1}\left(\boldsymbol{\mu}_m-\mathbf{B}\hat{\boldsymbol{\eta}}_m\right)\right]\right\}+const.\tag{2.28}$$

The term $KL_{norm}\left(\boldsymbol{\Sigma}_1\mid\boldsymbol{\Sigma}_2\right)$ is the Kullback-Leibler divergence [20] of $N^c\left(\mathbf{0},\boldsymbol{\Sigma}_2\right)$ from $N^c\left(\mathbf{0},\boldsymbol{\Sigma}_1\right)$, where $N^c\left(.,.\right)$ denotes the proper complex Gaussian PDF. The minimum of (2.28) w.r.t. $\boldsymbol{\mu}_m$ is obtained by setting $\boldsymbol{\mu}_m=\mathbf{B}\hat{\boldsymbol{\eta}}_m$. Therefore, (2.28) can be reduced to the following form

$$Q''\left(\mathbf{B},\left\{\mathbf{C}_m\right\}_{m=1}^{M},\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)=\sum_{m=1}^{M}\hat{w}_m KL_{norm}\left[\underbrace{\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H}_{\substack{\text{positive}\\\text{semi-definite}}}\mid\underbrace{\mathbf{C}_m}_{\text{diagonal}}\right].\tag{2.29}$$

The Pythagorean property of the Kullback-Leibler divergence [2], [15] implies that (2.29) can be decomposed in the following manner

$$Q''\left(\mathbf{B},\left\{\mathbf{C}_m\right\}_{m=1}^{M},\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)=\sum_{m=1}^{M}\hat{w}_m\left\{KL_{norm}\left[\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\mid\mathrm{DIAG}\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right)\right]+KL_{norm}\left[\mathrm{DIAG}\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right)\mid\mathbf{C}_m\right]\right\},\tag{2.30}$$

where $\mathrm{DIAG}\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right)$ denotes a diagonal matrix with the same diagonal elements of $\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H$. Thus, the objective function (2.30) is minimized for a fixed value of $\mathbf{B}$ when $\mathbf{C}_m=\mathrm{DIAG}\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right)$ and the attained minimum is

$$Q^*\left(\mathbf{B},\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)=\sum_{m=1}^{M}\hat{w}_m KL_{norm}\left[\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\mid\mathrm{DIAG}\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right)\right].\tag{2.31}$$

Therefore, we conclude, that under the GMM assumption, the objective function Q^* measures the deviation of $\left\{\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right\}_{m=1}^M$ from diagonality. The same result was obtained in [2] for the case of a "block-Gaussian" model. According to this model the observation signals are partitioned into M consecutive quasi-stationary segments, where the relative proportion and covariance matrix of the m^{th} quasi-stationary segment are denoted by $\hat{w}_m$ and $\hat{\mathbf{R}}_m$, respectively.

In Appendix C, it is shown that the objective function (2.31) can be rewritten as follows

$$Q^*\left(\mathbf{B},\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)=\sum_{m=1}^{M}\hat{w}_m\left[\log\left(\det\left(\mathrm{DIAG}\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right)\right)\right)-\log\left(\det\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right)\right)\right], \tag{2.32}$$

where $\det(\cdot)$ denotes the determinant operator. In the following, minimization of Q^* via joint diagonalization with nonorthogonal and orthogonal transformations is described.

2.2.2 ESTIMATION OF THE SEPARATION MATRIX VIA JOINT DIAGONALIZATION WITH NONORTHOGONAL TRANSFORMATIONS

The minimum of $Q^*\left(\mathbf{B},\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)$ is attained for a matrix **B** which jointly diagonalizes the estimated GMM covariance matrices. In this work, two nonorthogonal approximate joint diagonalization algorithms which minimize $Q^*\left(\mathbf{B},\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)$ w.r.t. **B** are evaluated. The first algorithm, offered by Pham [2], [25], is similar to the Jacobi method [22] and applies successive transformations on each pair of distinct rows of **B**, with the exception that the rows of **B** are not constrained to be orthogonal. BSS using Pham's algorithm is denoted as the GMMPHAM algorithm. The second algorithm is a novel technique for approximate joint diagonalization, offered in this work and denoted by the SVDJD algorithm. According to this method, the columns of **B** are estimated separately using iterative singular value decompositions of a weighted sum of the matrices to be diagonalized. Full description of this algorithm is given in Chapter 3. BSS using the SVDJD method is denoted as the GMMSVDJD algorithm.

In summary, solution of the BSS problem via nonorthogonal joint diagonalization comprises the following steps:

1) Estimate the distribution parameters of the observation signals via the EM algorithm for GMM parameter estimation [11]. If the GMM order is unknown, it may be determined using BIC [17], MDL

[18] or AIC [19].

2) Estimate **B** by applying nonorthogonal joint diagonalization algorithms on the estimated GMM covariance matrices.

The following example illustrates the step by step implementation of the algorithm. Twenty five hundred samples of two source signals were synthesized by the following GMM PDF: $f_{\mathbf{s}}\left(\mathbf{s}_t;\boldsymbol{\theta}^{(\mathbf{s})}\right)=\sum_{m=1}^{4} w_m \Phi\left(\mathbf{s}_t;\boldsymbol{\mu}_m;\mathbf{C}_m\right)$, where the univariate GMM order of each source was 2. The values of the mixing proportions, mean vectors and covariance matrices were: $w_1=0.06$, $w_2=0.14$, $w_3=0.24$, $w_4=0.56$, $\boldsymbol{\mu}_1=\left[-5,-5\right]^{\mathbf{T}}$, $\boldsymbol{\mu}_2=\left[-5,5\right]^{\mathbf{T}}$, $\boldsymbol{\mu}_3=\left[5,-5\right]^{\mathbf{T}}$, $\boldsymbol{\mu}_4=\left[5,5\right]^{\mathbf{T}}$, $\mathbf{C}_1=diag\left(10,1\right)$, $\mathbf{C}_2=diag\left(10,12\right)$, $\mathbf{C}_3=\left(3,1\right)$ and $\mathbf{C}_4=diag\left(3,12\right)$, respectively. The scatter plot of the independent source signals is depicted in Fig. 3.a. The source signals were mixed by $\mathbf{A}=\begin{bmatrix}5 & 3\\ -7 & 2\end{bmatrix}$, according to (2.1). Following the first step of the algorithm, the distribution parameters of the mixed source signals were estimated by applying the greedy EM algorithm for GMM parameter estimation [12], where the GMM order was set to 4. The estimated mean vectors and covariance matrices of the mixed sources are depicted in Fig. 3.b on top of the scatter plot of the observations. Following the second step of the algorithm, the separation matrix was estimated by applying the joint diagonalization algorithm, offered by Pham [2], [25], on the estimated GMM covariance matrices. The resulting estimated separation matrix was $\hat{\mathbf{B}}=\begin{bmatrix}0.804 & 0.573\\ -0.553 & 0.818\end{bmatrix}$. The source signals were estimated according to (2.2) and their scatter plot is depicted in Fig. 3.c. One can observe that due to the scaling and permutation ambiguities of the BSS problem, the estimated source signals are scaled and permuted.

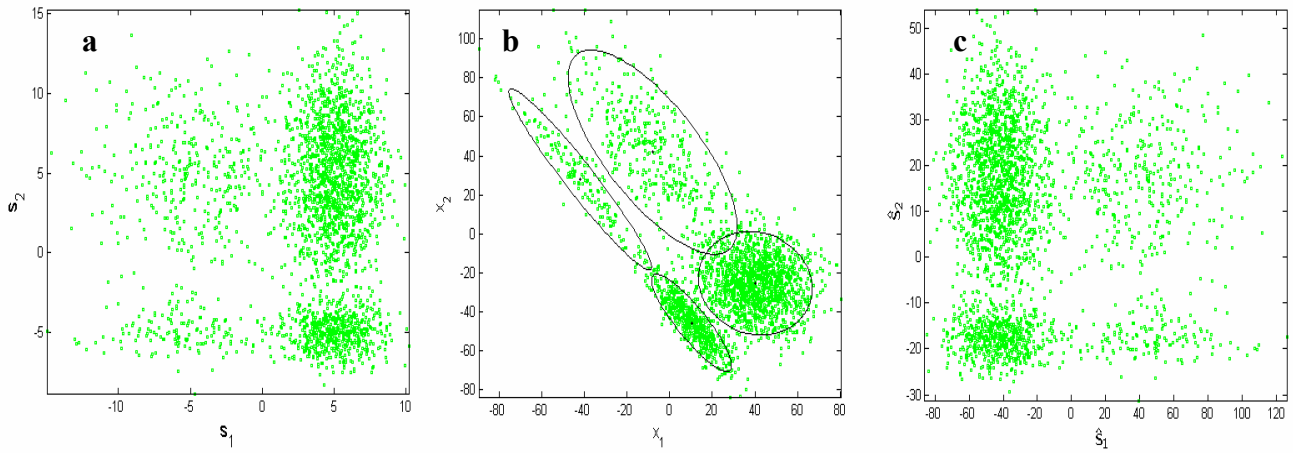

Fig. 3. a) Scatter plot of the source signals. b) Scatter plot of the mixed sources. The ellipses represent the estimated covariance matrices. c) Scatter plot of the estimated source signals.

2.2.3 ESTIMATION OF THE SEPARATION MATRIX VIA JOINT DIAGONALIZATION WITH ORTHOGONAL TRANSFORMATIONS

Under the assumption of white source signals (i.e. $\text{cov}(\mathbf{s}_t) = \mathbf{C} = \mathbf{I} \quad \forall\ t = 1,...,T$), this method implies prewhitening of the observation signals, so their generative model, described in Subsection 2.1.2, is extended. According to this extended model, $\mathbf{x}_t$ is prewhitened by a spatial whitening matrix, $\mathbf{W}$, and $\mathbf{z}_t = \mathbf{W}\mathbf{x}_t$ is formed. Thus, under the assumption of equal number of sensors and sources (L=K), it is implied by (2.1) and (2.2) that

$$\mathbf{z}_t = \mathbf{W}\mathbf{x}_t = \mathbf{W}\mathbf{A}\mathbf{s}_t = \tilde{\mathbf{A}}\mathbf{s}_t . \tag{2.33}$$

Therefore, according to (2.2) and (2.33)

$$\hat{\mathbf{s}}_t = \mathbf{B}\mathbf{x}_t = \mathbf{B}\mathbf{W}^{-1}\mathbf{z}_t = \tilde{\mathbf{B}}\mathbf{z}_t . \tag{2.34}$$

Under the conditions mentioned above, it is shown that $\tilde{\mathbf{B}}$ is a rotation matrix.

Claim 2.1:

Let $\text{cov}(\mathbf{s}_t) = \mathbf{C} = \mathbf{I} \quad \forall\ t = 1,...,T$, *then prewhitening of the observation signals implies that the separation matrix* $\tilde{\mathbf{B}}$ *is a rotation matrix.*

Proof 2.1:

The singular value decomposition of $\mathbf{A}$ *is given by*

$$\mathbf{A} = \underbrace{\mathbf{U}}_{orthonormal} \cdot \underbrace{\mathbf{D}}_{diagonal} \cdot \underbrace{\mathbf{V}^H}_{orthonormal} \tag{2.35}$$

According to (2.1), the covariance matrix of an observation vector at time instance t, is

$$\mathrm{cov}(\mathbf{x}_t) = \mathbf{A}\mathbf{C}\mathbf{A}^H = \boldsymbol{\psi}. \tag{2.36}$$

Substituting (2.34) into (2.36) implies that $\boldsymbol{\psi} = \mathbf{U}\mathbf{D}^2\mathbf{U}^H$, *so the whitening matrix of* $\mathbf{x}_t$ *is given by*

$$\mathbf{W} = \mathbf{D}^{-1}\mathbf{U}^H. \tag{2.37}$$

Thus, it is implied by (2.33), (2.35) and (2.37) that

$$\mathbf{z}_t = \mathbf{W}\mathbf{A}\mathbf{s}_t = \mathbf{V}^H\mathbf{s}_t. \tag{2.38}$$

Therefore, according to (2.34) and (2.38)

$$\tilde{\mathbf{B}} = \mathbf{V}, \tag{2.39}$$

where $\mathbf{V}$ *is a rotation matrix* □

It is implied by (2.33) that by replacing $\mathbf{A}$ with $\tilde{\mathbf{A}} = \mathbf{W}\mathbf{A}$, the generation process of the prewhitened observation signals can be embedded in the generative model described in Subsection 2.1.2. Hence, it can be shown in the same manner described in Subsection 2.1.2 that the joint PDF of the pre-whitened observation signals is also GMM with nondiagonal covariance matrices, as expressed below.

$$f_{\mathbf{z}}\left(\mathbf{z}_t; \boldsymbol{\theta}^{(\mathbf{z})}\right) = \sum_{m=1}^{M} w_m \Phi\left(\mathbf{z}_t; \boldsymbol{\eta}_m, \mathbf{R}_m\right). \tag{2.40}$$

The vector of unknown distribution parameters of the prewhitened observation signals is denoted by

$$\boldsymbol{\theta}^{(\mathbf{z})} = \left\{\boldsymbol{\theta}_m^{(\mathbf{z})}\right\}_{m=1}^{M} = \left\{w_m, \boldsymbol{\eta}_m, \mathbf{R}_m\right\}_{m=1}^{M}, \tag{2.41}$$

where $\tilde{\mathbf{A}}\boldsymbol{\mu}_m = \boldsymbol{\eta}_m$ and $\tilde{\mathbf{A}}\mathbf{C}_m\tilde{\mathbf{A}}^H = \mathbf{R}_m$.

Hence, $Q^*\left(\tilde{\mathbf{B}}, \hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{z})}\right)$ is derived in the same manner as $Q^*\left(\mathbf{B}, \hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)$ in (2.32), such that

$$Q^*\left(\tilde{\mathbf{B}},\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{z})}\right)=\sum_{m=1}^{M}\hat{w}_m\left[\log\left(\det\left(\mathrm{DIAG}\left(\tilde{\mathbf{B}}\hat{\mathbf{R}}_m\tilde{\mathbf{B}}^H\right)\right)\right)-\log\left(\det\left(\tilde{\mathbf{B}}\hat{\mathbf{R}}_m\tilde{\mathbf{B}}^H\right)\right)\right]. \tag{2.42}$$

The mixing proportions and covariance matrices of the prewhitened observation signals, estimated in the final step of the EM algorithm for GMM parameter estimation [11], are denoted by $\left\{\hat{w}_m\right\}_{m=1}^{M}$ and $\left\{\hat{\mathbf{R}}_m\right\}_{m=1}^{M}$, respectively.

The minimum of $Q^*\left(\tilde{\mathbf{B}},\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{z})}\right)$ is attained for a unitary matrix $\tilde{\mathbf{B}}$, which jointly diagonalizes the estimated GMM covariance matrices of the pre-whitened sensor signals. An approximate joint diagonalization algorithm, offered by Flury and Gautschi [13], which minimizes (2.42) w.r.t. $\tilde{\mathbf{B}}$, is applied in order to estimate the separation matrix. In similar to the Jacobi method [22], this algorithm applies successive orthogonal rotations on each pair of distinct columns of $\tilde{\mathbf{B}}$, where $\tilde{\mathbf{B}}$ is constrained to be unitary. BSS using the FG method is denoted as the GMMFG algorithm.

In summary, solution of the BSS problem via orthogonal joint diagonalization comprises the following steps:

1) Estimate the observations spatial whitening matrix $\hat{\mathbf{W}}$.
2) Prewhiten the observation signals according to (2.33).
3) Estimate the distribution parameters of the pre-whitened observation signals via the EM algorithm for GMM parameter estimation [11]. If the GMM order is unknown, it may be determined using BIC [17], MDL [18] or AIC [19].
4) Estimate $\tilde{\mathbf{B}}$ by applying the joint diagonalization algorithm, offered by Flury and Gautschi [13], on the estimated GMM covariance matrices.
5) Estimate **B** according to (2.34) by applying $\hat{\mathbf{B}}=\hat{\tilde{\mathbf{B}}}\hat{\mathbf{W}}$.

2.3 MORE SENSORS THAN SOURCES

In this section, the case of nonsquare mixing matrix **A**, i.e. the case in which the number of rows, denoted by *L*, is greater than the number of columns, denoted by *K*, is considered. The need for dimension reduction of the observation signals is arisen by the fact that the rank of the $L\times L$ covariance matrix of the observation signals, denoted by **R**, is *K*. The dimension reduction procedure is performed in the following steps:

1) Apply SVD of $\mathbf{R}$, i.e. $\mathbf{R} = \underbrace{\mathbf{U}_{\mathbf{R}}}_{\text{orthonormal}} \cdot \underbrace{\mathbf{S}_{\mathbf{R}}}_{\text{diagonal}} \cdot \underbrace{\mathbf{U}_{\mathbf{R}}^{H}}_{\text{orthonormal}}$.

2) Delete the columns of $\mathbf{U}_{\mathbf{R}}$, which correspond to zero diagonal elements of $\mathbf{S}_{\mathbf{R}}$, and create $\widehat{\mathbf{U}}_{\mathbf{R}}$ $(L \times K)$.

3) Apply the linear transformation of $\tilde{\mathbf{x}}_t = \widehat{\mathbf{U}}_{\mathbf{R}}^{H} \mathbf{x}_t \quad \forall\ t = 1,...,T$.

After applying these steps, the rank of $\widehat{\mathbf{R}} = \mathrm{cov}(\tilde{\mathbf{x}}_t)$ is full ($rank(\widehat{\mathbf{R}}) = K$). Due to dimension reduction, the generative model of the observation signals, described in Subsection 2.1.2, is extended. According to this extended model, $\mathbf{x}_t$ is left multiplied by an orthonormal $K \times L$ decorrelation matrix, $\widehat{\mathbf{U}}_{\mathbf{R}}^{H}$, such that

$$\tilde{\mathbf{x}}_t = \widehat{\mathbf{U}}_{\mathbf{R}}^{H} \mathbf{x}_t = \widehat{\mathbf{U}}_{\mathbf{R}}^{H} \mathbf{A} \mathbf{s}_t = \widehat{\mathbf{A}} \mathbf{s}_t \,, \tag{2.43}$$

where $\widehat{\mathbf{A}}$ is a $K \times K$ matrix. Therefore, according to (2.2) and (2.43)

$$\hat{\mathbf{s}}_t = \widehat{\mathbf{A}}^{-1} \tilde{\mathbf{x}}_t = \widehat{\mathbf{B}} \tilde{\mathbf{x}}_t \,. \tag{2.44}$$

It is implied by (2.43) that by replacing $\mathbf{A}$ with $\widehat{\mathbf{A}} = \widehat{\mathbf{U}}_{\mathbf{R}}^{H} \mathbf{A}$, the generation process of $\tilde{\mathbf{x}}_t$ can be embedded in the generative model, described in Subsection 2.1.2. Hence, in the same manner described in Section 2.1.2, it can be shown that the PDF of $\tilde{\mathbf{x}}_t$ is GMM with nondiagonal covariance matrices, as expressed by

$$f_{\tilde{\mathbf{x}}_t}\left(\tilde{\mathbf{x}}_t; \boldsymbol{\theta}^{(\tilde{\mathbf{x}}_t)}\right) = \sum_{m=1}^{M} w_m \Phi\left(\tilde{\mathbf{x}}_t; \boldsymbol{\eta}_m, \mathbf{R}_m\right). \tag{2.45}$$

The vector of unknown distribution parameters of the observation signals after dimension reduction is denoted by $\boldsymbol{\theta}^{(\tilde{\mathbf{x}}_t)} = \left\{w_m, \boldsymbol{\eta}_m, \mathbf{R}_m\right\}_{m=1}^{M}$, where $\widehat{\mathbf{A}}\boldsymbol{\mu}_m = \boldsymbol{\eta}_m$ and $\widehat{\mathbf{A}}\mathbf{C}_m\widehat{\mathbf{A}}^{H} = \mathbf{R}_m$. Hence, $Q^*(\widehat{\mathbf{B}})$ is derived in the same manner as $Q^*(\mathbf{B})$, such that

$$Q^*\left(\widehat{\mathbf{B}}, \hat{\boldsymbol{\theta}}_{ML}^{(\tilde{\mathbf{x}})}\right) = \sum_{m=1}^{M} \hat{w}_m \left[\log\left(\det\left(\mathrm{DIAG}\left(\widehat{\mathbf{B}}\hat{\mathbf{R}}_m\widehat{\mathbf{B}}^{H}\right)\right)\right) - \log\left(\det\left(\widehat{\mathbf{B}}\hat{\mathbf{R}}_m\widehat{\mathbf{B}}^{H}\right)\right)\right]. \tag{2.46}$$

The estimated mixing proportions and covariance matrices of the GMM of $\tilde{\mathbf{x}}_t$ are denoted by $\left\{\hat{w}_m\right\}_{m=1}^{M}$ and $\left\{\hat{\mathbf{R}}_m\right\}_{m=1}^{M}$, respectively. Hence, estimation of $\widehat{\mathbf{B}}$ can be performed by utilizing the GMMJD or GMMFG algorithms, described above. According to (2.44) $\hat{\mathbf{s}}_t = \widehat{\mathbf{B}}\tilde{\mathbf{x}}_t$. Since $\tilde{\mathbf{x}}_t = \widehat{\mathbf{U}}_{\mathbf{R}}^{H}\mathbf{x}_t$, then $\hat{\mathbf{s}}_t = \widehat{\mathbf{B}}\widehat{\mathbf{U}}_{\mathbf{R}}^{H}\mathbf{x}_t = \mathbf{B}\mathbf{x}_t$. Therefore, estimation of $\mathbf{B}$ can be performed in the following manner:

$$\hat{\mathbf{B}} = \hat{\hat{\mathbf{B}}}\widehat{\mathbf{U}}_{\mathbf{R}}^{H}. \tag{2.47}$$

2.4 SIMULATIONS

In this section, the separation performances of the GMMSVDJD, GMMPHAM, GMMFG, NIFA [5], JADE [3] and FastICA [16] algorithms are evaluated and compared by means of interference-to-signal ratio (ISR). Calculation of ISR is detailed in Appendix D.

This section is organized as follows. In Subsection 2.4.1, the expected separation performances of the tested algorithms are evaluated using synthetic data versus skewness level of the sources, rotation angle of a unitary mixing matrix, source statistical distribution class, sample size, number of sources, and SNR level of the sensors. In each test, the expected separation performances are measured via the averaged ISR, obtained by averaging ISR values corresponding to the same set of mixtures. In Subsection 2.4.2, the performances of the compared algorithms in separating mixtures of real speech signals are evaluated.

The compared algorithms were operated under the following overall settings: 1) GMM parameter estimation in the GMMSVDJD, GMMPHAM and GMMFG algorithms was performed via the greedy EM algorithm for GMM parameter estimation [12]. In the greedy approach the high dependence of the EM algorithm on initialization is overcome by optimal insertion of mixture components one after another. The number of EM iterations was set to 100; 2) Separation performance of the NIFA algorithms [5] was evaluated with 100 EM iterations. In each maximization step, 100 iterations with learning rate factor (LRF) of 0.005 were used for updating the source distribution parameters and the separation matrix coefficients; 3) A kurtosis-based contrast function was used in the FastICA algorithm [16].

2.4.1 SYNTHETIC DATA

2.4.1.1 EXPECTED SEPARATION PERFORMANCES AS A FUNCTION OF SKEWNESS LEVEL

The following trial compared the expected separation performances of the tested algorithms as a function of skewness levels of the source signals. Two source signals were synthesized by the following GMM PDF: $f_{\mathbf{s}}\left(\mathbf{s}_t;\boldsymbol{\theta}^{(\mathbf{s})}\right)=\sum_{m=1}^{3} w_m \Phi\left(\mathbf{s}_t;\boldsymbol{\mu}_m,\mathbf{C}_m\right)$, where the univariate GMM orders of the first and second sources were 1

and 3, respectively. The values of the mixing proportions, mean vectors and covariance matrices were: $w_1 = \frac{1}{3} + p$, $w_2 = \frac{1}{3}$, $w_3 = \frac{1}{3} - p$, $\boldsymbol{\mu}_1 = [0, -15]^{\mathbf{T}}$, $\boldsymbol{\mu}_2 = [0, 0]^{\mathbf{T}}$, $\boldsymbol{\mu}_3 = [0, 15]^{\mathbf{T}}$, $\mathbf{C}_1 = diag(5, 1)$, $\mathbf{C}_2 = diag(5, 7)$, and $\mathbf{C}_3 = diag(5, 10)$, respectively. According to the GMM parameters, one can notice that the first source is Gaussian and the second source is non-Gaussian.

The skewness [23] level was controlled by adjusting the value of $p \in [0, 0.25]$ according to the following formula:

$$\gamma_1 = \frac{m_3^{(p)} - 3m_1^{(p)}m_2^{(p)} + \left(m_1^{(p)}\right)^3}{\left(m_2^{(p)} - \left(m_1^{(p)}\right)^2\right)^{3/2}} = \frac{-54000p^3 - 810p^2 - 6795p + 135}{\left(-900p^2 - 9p + 156\right)^{3/2}}, \tag{2.48}$$

where $m_n^{(p)}$ denotes the n^{th} moment of the PDF of the non-Gaussian source signal, as a function of the adjustment parameter p. For each skewness level, which was controlled by adjusting the value of $p \in [0, 0.25]$, 1000 sets of source signals, containing T=1000 samples each, were synthesized and mixed by a random 2×2 mixing matrix, with elements drawn from the real standard normal distribution. A scatter of an arbitrary realization of mixed sources with skewness level of -0.9 is depicted in Fig. 4.a.

The compared algorithms were operated under the following settings: 1) The GMM order in the GMMJD and GMMFG algorithms was set to 3; 2) The PDFs of each unobserved source signal was modeled in the NIFA algorithm by univariate GMMs of order 1 and 3.

Fig. 4.b depicts the averaged ISR of each algorithm versus the skewness level. It can be seen that the performances of the JADE and FastICA algorithms are skewness-dependent due to the fact that the third order cumulant (i.e. skewness) is not considered by these methods. In contrast, the GMMSVDJD, GMMPHAM, GMMFG and NIFA algorithms, which apply flexible source distribution modeling, are skewness-independent.

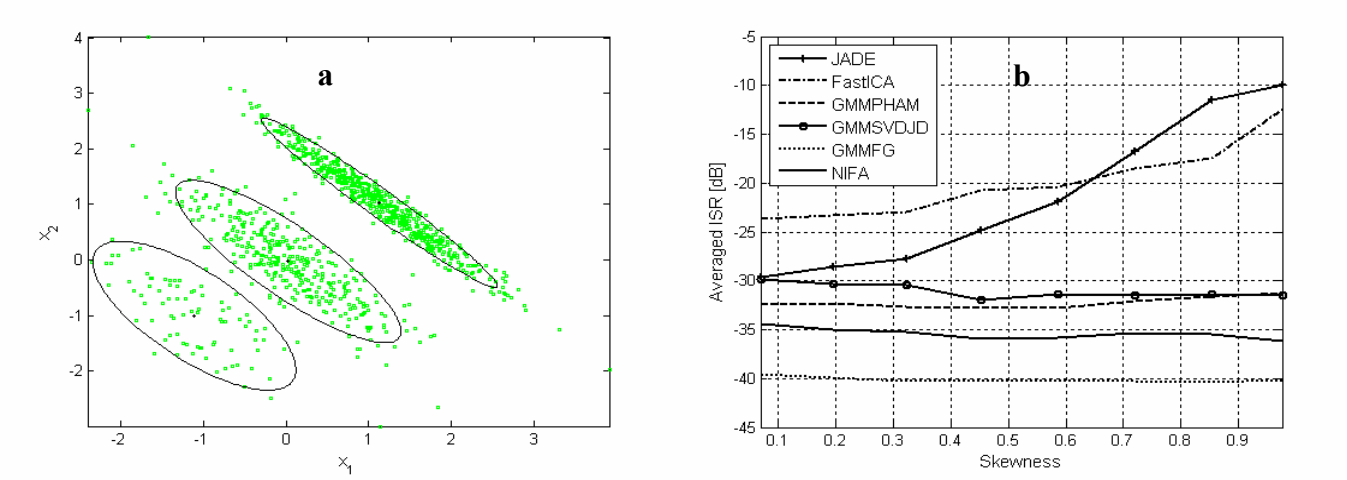

Fig. 4. a) Scatter plot of an arbitrary realization of mixed sources with skewness level of -0.9. The ellipses represent the estimated covariance matrices. b) The averaged ISR of the JADE, FastICA, GMMPHAM, GMMSVDJD, GMMFG and NIFA algorithms versus skewness level.

2.4.1.2 EXPECTED SEPARATION PERFORMANCES AS A FUNCTION OF ROTATION ANGLE

The following trial compared the expected separation performances of the tested algorithms as a function of the rotation angle of an orthonormal mixing matrix. Two source signals were synthesized by the same GMM used in the first trial, where the value p was set to $1/6$. The compared algorithms were operated under the same settings of the first trial. For each rotation angle $\phi \in [0,180^\circ]$, 1000 sets of source signals, containing T=1000 samples each were synthesized and mixed by the rotation matrix $\mathbf{A} = \begin{bmatrix} \cos\phi & \sin\phi \\ -\sin\phi & \cos\phi \end{bmatrix}$.

Fig. 5 depicts the behavior of the averaged ISR of each algorithm versus the rotation angle. In contrast to the JADE, FastICA, GMMSVDJD, GMMPHAM and GMMFG algorithms, the performance of the NIFA algorithm is rotation-dependent due to its incapability of automatically adapting the GMM order of each source to the rotation angle.

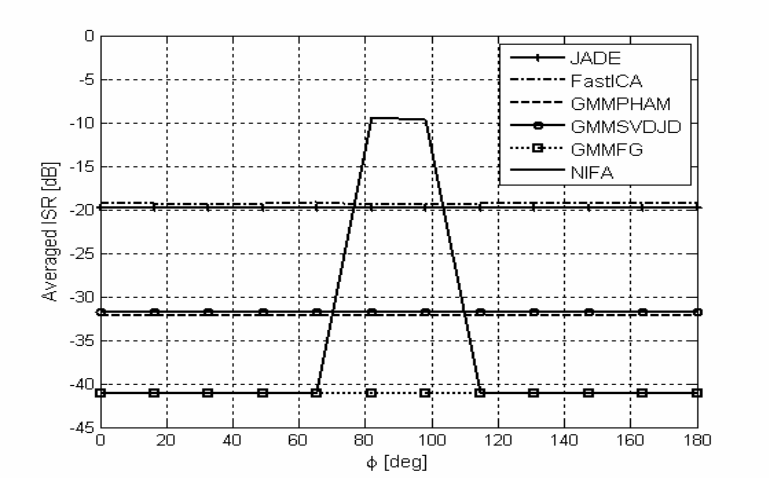

Fig. 5. The averaged ISR of the tested algorithms as a function of rotation angle.

2.4.1.3 EXPECTED SEPARATION PERFORMANCES AS A FUNCTION OF STATISTICAL DISTRIBUTION CLASS

The following trial compared the expected separation performances of the tested algorithms as a function of the statistical distribution class. The source densities were synthesized by a generalized Gaussian [21], which has the form $f(s) \propto \exp\left(-0.5|s|^{2/(1+\beta)}\right)$. By inferring the shape parameter, β, a wide class of unimodal PDFs can be characterized including uniform, Gaussian, Laplacian, and other sub and super-Gaussian densities. For example, the uniform, normal and Laplacian distributions are derived by choosing for $\beta \to -1$, $\beta = 0$ and $\beta = 1$, respectively.

The compared algorithms were operated under the following settings: 1) The GMM order in the GMMJD and GMMFG algorithms was determined according to BIC [17], where the maximal order allowed was set to 9; 2) The PDFs of each unobserved source signal was modeled in the NIFA algorithm by univariate GMMs of order 3; 3) For $\beta > 0$ (i.e. super-Gaussian densities), the mean vectors estimated by the GMMSVDJD, GMMPHAM, GMMFG and NIFA algorithms were constrained to zero.

For each $\beta \in \{10,9,8,7,6,5,4,3,2,1,0,-0.99\}$, 100 sets of unit variance two-dimensional source signals, containing T=5000 samples each, were synthesized. The sources were mixed by a random 2×2 mixing matrix, with elements drawn from the real standard normal distribution. Fig. 6.a depicts the behavior of the

averaged ISR of the tested algorithms versus β. Fig. 6.b, depicts the averaged number of Gaussians determined by the GMMSVDJD, GMMPHAM and GMMFG algorithms according to the BIC, as a function of β. One can observe that the number of Gaussians increases while β increases from 0 to 10. For $\beta = 0$, the performances of all the compared algorithms are poor, due to the fact that the sources are Gaussian. As β increases from 1 to 10, one can observe that the separation performances of the GMMSVDJD, GMMPHAM, GMMFG, JADE and FastICA algorithms improve, while the GMMJD and GMMFG algorithms outperform the JADE and FastICA algorithms. Regarding the NIFA algorithm, one can observe that its performance deteriorates while $\beta > 3$ and $\beta > 6$ for LRF=0.05 and LRF=0.005, respectively. The reason for that stems from the fact that when β increases, the sources distribution tails become heavier and more Gaussians (as observed in Fig. 6.b) are required for modeling of the probability density of the data. Since the number of Gaussians may be different in each direction and the NIFA algorithm is incapable to adapt the number of Gaussians in each direction, modeling mismatch is caused and performance deterioration is inflicted.

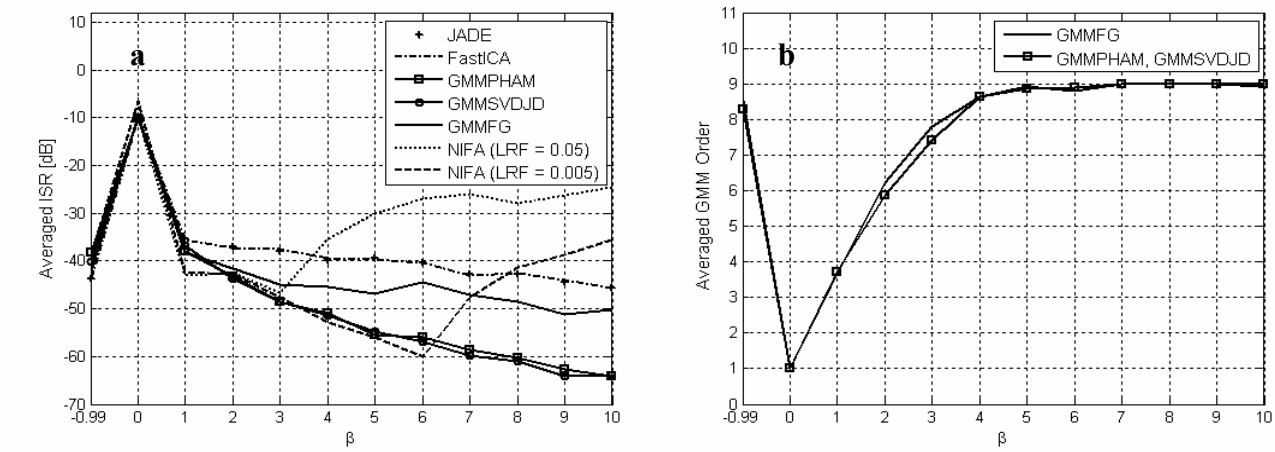

Fig. 6. a) The averaged ISR of the tested algorithms versus the generalized Gaussian shape parameter, β. b) The averaged GMM order, determined by the GMMJD and GMMFG algorithms according to the BIC, versus the generalized Gaussian shape parameter, β.

2.4.1.4 EXPECTED SEPARATION PERFORMANCES AS A FUNCTION OF SAMPLE SIZE

The following trial compared the expected separation performances and the averaged running time of the tested algorithms as a function of the sample size, *T*. For each $T = 50,100,500,1000,3000,5000$, 100 sets of two-dimensional unit variance, independent source signals were synthesized from the generalized Gaussian

density. The first and second sources were synthesized with shape parameter of $\beta = 1$ and $\beta = 10$, respectively. The sources were mixed by a random 2×2 mixing matrix, with elements drawn from the real standard normal distribution.

Fig. 7.a depicts the behavior of the averaged ISR of the tested algorithms versus *T*. Fig. 7.b depicts the averaged number of Gaussians determined by the GMMSVDJD, GMMPHAM and GMMFG algorithms according to the BIC, as a function of *T*. Fig. 7.c, depicts the averaged running time of the tested algorithms versus the sample size. The computer used for the simulations was IBM R-51 laptop computer with Intel centrino^{TM} processor. According to Fig. 7.a, one can observe that as *T* increases from 50 to 5000 the separation performances of the tested algorithms improve, where the GMMSVDJD, GMMPHAM and GMMFG algorithms perform better in comparison to the JADE, FastICA and NIFA algorithms. According to Fig. 7.b, one can observe that the number of Gaussians decreases while *T* decreases from 5000 to 50. This property enables the applicability of the proposed methods for small sample size. According to Fig. 7.c, one can observe that the running time of the NIFA, GMMSVDJD, GMMPHAM and GMMFG algorithms is much higher in comparison to the JADE and FastICA algorithms. This drawback stems from the fact that the NIFA, GMMSVDJD, GMMPHAM and GMMFG methods assume a much more abundant source distribution model. However, the averaged running time of the NIFA, GMMSVDJD, GMMPHAM and GMMFG algorithms does not increase dramatically with the sample size.

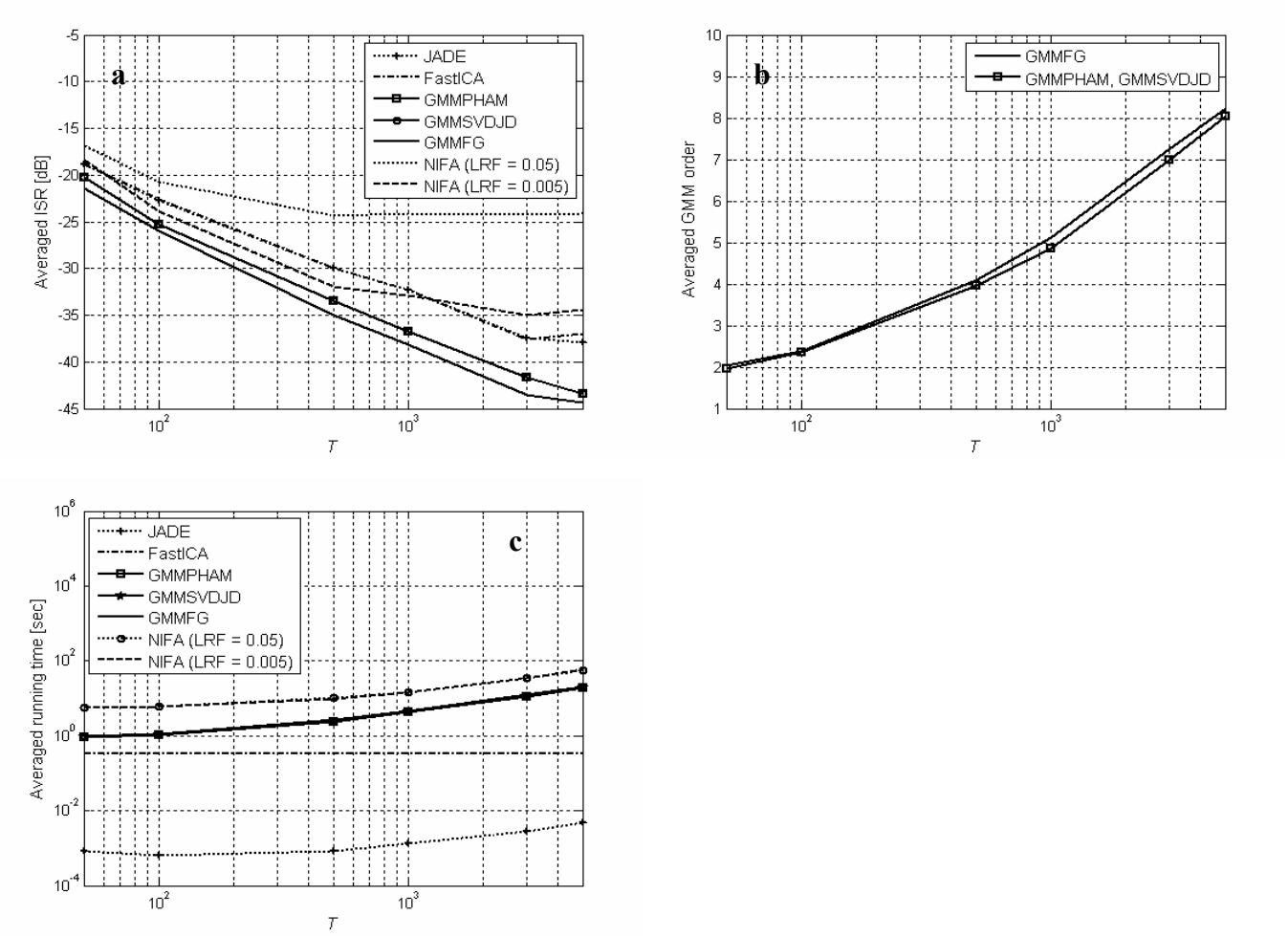

Fig. 7. a) The averaged ISR of the tested algorithms versus the sample size, b) The averaged GMM order, determined by the GMMJD and GMMFG algorithms according to the BIC, versus the sample size, c) The averaged running time of the tested algorithms as a function of the sample size.

2.4.1.5 EXPECTED SEPARATION PERFORMANCES AS A FUNCTION OF DIMENSION

The following trial compared the separation performances and the averaged running time of the tested algorithms versus the number of sources. Let K denote the number of sources. For each $K \in \{2,4,6,8,10\}$ 100 sets of K unit variance source signals, containing T=5000 samples each, were synthesized from the generalized Gaussian density with shape parameter $\beta = 10$. The source signals in each set were mixed by a random $K \times K$ mixing matrix, with elements drawn from the real standard normal distribution.

The compared algorithms were operated under the following settings: 1) The GMM order in the GMMSVDJD, GMMPHAM and GMMFG algorithms was determined according to BIC [17]. The maximal GMM order for two-dimensional mixtures was set to 9. For mixtures with dimension greater than two, the maximal GMM order was set to 30. It is noted that by fixing the maximal GMM order, a model with statistical dependence between the sources might be imposed due to the fact that some of the combinations between the univariate Gaussians are discarded. This fact might theoretically influence (in extreme cases) the results; 2) The mean vectors estimated by the GMMSVDJD, GMMPHAM, GMMFG and NIFA algorithms were constrained to zero; 3) The PDFs of each unobserved source signal was modeled in the NIFA algorithm by a univariate GMM of order 3.

Fig. 8.a depicts the averaged ISR of each algorithm as a function of K. One can observe that the separation performances of the GMMJD and GMMSVD are better in comparison to the JADE, FastICA and NIFA algorithms. Fig. 8.b depicts the averaged number of Gaussians, determined by the GMMSVDJD, GMMPHAM and GMMFG algorithms according to BIC. One can observe that the number of Gaussians increases while the number of sources increases from 2 to 4 and decreases while the number of sources increases from 4 to 10. The reason for the decrease stems from the central limit theorem, according to which the distribution of the sensors becomes more Gaussian when the number of sources increases. Fig. 8.c, depicts the averaged running time of the tested algorithms as a function of K. According to this figure, one can observe that the running time of the NIFA, GMMSVDJD, GMMPHAM and GMMFG algorithms is much higher in comparison to the JADE and FastICA algorithms. However, the averaged running time of the NIFA, GMMSVDJD, GMMPHAM and GMMFG algorithms does not increase dramatically with the number of sources.

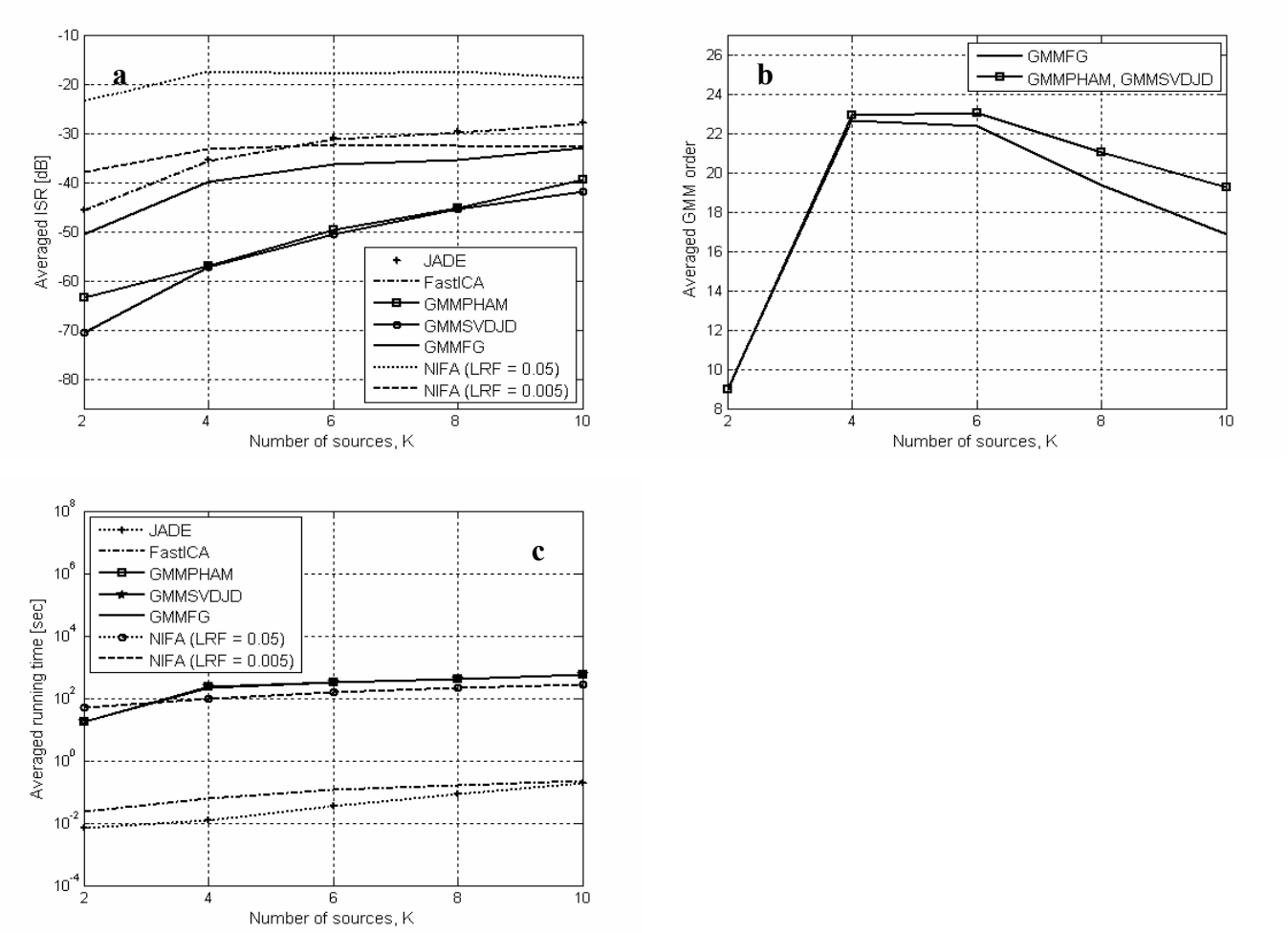

Fig. 8. a) The averaged ISR of tested algorithms as a function of the number of sources. b) The averaged GMM order, determined by the GMMJD and GMMFG algorithms according to the BIC, as a function of the number of source signals. c) The averaged running time of the tested algorithms as a function of the number of sources.

2.4.1.6 EXPECTED SEPARATION PERFORMANCES AS A FUNCTION OF SNR

The following trial compared the separation performances of the tested algorithms in the presence of additive white Gaussian noise. For each SNR, 100 sets of two-dimensional unit variance independent source signals, containing T=5000 samples each, were synthesized from the generalized Gaussian density. The first and second sources were synthesized with shape parameter $\beta = 1$ and $\beta = 10$, respectively. For each set, the observation signals were derived according to the following linear noisy mixing model:

$$\mathbf{x}_t = \mathbf{A}\mathbf{s}_t + \sigma\mathbf{n}_t \qquad t = 1,...,T\,, \tag{2.49}$$

where **A** is a random 2×2 mixing matrix, with elements drawn from the real standard normal distribution and **n** denotes an isotropic zero-mean Gaussian noise with identity covariance matrix. The value σ controlled the SNR levels according to the following formula:

$$\sigma^2 = \frac{1}{K} tr\left(\mathbf{A}\mathbf{A}^{\mathrm{T}}\right)\cdot 10^{-SNR/10}. \tag{2.50}$$

The compared algorithms were operated under the following settings: 1) The GMM order in the GMMSVDJD, GMMPHAM and GMMFG algorithms was determined according to the BIC [17], where the maximal GMM order was set to 9; 2) The mean vectors estimated by the GMMJD, GMMFG and NIFA algorithms were enforced to zero; 3) The PDF of each unobserved source signal was modeled in the NIFA algorithm by a univariate GMM of order 3.

Fig. 9 depicts the behavior of the averaged ISR of the tested algorithms versus SNR. One can observe that the GMMJD and GMMFG algorithms outperform the JADE, FastICA and NIFA algorithms for high SNRs. The performance of the NIFA algorithm for LRF=0.05 and LRF=0.005 deteriorates while the SNR increases from 20 dB and from 30 dB, respectively. The reason for that stems from the incapability of the NIFA algorithm to adapt the number of Gaussians in each direction, which may be different. This incapability causes modeling mismatch which inflicts performance deterioration. For low SNRs the modeling mismatch is obscured by the presence of noise and therefore is not dominant.

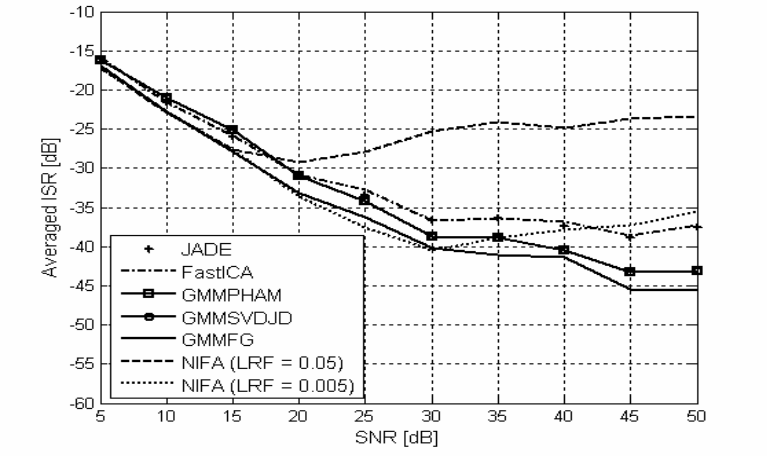

Fig. 9. The averaged ISR of the tested algorithms versus SNR.

2.4.2 REAL DATA

The following trials compare the performances of the tested algorithms, by means of ISR, in separating different mixture combinations of three 10 seconds long speech signals, sampled at 16 kHz and normalized to unit variance. The source signals are denoted by $\mathbf{S}_1$, $\mathbf{S}_2$ and $\mathbf{S}_3$.

The compared algorithms were operated under the following settings: 1) The GMM order in the GMMSVDJD, GMMPHAM and GMMFG algorithms was determined according to BIC [17], where the maximal GMM orders allowed for 2 and 3 dimensions were 9 and 30, respectively; 2) The mean vectors estimated by the GMMSVDJD, GMMPHAM, GMMFG and NIFA algorithms were constrained to zero; 3) The PDFs of each unobserved source signal was modeled in the NIFA algorithm by a univariate GMM of order 3.

In the first trial, $\mathbf{S}_1$ and $\mathbf{S}_2$ were mixed by the mixing matrix $\mathbf{A} = \begin{bmatrix} 5 & 3 \\ -7 & 8 \end{bmatrix}$. The scatter of the mixed sources is depicted in Fig. 9. The optimal GMM order, determined by the GMMSVDJD, GMMPHAM and GMMFG algorithms was 9.

In the second trial, $\mathbf{S}_1$, $\mathbf{S}_2$ and $\mathbf{S}_3$ were mixed by the mixing matrix $\mathbf{A}' = \begin{bmatrix} 5 & 3 & 1 \\ -7 & 8 & 6 \\ 9 & -5 & 4 \end{bmatrix}$. The optimal GMM orders, determined by the GMMSVDJD, GMMPHAM and GMMFG algorithms were 27, 27 and 26, respectively.

In the third trial, the case of more sensors than sources was tested by mixing $\mathbf{S}_1, \mathbf{S}_2$ and $\mathbf{S}_3$ with the mixing matrix $\mathbf{A}'' = \begin{bmatrix} 5 & -7 & 9 & 1 & -8 & -1 & 2 & 5 \\ 3 & 8 & -5 & 2 & 9 & -9 & 1 & -3 \\ 1 & 6 & 4 & -3 & 2 & 2 & 2 & -1 \end{bmatrix}^{\mathrm{T}}$. The optimal GMM orders, determined by the GMMSVDJD, GMMPHAM and GMMFG algorithms were 27, 27 and 26, respectively. Separation performances of the compared algorithms are depicted in Fig.11. One can observe that the best separation performance was achieved by the GMMSVDJD algorithm.

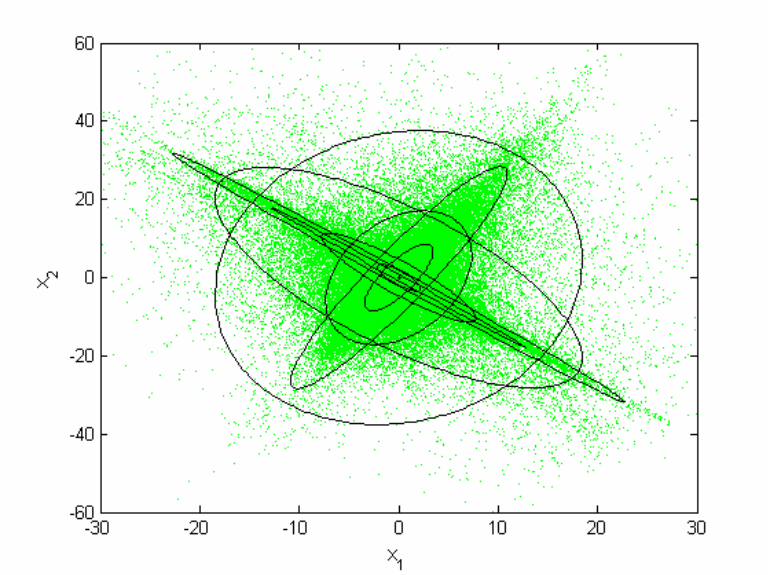

Fig. 10. Scatter plot of the mixture of $\mathbf{S}_1$ and $\mathbf{S}_2$. The ellipses represent the estimated covariance matrices, which assemble the GMM of the observation signals.

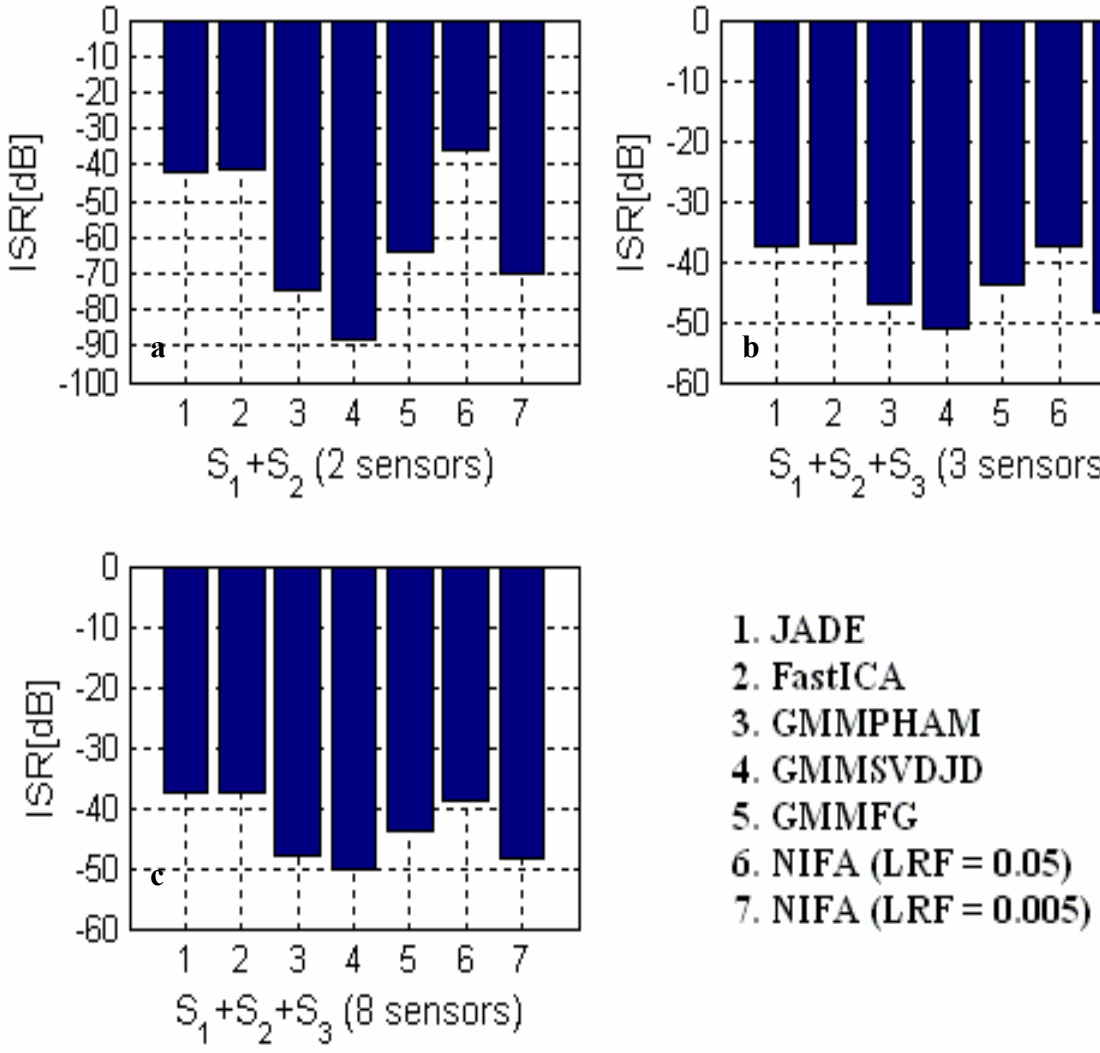

Fig. 11. Separation performances of the tested algorithms in separating a) two-dimensional mixture of two speech signals, b) three dimensional mixture of three speech signals, c) eight-dimensional mixture of three speech signals.

2.5 DISCUSSION AND CONCLUSIONS

Two novel algorithms for BSS of noiseless linear mixture of independent sources are proposed. The algorithms solve the BSS problem by estimation of the sensors distribution parameters via the EM algorithm for GMM parameter estimation, followed by estimation of the separation matrix via approximate joint diagonalization of the estimated GMM covariance matrices.

It was shown that estimation of the sensors distribution parameters via the EM algorithm amounts to obtaining a tight lower bound on the log-likelihood of a function of the separation matrix. It was also shown that joint diagonalization of the estimated GMM covariance matrices amounts to maximization of the obtained tight lower bound w.r.t. the separation matrix.

Generally, the GMMSVDJD and GMMPHAM are preferred upon the GMMFG algorithm since no prewhitening of the observations is performed. However, in cases where the number of observations is small, the use of the GMMFG algorithm is preferred upon the GMMSVDJD and GMMPHAM algorithms, since GMM parameter estimation is less erroneous when prewhitening of the observations is applied.

In comparison to methods like JADE [3] and FastICA [16], which use restrictive assumptions on the sources distribution, the proposed techniques are superior due to the fact that a flexible source density model is applied.

In contrast to our algorithms, the methods described in [4] - [8] utilize an EM algorithm, which jointly estimate the source distribution parameters and the mixing matrix coefficients. This approach has the following disadvantages. First, accurate initialization and order selection of the distribution model of the unobserved source signals is difficult, so the EM algorithm may converge into undesired maxima. Second, implementation of these approaches is complicated.

The learning rate factor in the NIFA algorithm is selected empirically and has a great effect on the separation performance. As observed by simulations, the NIFA algorithm is sensitive to model order selection in each dimension and cannot adapt the number of Gaussians in each direction. This drawback can cause model mismatch, which results in poor separation performance. Since the proposed methods do not estimate the PDF of each unobserved source, they are not affected by this drawback.

In the proposed methods, the distribution parameters of the observations are estimated apart from the separation matrix and therefore, GMM order selection using information theoretic criteria is trivial. In contrast to this, optimal selection of GMM order for each unobserved source in the methods described in [4]-[8] is much more complicated.

Theoretically, the EM algorithm for GMM parameter estimation of the sensor signals would become intractable as the number of sources increases. This is because the number of Gaussians grows exponentially with the number of sources. For example, K=10 sources with n_k=3 Gaussians for each source, result $M=3^{10}$ Gaussians. However, according to the simulation results, it is observed that due to finite sample size, the determined GMM order in high dimensions is always much smaller than the theoretical number of Gaussians. This property enables the applicability of the proposed methods also for large number of sources.

Finally, according to simulation results, the proposed BSS algorithms demonstrate superior separation performances in comparison to existing methods. However, this superiority comes at the expense of higher computational load, caused by the use of the EM algorithm for GMM parameter estimation [11].

3. Fast Approximate Joint Diagonalization of Positive-Definite Hermitian Matrices

Consider a set R of M matrices $\mathbf{R}_1, \mathbf{R}_2, ..., \mathbf{R}_M \in \mathbb{C}^{K \times K}$. The set R is said to be simultaneously diagonalizable if there exists a nonsingular matrix $\mathbf{B} \in \mathbb{C}^{K \times K}$ and M congruent diagonal matrices $\mathbf{\Lambda}_1, \mathbf{\Lambda}_2, ..., \mathbf{\Lambda}_M \in \mathbb{C}^{K \times K}$ such that

$$\mathbf{B}\mathbf{R}_m\mathbf{B}^H = \mathbf{\Lambda}_m \quad \forall\ m = 1, ..., M\ . \tag{3.1}$$

In real world applications the set R is usually unknown and needs to be estimated from a finite sample size. Due to estimation errors, the estimate of R, denoted by $\hat{\mathrm{R}} = \left\{\hat{\mathbf{R}}_m\right\}_{m=1}^{M}$, is perturbed and exact joint diagonalization of $\hat{\mathrm{R}}$ may not be achievable. Hence, the problem of approximate joint diagonalization seeks for a matrix **B** such that

$$\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H = \hat{\mathbf{\Lambda}}_m \tag{3.2}$$

are "as diagonal as possible" in a sense that a deviation measure of $\left\{\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right\}_{m=1}^{M}$ from diagonality is minimized w.r.t. **B**.

In this work, a new efficient iterative algorithm for approximate joint diagonalization of positive-definite Hermitian matrices, named as the SVDJD algorithm is proposed. The positive-definite assumption is motivated by the fact that in many applications [2], [30] R consists of covariance matrices of some random variables. We note that positive-definiteness of the matrices in R implies that $\mathbf{\Lambda}_1, \mathbf{\Lambda}_2, ..., \mathbf{\Lambda}_M \in \mathbb{C}^{K \times K}$. According to the proposed algorithm, a diagonalization matrix **B**, which is not constrained to be orthogonal, is estimated by optimization of a ML-based objective function, also used by Pham [2], [25]. The columns of **B** are estimated separately using iterative SVDs of a weighted sum of the matrices to be diagonalized. This property enables low computational load of $O\left(MK + K^4\right)$ per iteration, which is practical especially in cases of large amount of matrices.

This chapter is organized as follows. In Section 3.1, a ML-based objective function for the estimation of **B** is derived. Section 3.2 presents an iterative minimization algorithm of the objective function w.r.t. **B**. In Section 3.3, the convergence condition of the minimization algorithm is studied. In Section 3.4, the initialization of the proposed algorithm is discussed. In Section 3.5, the computational complexity of the

algorithm is calculated and compared with other existing methods for approximate joint diagonalization. In Section 3.6, the performance of the proposed algorithm is evaluated and compared with other existing techniques for approximate joint diagonalization. Finally, Section 3.7 summarizes the main points of this chapter.

3.1 DERIVATION OF A MAXIMUM LIKELIHOOD BASED OBJECTIVE FUNCTION

In this section, a ML-based objective function for estimation of the diagonalization matrix **B** is derived under the following measurements model. Let $\mathbf{X}_m = \left[\mathbf{x}_1^{(m)},...,\mathbf{x}_{N_m}^{(m)}\right]\left(m=1,...,M\right)$ denote M statistically independent populations of K-variate complex random vectors, where $\mathbf{x}_n^{(m)}$ and N_m denote the n^{th} vector and the sample size of the m^{th} population, respectively. In each population, the vectors $\mathbf{x}_n^{(m)} \overset{\text{i.i.d}}{\sim} N^c\left(\boldsymbol{\eta}_m, \mathbf{R}_m\right)$ $\left(n=1,...,N_m \;;\; m=1,...,M\right)$, where $N^c\left(.,.\right)$ denotes the proper complex Gaussian PDF, $\left\{\boldsymbol{\eta}_m\right\}_{m=1}^M$ denotes the mean vectors and $\left\{\mathbf{R}_m\right\}_{m=1}^M$ denotes a simultaneously diagonalizable family of complex covariance matrices such that (3.1) is satisfied.

Assuming that $\boldsymbol{\eta}_m = \mathbf{0} \;\; \forall \; m=1,...,M$, an unbiased estimate of $\mathbf{R}_m$ is given by

$$\hat{\mathbf{R}}_m = \frac{1}{N_m}\mathbf{X}_m\mathbf{X}_m^H \; . \tag{3.3}$$

The following definition is utilized in order to derive the joint likelihood function of $\mathbf{R}_m$ given $\hat{\mathbf{R}}_m$.

Definition 3.1:

Let $\mathbf{T}_m = \mathbf{X}_m\mathbf{X}_m^H$, where the $K \times N_m$ matrix $\mathbf{X}_m$ is distributed as $\text{vec}\left(\mathbf{X}_m\right) \sim N^c\left(\mathbf{0}, \mathbf{I}_{N_m} \otimes \mathbf{R}_m\right)$. Then $\mathbf{T}_m$ is said to have the complex central Wishart distribution with N_m degrees of freedom and covariance matrix $\mathbf{R}_m$, denoted by $\mathbf{T}_m \sim W_K^c\left(N_m, \mathbf{R}_m\right)$. Let $N_m \geq K$, the PDF of $\mathbf{T}_m = N_m\hat{\mathbf{R}}_m$ is given by [33]

$$f_{\mathbf{T}_m;\mathbf{R}_m}\left(N_m\hat{\mathbf{R}}_m;\mathbf{R}_m\right) = \frac{1}{\Gamma_K^c\left(N_m\right)\left(\det\left(\mathbf{R}_m\right)\right)^{N_m}} etr\left(-N_m\mathbf{R}_m^{-1}\hat{\mathbf{R}}_m\right)\left(\det\left(N_m\hat{\mathbf{R}}_m\right)\right)^{N_m-K}, \tag{3.4}$$

where Γ_K^c denotes the complex multivariate gamma function, "*etr*" is the exponential function of the trace

operator, and $\det(\cdot)$ denotes the determinant operator.

Since $\{\mathbf{X}_m\}_{m=1}^{M}$ are statistically independent, the matrices in $\hat{\mathrm{R}}$ are also statistically independent and therefore, according to (3.4) the joint likelihood function of $\mathrm{R}=$ given $\hat{\mathrm{R}}$ is

$$L(\mathbf{R}_1,...,\mathbf{R}_M) = c\cdot\prod_{m=1}^{M}\left(\det(\mathbf{R}_m)\right)^{-N_m} etr\left(-N_m\mathbf{R}_m^{-1}\hat{\mathbf{R}}_m\right), \tag{3.5}$$

where c denotes a constant. Equation (3.1) is satisfied because $\mathrm{R}=$ is a simultaneously diagonalizable set. Therefore, the joint likelihood of $\mathbf{B},\boldsymbol{\Lambda}_1,\boldsymbol{\Lambda}_2,...,\boldsymbol{\Lambda}_M$ given $\hat{\mathrm{R}}$ is

$$\tilde{L}(\mathbf{B},\boldsymbol{\Lambda}_1,\boldsymbol{\Lambda}_2,...,\boldsymbol{\Lambda}_M) = c\cdot\prod_{m=1}^{M}\left(\det\left(\mathbf{B}^H\boldsymbol{\Lambda}_m^{-1}\mathbf{B}\right)\right)^{N_m} etr\left(-N_m\mathbf{B}^H\boldsymbol{\Lambda}_m^{-1}\mathbf{B}\hat{\mathbf{R}}_m\right). \tag{3.6}$$

and the corresponding normalized joint log-likelihood of $\mathbf{B},\boldsymbol{\Lambda}_1,\boldsymbol{\Lambda}_2,...,\boldsymbol{\Lambda}_M$ given $\hat{\mathrm{R}}$ is

$$\begin{aligned} Q(\mathbf{B},\boldsymbol{\Lambda}_1,\boldsymbol{\Lambda}_2,...,\boldsymbol{\Lambda}_M) &= -\frac{1}{N}\log\tilde{L}(\mathbf{B},\boldsymbol{\Lambda}_1,\boldsymbol{\Lambda}_2,...,\boldsymbol{\Lambda}_M) \\ &= \sum_{m=1}^{M}\hat{w}_m\left[tr\left(\mathbf{B}^H\boldsymbol{\Lambda}_m^{-1}\mathbf{B}\hat{\mathbf{R}}_m\right)-\log\left(\det\left(\mathbf{B}^H\boldsymbol{\Lambda}_m^{-1}\mathbf{B}\right)\right)\right]+\log c, \end{aligned} \tag{3.7}$$

where N denotes the total sample size, such that $N=\sum_{m=1}^{M}N_m$ and $\hat{w}_m=\frac{N_m}{N}$. In Appendix E, it is shown that

$$Q(\mathbf{B},\boldsymbol{\Lambda}_1,\boldsymbol{\Lambda}_2,...,\boldsymbol{\Lambda}_M) = \sum_{m=1}^{M}\hat{w}_m KL_{norm}\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H \mid \boldsymbol{\Lambda}_m\right)+c', \tag{3.8}$$

where c' denotes a constant and $KL_{norm}\left(\boldsymbol{\Sigma}_1\mid\boldsymbol{\Sigma}_2\right)$ is the Kullback-Leibler divergence [20] of $N^c\left(\mathbf{0},\boldsymbol{\Sigma}_2\right)$ from $N^c\left(\mathbf{0},\boldsymbol{\Sigma}_1\right)$.

The Pythagorean property of the Kullback-Leibler divergence [15] implies that (3.9) can be decomposed in the following manner

$$\begin{aligned} &Q(\mathbf{B},\boldsymbol{\Lambda}_1,\boldsymbol{\Lambda}_2,...,\boldsymbol{\Lambda}_M) = \\ &\sum_{m=1}^{M}\hat{w}_m\left[KL_{norm}\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H \mid \mathrm{DIAG}\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right)\right)+KL_{norm}\left(\mathrm{DIAG}\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right)\mid\boldsymbol{\Lambda}_m\right)\right]+c', \end{aligned} \tag{3.9}$$

where $\mathrm{DIAG}\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right)$ denotes a diagonal matrix with the same diagonal elements of $\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H$. Thus, the objective function in (3.9) is minimized for a fixed value of $\mathbf{B}$ when $\boldsymbol{\Lambda}_m=\mathrm{DIAG}\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right)$ and the attained minimum is

$$Q^*(\mathbf{B}) = \sum_{m=1}^{M} \hat{w}_m KL_{norm}\left[\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H \mid \mathrm{DIAG}\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right)\right]. \tag{3.10}$$

Therefore, we conclude that the normalized log-likelihood of $\mathbf{B}, \mathbf{\Lambda}_1, \mathbf{\Lambda}_2, ..., \mathbf{\Lambda}_M$ given $\hat{\mathrm{R}}$, leads to an objective function, which measures the deviation of $\left\{\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right\}_{m=1}^{M}$ from diagonality. In Appendix C, it is shown that the objective function in (3.11) can be expressed as

$$Q^*(\mathbf{B}) = \sum_{m=1}^{M} \hat{w}_m \left[\log\left(\det\left(\mathrm{DIAG}\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right)\right)\right) - \log\left(\det\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right)\right)\right]. \tag{3.11}$$

We note that the same ML-based objective function was obtained in Subsection 2.2.1 in the context of blind separation of statistically independent mixed sources, obeying a Gaussian mixture distribution model [14]. In this case, the EM algorithm [10] for GMM parameter estimation [11] was utilized for the estimation of $\{\hat{w}_m\}_{m=1}^{M}$ and $\left\{\hat{\mathbf{R}}_m\right\}_{m=1}^{M}$, which denote the estimated mixing proportions and covariance matrices of the Gaussians, respectively. The diagonalization matrix $\mathbf{B}$ was physically the separation matrix. The same objective function was also obtained in [2] for the case of blind separation of nonstationary independent sources.

3.2 MINIMIZATION ALGORITHM

In this section, an iterative algorithm for minimization of (3.11) w.r.t. $\mathbf{B}$ is derived. Direct minimization of (3.11) w.r.t $\mathbf{B}$ is analytically cumbersome. Therefore, $\mathbf{B}$ is decomposed into $\mathbf{B} = \breve{\mathbf{B}}\hat{\mathbf{W}}$, where $\hat{\mathbf{W}}$ is a whitening matrix of $\hat{\mathbf{R}} = \sum_{m=1}^{M} \hat{w}_m \hat{\mathbf{R}}_m$, and

$$Q^*(\breve{\mathbf{B}}) = \sum_{m=1}^{M} \hat{w}_m \left[\log\left(\det\left(\mathrm{DIAG}\left(\breve{\mathbf{B}}\left(\hat{\mathbf{W}}\hat{\mathbf{R}}_m\hat{\mathbf{W}}^H\right)\breve{\mathbf{B}}^H\right)\right)\right) - \log\left(\det\left(\breve{\mathbf{B}}\hat{\mathbf{W}}\hat{\mathbf{R}}_m\hat{\mathbf{W}}^H\breve{\mathbf{B}}^H\right)\right)\right] \tag{3.12}$$

is minimized w.r.t. $\breve{\mathbf{B}}$ (calculation of $\hat{\mathbf{W}}$ is described in Appendix H). Since $\hat{\mathbf{W}}$ and $\hat{\mathrm{R}}$ are independent of $\breve{\mathbf{B}}$ (3.12) can be reduced to the following form

$$Q'(\breve{\mathbf{B}}) = \sum_{m=1}^{M} \hat{w}_m \left[\log\left(\det\left(\mathrm{DIAG}\left(\breve{\mathbf{B}}\left(\hat{\mathbf{W}}\hat{\mathbf{R}}_m\hat{\mathbf{W}}^H\right)\breve{\mathbf{B}}^H\right)\right)\right) - \log\left(\det\left(\breve{\mathbf{B}}\breve{\mathbf{B}}^H\right)\right)\right]. \tag{3.13}$$

In Appendix H, it is shown that since $\hat{\mathbf{W}}$ is a whitening matrix of $\hat{\mathbf{R}}$, then $\breve{\mathbf{B}}$ is approximately unitary and thus

$$Q'\left(\breve{\mathbf{B}}\right) \approx \sum_{m=1}^{M} \hat{w}_m \left[\log\left(\det\left(\mathrm{DIAG}\left(\breve{\mathbf{B}} \left(\hat{\mathbf{W}} \hat{\mathbf{R}}_m \hat{\mathbf{W}}^H \right) \breve{\mathbf{B}}^H \right) \right) \right) - \log\left(\det\left(\mathrm{DIAG}\left(\breve{\mathbf{B}} \breve{\mathbf{B}}^H \right) \right) \right) \right] = Q''\left(\breve{\mathbf{B}}\right) \quad (3.14)$$

Let $\breve{\mathbf{B}} = \left[\breve{\mathbf{b}}_1, ..., \breve{\mathbf{b}}_K \right]^H$ (3.14) can be written in the following manner

$$Q''\left(\breve{\mathbf{B}}\right) = \sum_{m=1}^{M} \hat{w}_m \sum_{k=1}^{K} \log\left(\frac{\breve{\mathbf{b}}_k^H}{\left\| \breve{\mathbf{b}}_k \right\|_2} \hat{\mathbf{W}} \hat{\mathbf{R}}_m \hat{\mathbf{W}}^H \frac{\breve{\mathbf{b}}_k}{\left\| \breve{\mathbf{b}}_k \right\|_2} \right) = \sum_{m=1}^{M} \hat{w}_m \sum_{k=1}^{K} \log\left(\tilde{\mathbf{b}}_k^H \hat{\mathbf{W}} \hat{\mathbf{R}}_m \hat{\mathbf{W}}^H \tilde{\mathbf{b}}_k \right), \quad (3.15)$$

where $\left\| \tilde{\mathbf{b}}_k \right\|_2 = 1 \ \forall k = 1, ..., K$. The function Q'' is minimized under the constraint of $\left\| \tilde{\mathbf{b}}_k \right\|_2 = 1 \ \forall k = 1, ..., K$. Thus, let $\hat{\mathbf{R}}_m = \hat{\mathbf{W}} \hat{\mathbf{R}}_m \hat{\mathbf{W}}^H$ the objective function becomes

$$Q'''\left(\tilde{\mathbf{B}}\right) = \sum_{m=1}^{M} \hat{w}_m \sum_{k=1}^{K} \left[\log\left(\tilde{\mathbf{b}}_k^H \hat{\mathbf{R}}_m \tilde{\mathbf{b}}_k \right) - \lambda_k \left(\tilde{\mathbf{b}}_k^H \tilde{\mathbf{b}}_k - 1 \right) \right], \quad (3.16)$$

where $\left\{ \lambda_k \right\}_{k=1}^{K}$ are the Lagrange multipliers. Minimization of Q''' w.r.t. $\tilde{\mathbf{B}}$, causes a scaled estimation of $\mathbf{B}$. In BSS applications, this inherent limitation is well known and usually tolerable. Equating the partial derivatives of Q''' w.r.t. $\left\{ \tilde{\mathbf{b}}_k \right\}_{k=1}^{K}$ to zero, yields

$$\sum_{m=1}^{M} \hat{w}_m \left(\tilde{\mathbf{b}}_k^H \hat{\mathbf{R}}_m \tilde{\mathbf{b}}_k \right)^{-1} \hat{\mathbf{R}}_m \tilde{\mathbf{b}}_k = \lambda_k \tilde{\mathbf{b}}_k . \quad (3.17)$$

It is noted that the complex derivatives are defined in [35]. Left multiplication of (3.17) by $\tilde{\mathbf{b}}_k^H$, and applying the constraint $\tilde{\mathbf{b}}_k^H \tilde{\mathbf{b}}_k = 1$, implies that $\lambda_k = 1 \ \ \forall \ k = 1, ..., K$. Therefore, (3.17) can be rewritten in the form

$$\mathbf{G}\left(\tilde{\mathbf{b}}_k \right) \tilde{\mathbf{b}}_k = \tilde{\mathbf{b}}_k , \quad (3.18)$$

where

$$\mathbf{G}\left(\tilde{\mathbf{b}}_k \right) \square \sum_{m=1}^{M} \hat{w}_m \left(\tilde{\mathbf{b}}_k^H \hat{\mathbf{R}}_m \tilde{\mathbf{b}}_k \right)^{-1} \hat{\mathbf{R}}_m \quad (3.19)$$

is a Hermitian matrix.

Direct solution of (3.18) is analytically cumbersome. Therefore, an iterative algorithm, which minimizes the L_2 norm of the difference between the l.h.s and r.h.s of (3.18) w.r.t. $\tilde{\mathbf{b}}_k$ $\left(k=1,...,K\right)$, is derived. The L_2 norm of $\mathbf{G}\left(\tilde{\mathbf{b}}_k\right)\tilde{\mathbf{b}}_k-\tilde{\mathbf{b}}_k$ is

$$\xi\left(\tilde{\mathbf{b}}_k\right)=\tilde{\mathbf{b}}_k^H\left(\mathbf{G}\left(\tilde{\mathbf{b}}_k\right)-\mathbf{I}\right)^2\tilde{\mathbf{b}}_k. \tag{3.20}$$

In order to enforce unit L_2 norm on $\tilde{\mathbf{b}}_k$, the following expression is minimized w.r.t. $\tilde{\mathbf{b}}_k$, and α_k, $k=1,...,K$

$$\xi'\left(\tilde{\mathbf{b}}_k,\alpha_k\right)=\tilde{\mathbf{b}}_k^H\left(\mathbf{G}\left(\tilde{\mathbf{b}}_k\right)-\mathbf{I}\right)^2\tilde{\mathbf{b}}_k-\alpha_k\left(\tilde{\mathbf{b}}_k^H\tilde{\mathbf{b}}_k-1\right), \tag{3.21}$$

where α_k is the Lagrange multiplier. It is noted that since $\xi\left(\tilde{\mathbf{b}}_k\right)$ is real, then α_k is real. Let $\tilde{\mathbf{b}}_k^*$ be defined as

$$\tilde{\mathbf{b}}_k^*=\arg\min_{\tilde{\mathbf{b}}_k}\min_{\alpha_k}\xi'\left(\tilde{\mathbf{b}}_k,\alpha_k\right). \tag{3.22}$$

The direct minimization in (3.22) is analytically cumbersome. Therefore, minimization of (3.21) w.r.t. $\tilde{\mathbf{b}}_k$ is carried out by iterative minimization of the following auxiliary function

$$\psi\left(\tilde{\mathbf{b}}_k,\tilde{\mathbf{b}}_k^{\text{old}},\alpha_k\right)=\tilde{\mathbf{b}}_k^H\left(\mathbf{G}\left(\tilde{\mathbf{b}}_k^{\text{old}}\right)-\mathbf{I}\right)^2\tilde{\mathbf{b}}_k-\alpha_k\left(\tilde{\mathbf{b}}_k^H\tilde{\mathbf{b}}_k-1\right), \tag{3.23}$$

where $\tilde{\mathbf{b}}_k^{old}$ is an initially guessed vector in the vicinity of $\tilde{\mathbf{b}}_k^*$ and

$$\tilde{\mathbf{b}}_k^{\text{new}}=\arg\min_{\tilde{\mathbf{b}}_k}\min_{\alpha_k}\psi\left(\tilde{\mathbf{b}}_k,\tilde{\mathbf{b}}_k^{\text{old}},\alpha_k\right). \tag{3.24}$$

This process is iterated until convergence, where $\tilde{\mathbf{b}}_k^{\text{old}}$ of the next iteration is $\tilde{\mathbf{b}}_k^{\text{new}}$ of the current one. In each iteration, minimization of $\psi\left(\tilde{\mathbf{b}}_k,\tilde{\mathbf{b}}_k^{\text{old}},\alpha_k\right)$ is carried out by equating its derivative w.r.t. $\tilde{\mathbf{b}}_k$ to zero. Hence, the following equation is solved

$$\left(\mathbf{G}\left(\tilde{\mathbf{b}}_k^{\text{old}}\right)-\mathbf{I}\right)^2\tilde{\mathbf{b}}_k^{\text{new}}=\alpha_k\tilde{\mathbf{b}}_k^{\text{new}}. \tag{3.25}$$

According to (3.24) and (3.25), $\tilde{\mathbf{b}}_k^{\text{new}}$ is the eigenvector of $\left(\mathbf{G}\left(\tilde{\mathbf{b}}_k^{\text{old}}\right)-\mathbf{I}\right)^2$, corresponding to the minimal eigenvalue, α_k. Since $\left(\mathbf{G}\left(\tilde{\mathbf{b}}_k^{\text{old}}\right)-\mathbf{I}\right)^2$ is Hermitian, then α_k is real.

Finally, the iterative algorithm for the estimation of $\tilde{\mathbf{b}}_k$, named as SVDJD comprises the following steps:

1. Set $k=1$**.**

2. Let $l=1$**. Initialize** $\tilde{\mathbf{b}}_k^{(0)}$**.**

3. **Solve** $\left(\mathbf{G}\left(\tilde{\mathbf{b}}_k^{(l-1)}\right)-\mathbf{I}\right)^2 \tilde{\mathbf{b}}_k^{(l)} = \alpha_k^{(l)} \tilde{\mathbf{b}}_k^{(l)}$ **and pick** $\tilde{\mathbf{b}}_k^{(l)}$**, corresponding to the minimal eigenvalue,** $\alpha_k^{(l)}$**.**
4. **If** $\alpha_k^{(l)} > \varepsilon$ **(**ε **is a predefined small positive threshold), then** $l = l+1$ **and go to step 3.**
5. **If** $\alpha_k^{(l)} \le \varepsilon$**, then if** $k < K$**, then** $k = k+1$ **and go to step 2, else stop.**

In order to examine the typical averaged convergence patterns of $\xi\left(\tilde{\mathbf{b}}_k^{(l)}\right)$, a data set consisting of 1000 random realizations of the complex three-dimensional matrix set $\left\{\mathbf{A}\mathbf{\Lambda}_m\mathbf{A}^H + \sigma^2\mathbf{E}_m\mathbf{E}_m^H\right\}_{m=1}^M$ was generated. The elements of the matrix $\mathbf{A}$ are drawn from a complex standard normal distribution and the matrices $\left\{\mathbf{\Lambda}_m\right\}_{m=1}^M$ are real diagonal with elements uniformly distributed in $(0,1]$. The matrices $\left\{\mathbf{E}_m\right\}_{m=1}^M$ are perturbation matrices, randomized from a complex normal standard distribution and the scalar σ^2 was set to 0.01. The number of matrices, *M*, was set to 25 and the estimated weights, $\left\{\hat{w}_m\right\}_{m=1}^M$, were equally set to $\frac{1}{M}$. Fig. 12, depicts the average values of $\log\xi\left(\tilde{\mathbf{b}}_k^{(l)}\right)$ for $k = 1,2,3$, as a function of the iteration index *l*. One can notice that the typical averaged convergence pattern is exponential and that the algorithm converges after approximately 20-25 iterations in each dimension.

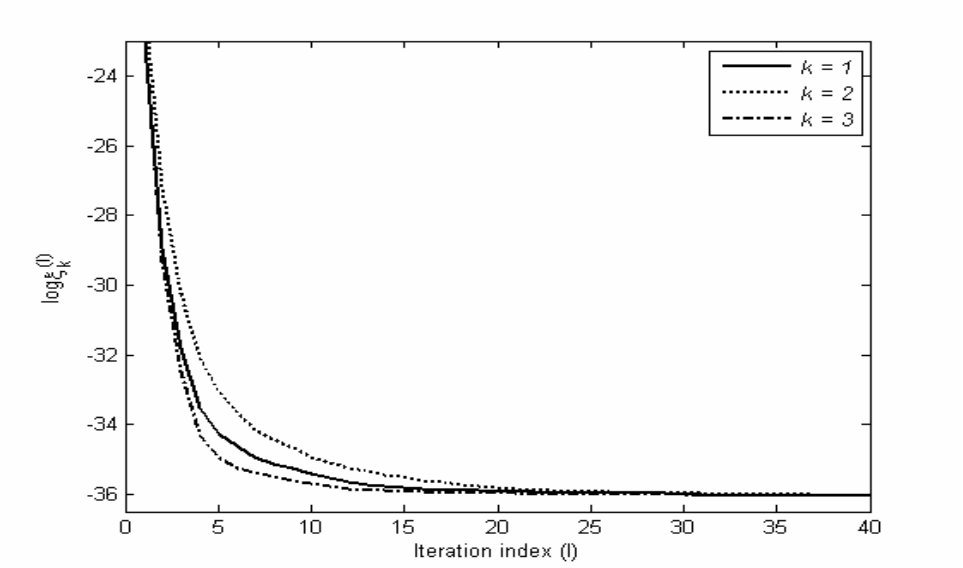

Fig. 12. Typical averaged convergence patterns of the minimization algorithm.

3.3 CONVERGENCE

In this section, a sufficient convergence condition of the iterative minimization algorithm, described in Section 3.2, is derived. The existence of the condition is studied by simulations. Let l denote an iteration index, and let $\mathbf{e}_k^{(l)} \triangleq \tilde{\mathbf{b}}_k^{(l)} - \tilde{\mathbf{b}}_k^*$. A sufficient condition for $\left\|\mathbf{e}_k^{(l)}\right\|_2^2 \le \left\|\mathbf{e}_k^{(l-1)}\right\|_2^2 \quad \forall\, k \in 1,...,K$ and $\forall\, l = 1,...,L_k$, is derived in the following manner.

Let $\mathbf{x}, \mathbf{y} \in \mathbb{C}^K$. Then according to (3.24)

$$\psi(\mathbf{x},\mathbf{y},\alpha) = \mathbf{x}^H \left(\mathbf{G}(\mathbf{y}) - \mathbf{I}\right)^2 \mathbf{x} - \alpha\left(\mathbf{x}^H\mathbf{x} - 1\right). \tag{3.26}$$

The second order Taylor approximation of $\psi(\mathbf{x},\mathbf{y},\alpha)$ around $\mathbf{x} = \tilde{\mathbf{b}}_k^*$, $\mathbf{y} = \tilde{\mathbf{b}}_k^*$ and $\alpha_k^* = 0$ is

$$\begin{aligned}\psi(\mathbf{x},\mathbf{y},\alpha) \approx{}& \psi\left(\tilde{\mathbf{b}}_k^*, \tilde{\mathbf{b}}_k^*, \alpha_k^*\right) + \left[\begin{array}{ccc}\frac{\partial\psi}{\partial\mathbf{x}} & \frac{\partial\psi}{\partial\mathbf{y}} & \frac{\partial\psi}{\partial\alpha}\end{array}\right]\Bigg|_{\substack{\mathbf{x}=\tilde{\mathbf{b}}_k^* \\ \mathbf{y}=\tilde{\mathbf{b}}_k^* \\ \alpha=\alpha_k^*}} \begin{bmatrix}\mathbf{x}-\tilde{\mathbf{b}}_k^* \\ \mathbf{y}-\tilde{\mathbf{b}}_k^* \\ \alpha-\alpha_k^*\end{bmatrix} \\ &+ \begin{bmatrix}\mathbf{x}-\tilde{\mathbf{b}}_k^* \\ \mathbf{y}-\tilde{\mathbf{b}}_k^* \\ \alpha-\alpha_k^*\end{bmatrix}^H \underbrace{\begin{bmatrix}\frac{\partial^2\psi}{\partial\mathbf{x}\partial\mathbf{x}^H} & \frac{\partial^2\psi}{\partial\mathbf{y}\partial\mathbf{x}^H} & \frac{\partial^2\psi}{\partial\alpha\partial\mathbf{x}^H} \\ \frac{\partial^2\psi}{\partial\mathbf{x}\partial\mathbf{y}^H} & \frac{\partial^2\psi}{\partial\mathbf{y}\partial\mathbf{y}^H} & \frac{\partial^2\psi}{\partial\alpha\partial\mathbf{y}^H} \\ \frac{\partial^2\psi}{\partial\mathbf{x}\partial\alpha} & \frac{\partial^2\psi}{\partial\mathbf{y}\partial\alpha} & \frac{\partial^2\psi}{\partial^2\alpha}\end{bmatrix}}_{\mathbf{H}}\Bigg|_{\substack{\mathbf{x}=\tilde{\mathbf{b}}_k^* \\ \mathbf{y}=\tilde{\mathbf{b}}_k^* \\ \alpha=\alpha_k^*}} \begin{bmatrix}\mathbf{x}-\tilde{\mathbf{b}}_k^* \\ \mathbf{y}-\tilde{\mathbf{b}}_k^* \\ \alpha-\alpha_k^*\end{bmatrix},\end{aligned} \tag{3.27}$$

where $\mathbf{H}$ denotes the Hessian matrix. According to (3.23) the minimum of $\psi(\mathbf{x},\mathbf{y},\alpha)$ is obtained for $\mathbf{x} = \tilde{\mathbf{b}}_k^*$, $\mathbf{y} = \tilde{\mathbf{b}}_k^*$ and $\alpha = \alpha_k^*$, where $\psi\left(\tilde{\mathbf{b}}_k^*, \tilde{\mathbf{b}}_k^*, \alpha_k^*\right) = 0$. Therefore, the gradient of $\psi(\mathbf{x},\mathbf{y},\alpha)$ at $\mathbf{x} = \tilde{\mathbf{b}}_k^*$, $\mathbf{y} = \tilde{\mathbf{b}}_k^*$ and $\alpha = \alpha_k^*$ is zero, and (3.27) reduces to the following form.

$$\begin{aligned}\psi(\mathbf{x},\mathbf{y},\alpha) \approx{}& \left(\mathbf{x}-\tilde{\mathbf{b}}_k^*\right)^H \mathbf{H}_{1,1}\left(\mathbf{x}-\tilde{\mathbf{b}}_k^*\right) + 2\left(\mathbf{x}-\tilde{\mathbf{b}}_k^*\right)^H \mathbf{H}_{1,2}\left(\mathbf{y}-\tilde{\mathbf{b}}_k^*\right) + 2\left(\mathbf{x}-\tilde{\mathbf{b}}_k^*\right)^H \mathbf{H}_{1,3}\alpha \\ &+ 2\left(\mathbf{y}-\tilde{\mathbf{b}}_k^*\right)^H \mathbf{H}_{2,3}\alpha + \left(\mathbf{y}-\tilde{\mathbf{b}}_k^*\right)^H \mathbf{H}_{2,2}\left(\mathbf{y}-\tilde{\mathbf{b}}_k^*\right) + \mathbf{H}_{3,3}\alpha^2,\end{aligned} \tag{3.28}$$

Calculation of the Hessian submatrices is described in Appendix F. According to (F.2), (F.9) and (F.10) it is implied that

$$\mathbf{H}_{1,1}=\left(\mathbf{G}\left(\tilde{\mathbf{b}}_k^*\right)-\mathbf{I}\right)^2\ ;\ \mathbf{H}_{1,2}=-\left(\mathbf{G}\left(\tilde{\mathbf{b}}_k^*\right)-\mathbf{I}\right)\mathbf{P}\left(\tilde{\mathbf{b}}_k^*\right)\ ; \tag{3.29}$$
$$\mathbf{H}_{1,3}=-\tilde{\mathbf{b}}_k^*\ ;\ \mathbf{H}_{2,3}=\mathbf{0}\ ;\ \mathbf{H}_{3,3}=\mathbf{0},$$

where according to (F.8) $\mathbf{P}\left(\mathbf{b}_k^*\right)=\sum_{m=1}^{M}\hat{w}_m\frac{\widehat{\mathbf{R}}_m\mathbf{b}_k^*\mathbf{b}_k^{*H}\widehat{\mathbf{R}}_m}{\left(\mathbf{b}_k^{*H}\widehat{\mathbf{R}}_m\mathbf{b}_k^*\right)^2}$. Therefore, (3.28) can be written in the following manner

$$\begin{aligned}\psi\left(\mathbf{x},\mathbf{y},\alpha\right)\approx&\left(\mathbf{x}-\tilde{\mathbf{b}}_k^*\right)^H\mathbf{H}_{1,1}\left(\mathbf{x}-\tilde{\mathbf{b}}_k^*\right)+2\left(\mathbf{x}-\tilde{\mathbf{b}}_k^*\right)^H\mathbf{H}_{1,2}\left(\mathbf{y}-\tilde{\mathbf{b}}_k^*\right)\\&+2\alpha\left(\mathbf{x}-\tilde{\mathbf{b}}_k^*\right)^H\mathbf{H}_{1,3}+\left(\mathbf{y}-\tilde{\mathbf{b}}_k^*\right)^H\mathbf{H}_{2,2}\left(\mathbf{y}-\tilde{\mathbf{b}}_k^*\right).\end{aligned} \tag{3.30}$$

Equating the derivative of ψ w.r.t. α to zero implies that

$$\frac{\partial\psi\left(\mathbf{x},\mathbf{y},\alpha\right)}{\partial\alpha}=0\Rightarrow-\left(\mathbf{x}-\tilde{\mathbf{b}}_k^*\right)^H\tilde{\mathbf{b}}_k^*=0\,. \tag{3.31}$$

This implies that the projection of the error in the direction of $\tilde{\mathbf{b}}_k^*$ is zero. In fact, we can learn that the algorithm nullifies the error in the direction of $\tilde{\mathbf{b}}_k^*$ and minimizes the error in directions orthogonal to $\tilde{\mathbf{b}}_k^*$.

Minimization of $\psi\left(\mathbf{x},\mathbf{y},\alpha\right)$ w.r.t $\mathbf{x}$ and α, as described in (3.24), is carried out by equating the derivative of ψ w.r.t. $\mathbf{x}$ to zero. Hence,

$$\frac{\partial\psi\left(\mathbf{x},\mathbf{y},\alpha\right)}{\partial\mathbf{x}}=\mathbf{0}\Rightarrow\mathbf{H}_{1,1}\left(\mathbf{x}-\tilde{\mathbf{b}}_k^*\right)+2\mathbf{H}_{1,2}\left(\mathbf{y}-\tilde{\mathbf{b}}_k^*\right)+2\alpha\tilde{\mathbf{b}}_k^*=\mathbf{0}\,. \tag{3.32}$$

Left multiplication of (3.32) by $\tilde{\mathbf{b}}_k^{*H}$ and applying that $\mathbf{G}\left(\tilde{\mathbf{b}}_k^*\right)\tilde{\mathbf{b}}_k^*=\tilde{\mathbf{b}}_k^*$ yields

$$\begin{aligned}\alpha=&-\frac{1}{2}\tilde{\mathbf{b}}_k^{*H}\mathbf{H}_{1,1}\left(\mathbf{x}-\tilde{\mathbf{b}}_k^*\right)-\tilde{\mathbf{b}}_k^{*H}\mathbf{H}_{1,2}\left(\mathbf{y}-\tilde{\mathbf{b}}_k^*\right)\\=&-\frac{1}{2}\underbrace{\tilde{\mathbf{b}}_k^{*H}\left(\mathbf{G}\left(\tilde{\mathbf{b}}_k^*\right)-\mathbf{I}\right)^2}_{\mathbf{0}^H}\left(\mathbf{x}-\tilde{\mathbf{b}}_k^*\right)+\underbrace{\tilde{\mathbf{b}}_k^{*H}\left(\mathbf{G}\left(\tilde{\mathbf{b}}_k^*\right)-\mathbf{I}\right)}_{\mathbf{0}^H}\mathbf{P}\left(\tilde{\mathbf{b}}_k^*\right)\left(\mathbf{y}-\tilde{\mathbf{b}}_k^*\right)=0.\end{aligned} \tag{3.33}$$

Substitution of (3.33) into (3.32) implies that

$$\begin{aligned}&\mathbf{H}_{1,1}\left(\mathbf{x}-\tilde{\mathbf{b}}_k^*\right)=-2\mathbf{H}_{1,2}\left(\mathbf{y}-\tilde{\mathbf{b}}_k^*\right)\\&\Rightarrow\left(\mathbf{G}\left(\tilde{\mathbf{b}}_k^*\right)-\mathbf{I}\right)^2\left(\mathbf{x}-\tilde{\mathbf{b}}_k^*\right)=2\left(\mathbf{G}\left(\tilde{\mathbf{b}}_k^*\right)-\mathbf{I}\right)\mathbf{P}\left(\tilde{\mathbf{b}}_k^*\right)\left(\mathbf{y}-\tilde{\mathbf{b}}_k^*\right).\end{aligned} \tag{3.34}$$

Let $\mathbf{x}=\tilde{\mathbf{b}}_k^{(l)}$ and $\mathbf{y}=\tilde{\mathbf{b}}_k^{(l-1)}$, where $\tilde{\mathbf{b}}_k^{(l)}$ and $\tilde{\mathbf{b}}_k^{(l-1)}$ are analogous to $\tilde{\mathbf{b}}_k^{\text{new}}$ and $\tilde{\mathbf{b}}_k^{\text{old}}$, respectively, as described in (3.24). Equation (3.34) can be written as

$$\left(\mathbf{G}\left(\tilde{\mathbf{b}}_k^*\right)-\mathbf{I}\right)^2 \mathbf{e}_k^{(l)} = 2\left(\mathbf{G}\left(\tilde{\mathbf{b}}_k^*\right)-\mathbf{I}\right)\mathbf{P}\left(\tilde{\mathbf{b}}_k^*\right)\mathbf{e}_k^{(l-1)}. \tag{3.35}$$

Equation (3.35) relates the error vector in the l^{th} iteration to the error vector in the $(l-1)^{\text{th}}$ iteration. Let $\mathbf{u}_k^t$ denote an eigenvector of $\mathbf{G}\left(\tilde{\mathbf{b}}_k^*\right)$, which is perpendicular to $\tilde{\mathbf{b}}_k^*$. Left multiplication of (3.35) by $\mathbf{u}_k^{t\,H}$ implies that

$$\left(\lambda_k^t-1\right)^2 \mathbf{u}_k^{t\,H}\mathbf{e}_k^{(l)} = 2\left(\lambda_k^t-1\right)\mathbf{u}_k^{t\,H}\mathbf{P}\left(\tilde{\mathbf{b}}_k^*\right)\mathbf{e}_k^{(l-1)}, \tag{3.36}$$

where λ_k^t is an eigenvalue of $\mathbf{G}\left(\tilde{\mathbf{b}}_k^*\right)$, corresponding to $\mathbf{u}_k^t$. Let $\mathbf{P}\left(\tilde{\mathbf{b}}_k^*\right)=\sum_{t=1}^{K}\varphi_k^t\mathbf{v}_k^t\mathbf{v}_k^{t\,H}$, where φ_k^t and $\mathbf{v}_k^t$ denote an eigenvalue and a corresponding eigenvector of $\mathbf{P}\left(\tilde{\mathbf{b}}_k^*\right)$, respectively. Since according to (F.8), one of the eigenvectors of $\mathbf{P}\left(\tilde{\mathbf{b}}_k^*\right)$ is $\tilde{\mathbf{b}}_k^*$ with corresponding eigenvalue is equal to 1, then $\mathbf{P}\left(\tilde{\mathbf{b}}_k^*\right)=\underbrace{\sum_{t=1}^{K-1}\varphi_k^t\mathbf{v}_k^t\mathbf{v}_k^{t\,H}}_{\tilde{\mathbf{P}}\left(\tilde{\mathbf{b}}_k^*\right)}+\tilde{\mathbf{b}}_k^*\tilde{\mathbf{b}}_k^{*H}$. Substitution of $\mathbf{P}\left(\tilde{\mathbf{b}}_k^*\right)$ into (3.36) and using the fact that according to (3.31) $\mathbf{e}_k^{(l)}\perp\tilde{\mathbf{b}}_k^*\ \forall l$ implies that

$$\mathbf{u}_k^{t\,H}\mathbf{e}_k^{(l)} = \frac{2}{\left(\lambda_k^t-1\right)}\mathbf{u}_k^{t\,H}\tilde{\mathbf{P}}\left(\tilde{\mathbf{b}}_k^*\right)\mathbf{e}_k^{(l-1)}. \tag{3.37}$$

Hence,

$$\mathbf{e}_k^{(l)H}\mathbf{u}_k^t\mathbf{u}_k^{t\,H}\mathbf{e}_k^{(l)} = \frac{4}{\left(\lambda_k^t-1\right)^2}\mathbf{e}_k^{(l-1)H}\tilde{\mathbf{P}}^H\left(\tilde{\mathbf{b}}_k^*\right)\mathbf{u}_k^t\mathbf{u}_k^{t\,H}\tilde{\mathbf{P}}\left(\tilde{\mathbf{b}}_k^*\right)\mathbf{e}_k^{(l-1)} \tag{3.38}$$

Summation of (3.38) over *t* implies that

$$\mathbf{e}_k^{(l)H}\underbrace{\sum_{t=1}^{K-1}\mathbf{u}_k^t\mathbf{u}_k^{t\,H}}_{\mathbf{I}-\tilde{\mathbf{b}}_k^*\tilde{\mathbf{b}}_k^{*H}}\mathbf{e}_k^{(l)} = \mathbf{e}_k^{(l-1)H}\tilde{\mathbf{P}}\left(\tilde{\mathbf{b}}_k^*\right)\sum_{t=1}^{K-1}\frac{4}{\left(\lambda_k^t-1\right)^2}\mathbf{u}_k^t\mathbf{u}_k^{t\,H}\tilde{\mathbf{P}}\left(\tilde{\mathbf{b}}_k^*\right)\mathbf{e}_k^{(l-1)}. \tag{3.39}$$

Let $\mathbf{F}\left(\tilde{\mathbf{b}}_k^*\right)=\sum_{t=1}^{K-1}\frac{2}{\left(\lambda_k^t-1\right)}\mathbf{u}_k^t\mathbf{u}_k^{t\,H}$. According to (3.31), since the error is orthogonal to $\mathbf{b}_k^*$ then (3.39) can be written in the following manner

$$\mathbf{e}_k^{(l)H}\mathbf{e}_k^{(l)} = \mathbf{e}_k^{(l-1)H}\tilde{\mathbf{P}}^H\left(\tilde{\mathbf{b}}_k^*\right)\mathbf{F}^H\left(\tilde{\mathbf{b}}_k^*\right)\mathbf{F}\left(\tilde{\mathbf{b}}_k^*\right)\tilde{\mathbf{P}}\left(\mathbf{b}_k^*\right)\mathbf{e}_k^{(l-1)}. \tag{3.40}$$

Therefore,

$$\left\|\mathbf{e}_k^{(l)}\right\|_2^2 = \left\|\mathbf{F}\left(\tilde{\mathbf{b}}_k^*\right)\tilde{\mathbf{P}}\left(\tilde{\mathbf{b}}_k^*\right)\mathbf{e}_k^{(l-1)}\right\|_2^2. \tag{3.41}$$

According to the Cauchy-Schwartz inequality

$$\left\|\mathbf{e}_k^{(l)}\right\|_2^2 \le \left\|\mathbf{F}\left(\tilde{\mathbf{b}}_k^*\right)\tilde{\mathbf{P}}\left(\tilde{\mathbf{b}}_k^*\right)\right\|_2^2 \left\|\mathbf{e}_k^{(l-1)}\right\|_2^2. \tag{3.42}$$

Let $\|\mathbf{A}\|_2 \equiv \max\left\{\sqrt{\sigma} : \sigma \text{ is an eigenvalue of } \mathbf{A}^H\mathbf{A}\right\}$, where $\mathbf{A} \in \mathbb{C}^{K\times K}$. If the maximal eigenvalue of $\mathbf{F}\left(\tilde{\mathbf{b}}_k^*\right)\tilde{\mathbf{P}}\left(\tilde{\mathbf{b}}_k^*\right)$ is equal to or smaller than 1, then

$$\left\|\mathbf{e}_k^{(l)}\right\|_2^2 \le \left\|\mathbf{e}_k^{(l-1)}\right\|_2^2 \tag{3.43}$$

and convergence is guarantied.

The existence of the sufficient condition for convergence is studied in the following simulation. Let K and σ^2 denote a matrix dimension and perturbation level, respectively. For each $K = 2,5,10$ and $\sigma^2 = 10,1,0.1,0.01,0.001,0.0001,0.00001$, a data set consisting of 1000 random realizations of the matrix set $\left\{\mathbf{A}\mathbf{\Lambda}_m\mathbf{A}^H + \sigma^2\mathbf{E}_m\mathbf{E}_m^H\right\}_{m=1}^M$ was generated. The elements of the matrix $\mathbf{A}$ are drawn from a real standard normal distribution and the matrices $\{\mathbf{\Lambda}_m\}_{m=1}^M$ are real diagonal with elements uniformly distributed in $(0,1]$. The matrices $\{\mathbf{E}_m\}_{m=1}^M$ are perturbation matrices, randomized from a real normal standard distribution. The number of matrices, *M* was set to 100 and the weights, $\{\hat{w}_m\}_{m=1}^M$, were equally set to $\frac{1}{M}$. For each K and σ^2 the number of events satisfying $\max_k\left\{\max \text{eigval}\left(\mathbf{F}\left(\tilde{\mathbf{b}}_k^*\right)\tilde{\mathbf{P}}\left(\tilde{\mathbf{b}}_k^*\right)\right)\right\}_{k=1}^K \le 1$ was divided by 1000 to obtain the estimated probability of existence of the sufficient convergence condition. Fig. 13, depicts the estimated probability of existence of the sufficient convergence condition versus σ^2.

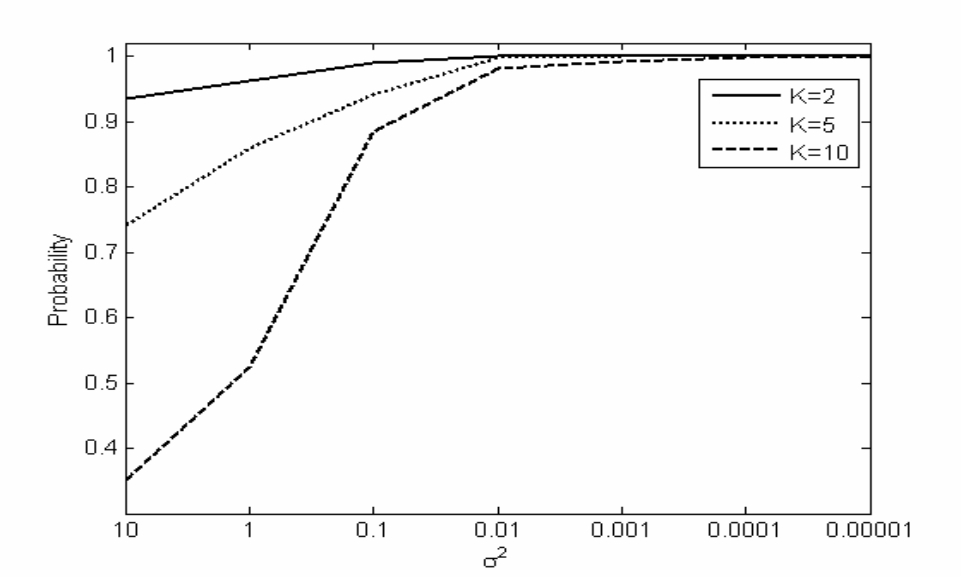

Fig. 13. The probability of existence of the sufficient convergence condition versus σ^2 for different K.

It is important to note that the simulations showed convergence in all randomized cases.

3.4 INITIALIZATION

According to the iterative algorithm for the estimation of $\left\{\tilde{\mathbf{b}}_k\right\}_{k=1}^{K}$, described in Section 3.2, each $\tilde{\mathbf{b}}_k$ is estimated separately. Hence, improper initialization of $\left\{\tilde{\mathbf{b}}_k\right\}_{k=1}^{K}$, might consequence convergence into undesired identical solutions ($\tilde{\mathbf{b}}_k$'s), such that $\text{rank}\left(\tilde{\mathbf{B}}\right) < K$. In this section, we propose a method for easy and robust initialization of $\left\{\tilde{\mathbf{b}}_k\right\}_{k=1}^{K}$. We begin by showing that $\tilde{\mathbf{b}}_k$ is a linear combination of the eigenvectors of $\left\{\widehat{\mathbf{R}}_m\right\}_{m=1}^{M}$. According to (3.17) and (3.18),

$$\tilde{\mathbf{b}}_k = \sum_{m=1}^{M} \hat{w}_m \left(\tilde{\mathbf{b}}_k^H \widehat{\mathbf{R}}_m \tilde{\mathbf{b}}_k\right)^{-1} \widehat{\mathbf{R}}_m \tilde{\mathbf{b}}_k . \tag{3.44}$$

Let

$$\widehat{\mathbf{R}}_m = \sum_{k=1}^{K} \hat{\lambda}_{k,m} \hat{\mathbf{v}}_{k,m} \hat{\mathbf{v}}_{k,m}^H , \tag{3.45}$$

where $\hat{\lambda}_{k,m}$ and $\hat{\mathbf{v}}_{k,m}$ are the k^{th} eigenvalue and eigenvector of $\widehat{\mathbf{R}}_m$, respectively. Substituting (3.45) into (3.44) yields the following expression

$$\tilde{\mathbf{b}}_k = \sum_{m=1}^{M} \hat{w}_m \frac{\sum_{j=1}^{K} \hat{\lambda}_{j,m} \left\langle \hat{\mathbf{v}}_{j,m}, \tilde{\mathbf{b}}_k \right\rangle \hat{\mathbf{v}}_{j,m}}{\sum_{j=1}^{K} \hat{\lambda}_{j,m} \left| \left\langle \hat{\mathbf{v}}_{j,m}, \tilde{\mathbf{b}}_k \right\rangle \right|^2}, \tag{3.46}$$

where $\langle \cdot, \cdot \rangle$ and $|\cdot|$ denote the scalar product and absolute operators, respectively.

According to Appendix H, the set $\left\{ \hat{\mathbf{v}}_{k,m} \right\}_{k=1,m=1}^{K,M}$ can be partitioned into *K* distinct clusters of eigenvectors, as depicted in Fig. 14. Therefore, each $\tilde{\mathbf{b}}_k$ has inner products close to 1 with the eigenvectors attributed to one and only one cluster and inner products close to 0 with the eigenvectors attributed to the rest of the clusters. Therefore, (3.46) can be approximated in the following manner

$$\tilde{\mathbf{b}}_k \approx \sum_{m=1}^{M} \hat{w}_m \hat{\mathbf{v}}_{k,m} \tag{3.47}$$

and $\left\{ \tilde{\mathbf{b}}_k \right\}_{k=1}^{K}$ may be initialized accordingly.

Since eigenvalue decomposition is unique up to phase (sign in the real case) and permutation, the order and phases of the vectors in each eigenvectors matrix of $\left\{ \widehat{\mathbf{R}}_m \right\}_{m=1}^{M}$ is arbitrary. As a consequence, initialization of $\left\{ \tilde{\mathbf{b}}_k \right\}_{k=1}^{K}$ according to (3.47) should be preceded by order and phases coordination of the vectors in each eigenvectors matrix of $\left\{ \widehat{\mathbf{R}}_m \right\}_{m=1}^{M}$. An algorithm for coordination of the order and phases of the vectors in each eigenvectors matrix of $\left\{ \widehat{\mathbf{R}}_m \right\}_{m=1}^{M}$ is described in Appendix G.

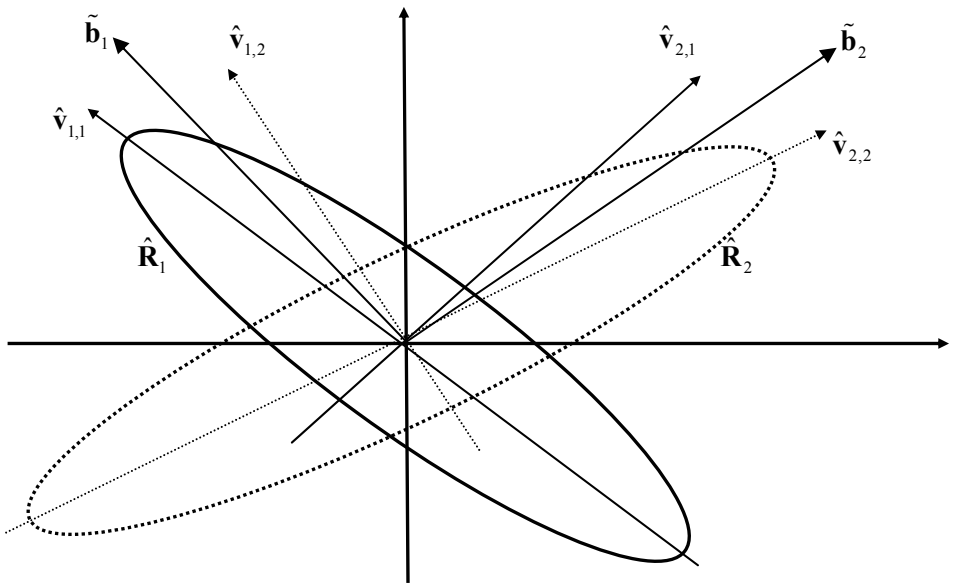

Fig. 14. Illustration of a matrix set, having distinct clusters of eigenvalues.

Finally, the use of the proposed initialization method in estimating $\tilde{\mathbf{B}}$ is summarized in the following algorithm:

1. **Estimate $\hat{\mathbf{W}}$ according to (H.11)**
2. **Estimate the eigenvectors of $\left\{\widehat{\mathbf{R}}_m = \hat{\mathbf{W}}\hat{\mathbf{R}}_m\hat{\mathbf{W}}^H\right\}_{m=1}^{M}$, denoted by $\left\{\hat{\mathbf{v}}_{k,m}\right\}_{k=1,m=1}^{K,M}$.**
3. **Synchronize the order and signs of $\left\{\hat{\mathbf{v}}_{k,m}\right\}_{k=1,m=1}^{K,M}$ according to the algorithm described in Appendix G.**
4. **Let $\tilde{\mathbf{B}} = \left[\tilde{\mathbf{b}}_1,...,\tilde{\mathbf{b}}_K\right]^H$. Initialize $\tilde{\mathbf{b}}_k \approx \sum_{m=1}^{M}\hat{w}_m\hat{\mathbf{v}}_{k,m}$ according to (3.47).**
5. **Estimate $\tilde{\mathbf{B}}$ using the minimization algorithm described in Section 3.2 i.e. $\hat{\tilde{\mathbf{B}}} = \arg\min_{\tilde{\mathbf{B}}} Q'''\left(\tilde{\mathbf{B}}\right)$.**
6. **Estimate $\hat{\mathbf{B}} = \hat{\tilde{\mathbf{B}}}\hat{\mathbf{W}}$.**

3.5 COMPUTATIONAL COMPLEXITY ASPECTS

In this section, an asymptotic computational load analysis of the proposed algorithm is presented and compared to other existing techniques for approximate joint diagonalization. According to the minimization algorithm described in Section 3.2, for each dimension, denoted by k $(k = 1,...,K)$, and in each iteration, the matrix $\left(\mathbf{G}\left(\tilde{\mathbf{b}}_k\right) - \mathbf{I}\right)^2$ is derived according to (3.20) and its eigenvectors and eigenvalues are computed using SVD. The computational load of calculating $\left(\mathbf{G}\left(\tilde{\mathbf{b}}_k\right) - \mathbf{I}\right)^2$ is $O(M)$ and according to [22] the computational load of the SVD algorithm is $O\left(K^3\right)$. Therefore, assuming identical number of iterations per dimension, the asymptotic computational load of the algorithm per iteration is $O\left(MK + K^4\right)$.

The computational loads per iteration of the proposed method and of other existing techniques for approximate joint diagonalization, like Pham's [2], [25], AC/DC [27] and FFDIAG [29] algorithms are presented in Table 3.1.

TABLE 3.1

Asymptotic computational load per iteration of SVDJD, Pham's, AC/DC and FFDIAG algorithms.

ALGORITHM	COMPUTATIONAL LOAD
SVDJD	$O\left(MK + K^4\right)$
PHAM	$O\left(MK^2\right)$
AC/DC	$O\left(MK^3\right)$
FFDIAG	$O\left(MK^2\right)$

Observing Table 3.1, one can notice that the SVDJD is computationally the most efficient, in comparison to Pham's, AC/DC and FFDIAG algorithms, whenever $M > \frac{K^3}{K-1}$. This case may be encountered, for example, when K=5 and M=100.

3.6 SIMULATIONS

In this section, the performance of the SVDJD algorithm is compared with state-of-the-art algorithms for approximate joint diagonalization, like Pham's [2], [25], AC/DC [27] and the FFDIAG algorithm [29]. The first trial evaluates the performance of the SVDJD algorithm in a scenario of orthogonal joint diagonalization problem. The second trial evaluates the performance of the proposed method in the case of nonorthogonal joint diagonalization problem, and the third trial evaluates the performance of the proposed algorithm in BSS applications.

3.6.1 ORTHOGONAL JOINT DIAGONALIZATION

In this test the performance of the proposed algorithm was evaluated for orthogonal joint diagonalization. The data set consisted of 1000 random realizations of the real two-dimensional matrix set $\left\{\mathbf{A}\left(\mathbf{E}_m\boldsymbol{\Lambda}_m\mathbf{E}_m^H\right)\mathbf{A}^H\right\}_{m=1}^{M}$, where $\mathbf{A}$ is a rotation matrix with rotation angle uniformly distributed in $(-180^\circ,180^\circ]$, the matrices $\left\{\boldsymbol{\Lambda}_m\right\}_{m=1}^{M}$ are diagonal with elements uniformly distributed in $(0,1]$ and the matrices $\left\{\mathbf{E}_m\right\}_{m=1}^{M}$ are perturbation rotation matrices with rotation angles uniformly distributed in $(-10^\circ,10^\circ]$. The number of matrices, M was set to 50 such that the condition $M > \frac{K^3}{K-1}$ is satisfied. The estimated weights, $\left\{\hat{w}_m\right\}_{m=1}^{M}$, were equally set to $\frac{1}{M}$. In each realization, the values of the objective function $Q^*\left(\mathbf{B}\right)$, total running time and total running time per iteration were calculated for the SVDJD, Pham's, AC/DC and FFDIAG algorithms. Calculation of the running period per iteration of the SVDJD algorithm is given by

$$T_{It} = \frac{T_{Tot}\cdot K}{\sum_{k=1}^{K} L_k}, \tag{3.48}$$

where T_{It}, T_{Tot} and L_k denote the running time per iteration, the total running time, and the number of iterations in the k^{th} dimension, respectively. Calculation of the running time per iteration of Pham's, AC/DC and FFDIAG algorithms was performed by dividing the total running time by the number of iterations. The mean values, 5th and 95th percentiles of $Q^*\left(\mathbf{B}\right)$, estimated for each algorithm, are depicted in Fig. 15.a. The average running time per iteration and total running time of each algorithm are depicted in

Fig. 15.b. According to Fig. 15.a, the best diagonalization performance was obtained by Pham's algorithm. The SVDJD algorithm was superior to the AC/DC and FFDIAG algorithm, which performed approximately the same. The difference between the SVDJD and Pham's algorithms is caused by the fact that in each iteration Pham's algorithm is maximizing a lower bound on the decrease in $Q^*(\mathbf{B})$ (caused by a two-dimensional linear transformation of distinct rows in **B**), whereas the SVDJD algorithm is aiming to solve (3.19), where its solutions are the scaled rows of **B**. Fig. 15.b implies that the shortest averaged running time per iteration was obtained, as expected, by the SVDJD algorithm. It is also implied by Fig. 15.b that the averaged total running time was significantly shorter in the SVDJD algorithm.

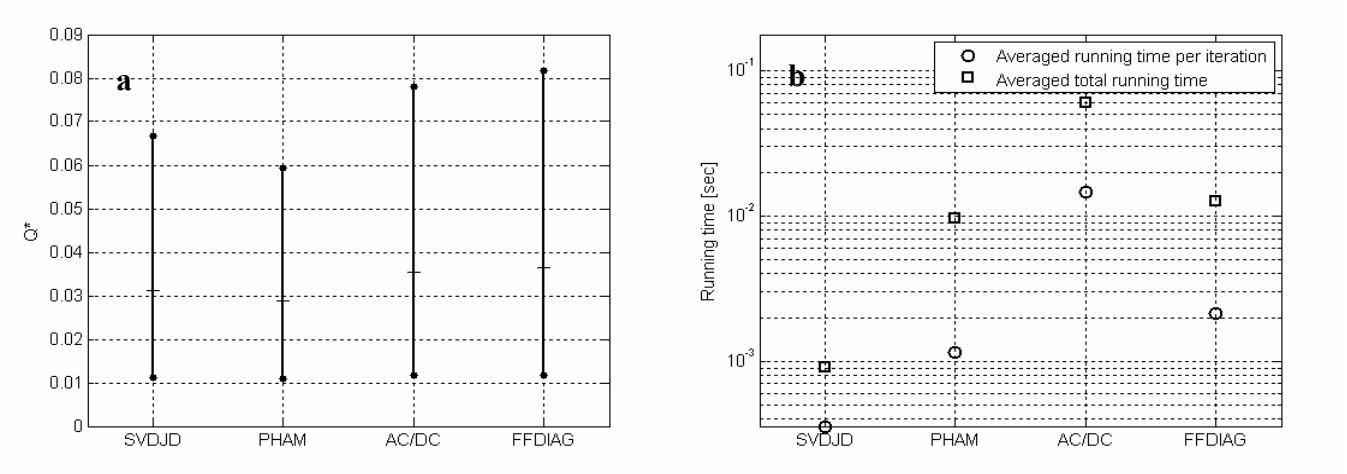

Fig. 15. a) The mean values, 5th and 95th percentiles of $Q^*(\mathbf{B})$ obtained by the SVDJD, Pham's, AC/DC and FFDIAG algorithms. The '$-$' mark denotes the mean value, and the lower and upper '•' marks denote the 5th and 95th percentiles, respectively. b) The averaged running time per iteration and the averaged total running time in seconds of each algorithm.

3.6.2 NONORTHOGONAL JOINT DIAGONALIZATION

In this test, the performance of the proposed algorithm was evaluated for nonorthogonal joint diagonalization problem.

Let σ^2 denote a perturbation level. For each $\sigma^2 = 0.1, 0.01, 0.001, 0.0001, 0.00001$, a data set consisting of 1000 random realizations of the five-dimensional matrix set $\left\{\mathbf{A}\mathbf{\Lambda}_m\mathbf{A}^H + \sigma^2\mathbf{E}_m\mathbf{E}_m^H\right\}_{m=1}^M$ was generated. The elements of the matrix $\mathbf{A}$ are drawn from a real standard normal distribution and the matrices $\left\{\mathbf{\Lambda}_m\right\}_{m=1}^M$ are real diagonal with elements uniformly distributed in $(0,1]$. The matrices $\left\{\mathbf{E}_m\right\}_{m=1}^M$ are perturbation matrices,

randomized from a real normal standard distribution. The number of matrices, *M*, was set to 100 such that the condition $M > \frac{K^3}{K-1}$ is satisfied and the estimated weights, $\{\hat{w}_m\}_{m=1}^{M}$, were equally set to $\frac{1}{M}$. In each realization, the values of the objective function $Q^*(\mathbf{B})$, total running time and total running time per iteration were calculated for the SVDJD, Pham's, AC/DC and FFDIAG algorithms. Calculation of the running time per iteration was performed as described in (3.48). The mean values, 5th and 95th percentiles of $Q^*(\mathbf{B})$, estimated for each algorithm for each perturbation level, are depicted in Fig. 16. The averaged running time per iteration of each algorithm as a function of the perturbation level, σ^2, are depicted in Fig. 16.a and the averaged total running time of each algorithm as a function of the perturbation level, σ^2, are depicted in Fig. 16.b.

According to Fig.16, one can observe that in low perturbation levels, the SVDJD algorithm outperforms the other algorithms. Fig. 17.a implies that the shortest running time per iteration was obtained by the SVDJD algorithm. According to Fig. 17.b the averaged total running time was significantly shorter in the SVDJD algorithm in comparison to Pham's, AC/DC and FFDIAG algorithms.

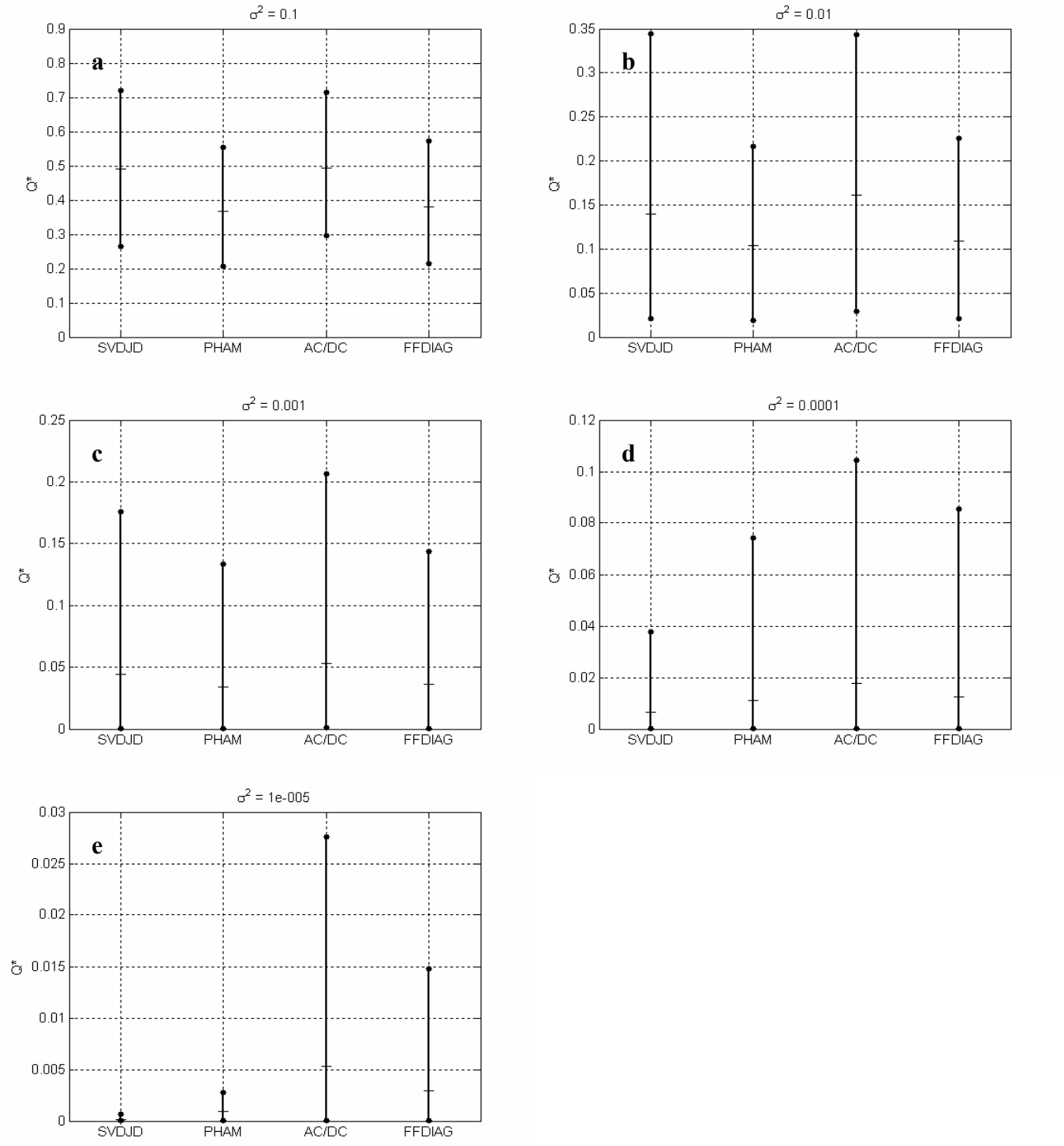

Fig. 16. The mean values, 5th and 95th percentiles of $Q^*(\mathbf{B})$ obtained by the SVDJD, Pham's, AC/DC and FFDIAG algorithms for various perturbation levels. The '—' mark denotes the mean value, and the lower and upper '•' marks denote the 5th and 95th percentiles, respectively.

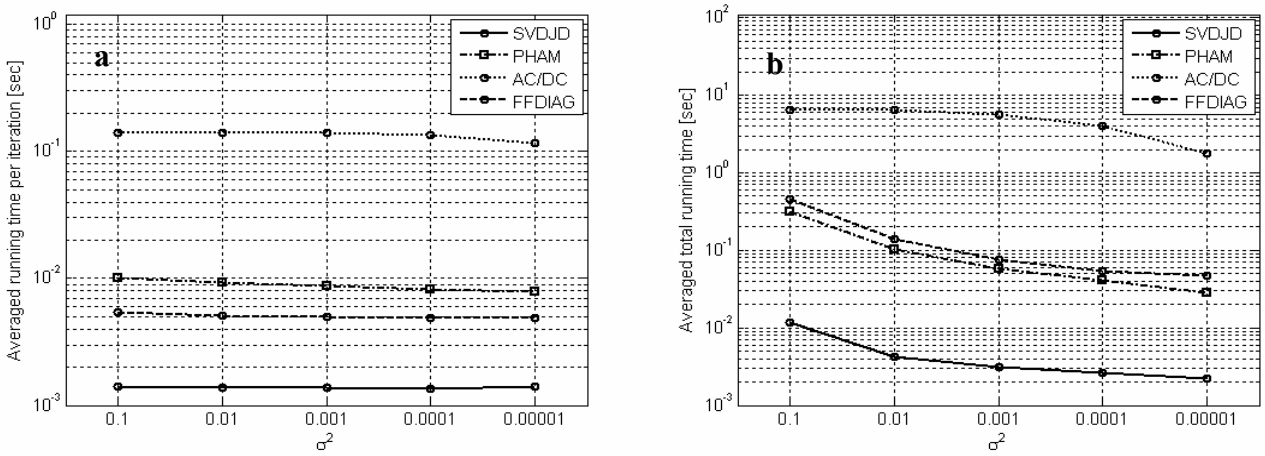

Fig. 17. The averaged running time per iteration (a) and the averaged total running time (b) of the SVDJD, Pham's, AC/DC and FFDIAG algorithms as a function of the perturbation level, σ^2.

3.6.3 BSS APPLICATION

In this test, the performance of the proposed algorithm was evaluated in solving a linear instantaneous noiseless BSS problem. Three ten-second long speech signals, sampled at 16 kHz and normalized to unit variance were mixed by a nonorthogonal 3×3 mixing matrix, **A**.

Blind separation of the mixed sources was performed in two manners. In the first manner, the non-Gaussianity of the independent sources was exploited while in the second manner the nonstationarity of independent sources was exploited.

In the first manner, in which non-Gaussianity of sources was exploited, the distribution of the mixed source signals was modeled by GMM of order 27 (with 27 full covariance matrices). The covariance matrices and weights were estimated via the greedy EM algorithm for GMM parameter estimation [12]. In [30] it was shown that the joint diagonalization matrix of the estimated covariance matrices is the estimated separation matrix, $\hat{\mathbf{B}}$. Joint diagonalization of the estimated covariance matrices was performed using the SVDJD, Pham's, AC/DC and FFDIAG algorithms. Blind separation performances were measured by means of ISR for each method. Calculation of ISR is described in Appendix D. The running time per iteration, total running time, diagonalization performance and the ISR obtained by each algorithm, are presented in Fig. 18. According to Fig. 18, the running time per iteration and total running time of the SVDJD algorithm were significantly shorter in comparison to the other algorithms. The best diagonalization performances were

obtained by the SVDJD and Pham's algorithms. The best separation performance was obtained by the SVDJD algorithm.

In the second manner, in which nonstationarity of the sources was exploited, the distribution of the mixed source signals was modeled according to the "block-Gaussian" model, described in [2]. According to this model, the mixed sources are partitioned into M consecutive quasi-stationary segments, where the relative proportion and the estimated covariance matrix of the m^{th} quasi-stationary segment are denoted by $\hat{w}_m$ and $\hat{\mathbf{R}}_m$, respectively. In [2] it was shown that the joint diagonalization matrix of the estimated covariance matrices is the estimated separation matrix, $\hat{\mathbf{B}}$. In this test, the mixed speech sources were partitioned into quasi-stationary segments with fixed duration using rectangular frame size of 15 msec with frame rate of 33%. The resulting model order (i.e. number of covariance matrices) was M=861 and the estimated weights, $\{\hat{w}_m\}_{m=1}^{M}$, were equally set to $\frac{1}{M}$. Joint diagonalization of the estimated covariance matrices was performed using the SVDJD, Pham's, AC/DC and FFDIAG algorithms and blind separation performances were measured by means of ISR. The running time per iteration, total running time, diagonalization performance and the ISR, obtained by each algorithm are presented in Fig. 19. According to Fig. 19, the running time per iteration and total running time of the SVDJD algorithm were significantly shorter in comparison to the other algorithms. The best diagonalization performances were obtained by the SVDJD and Pham's algorithms. The best separation performance was obtained by the SVDJD algorithm.

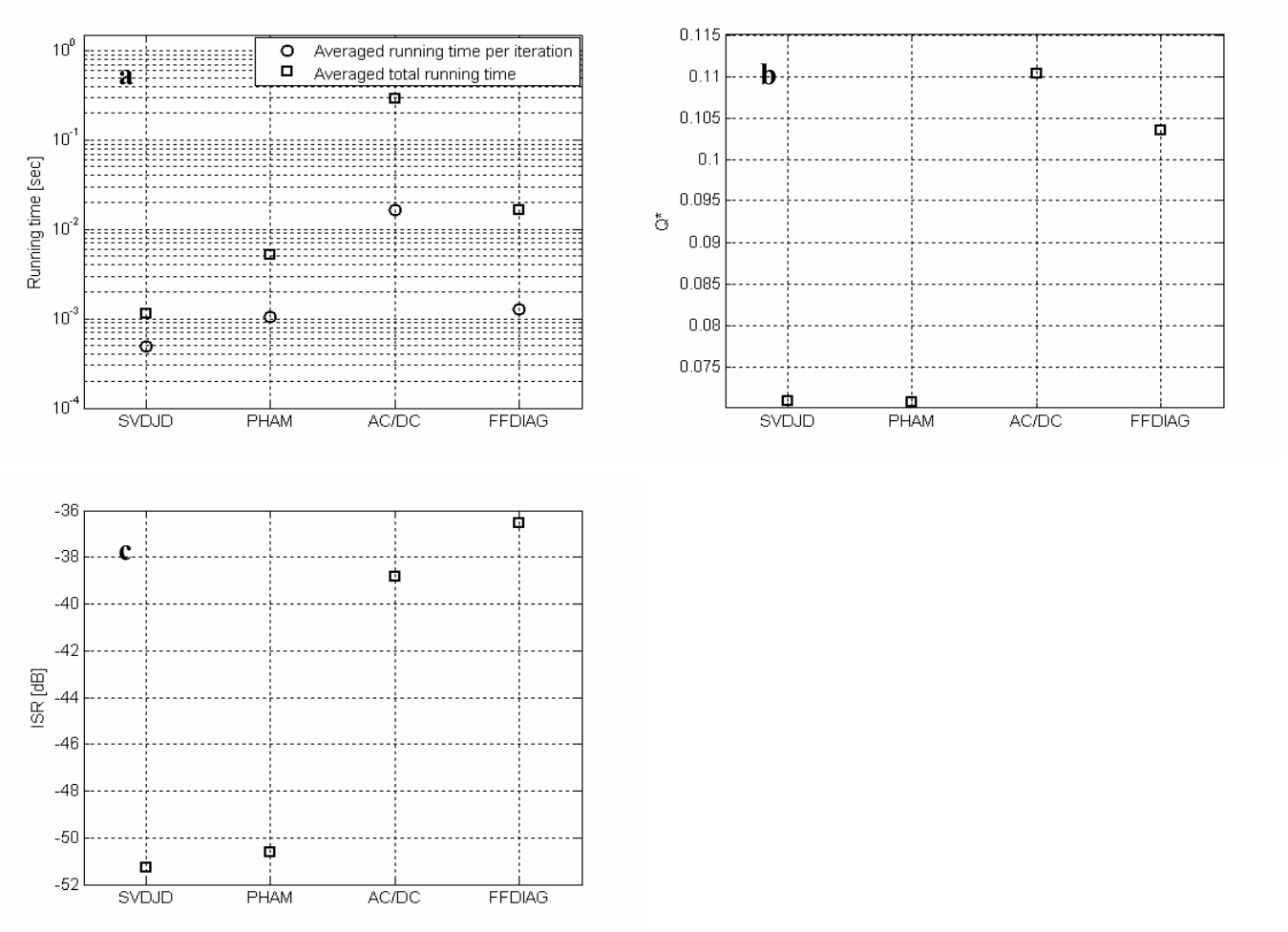

Fig. 18. a) The averaged running time per iteration and the averaged total running time of the compared algorithms. b) The values of the objective function, $Q^{*}\left(\mathbf{B}\right)$, obtained by each algorithm. c) The averaged ISR, obtained by each algorithm.

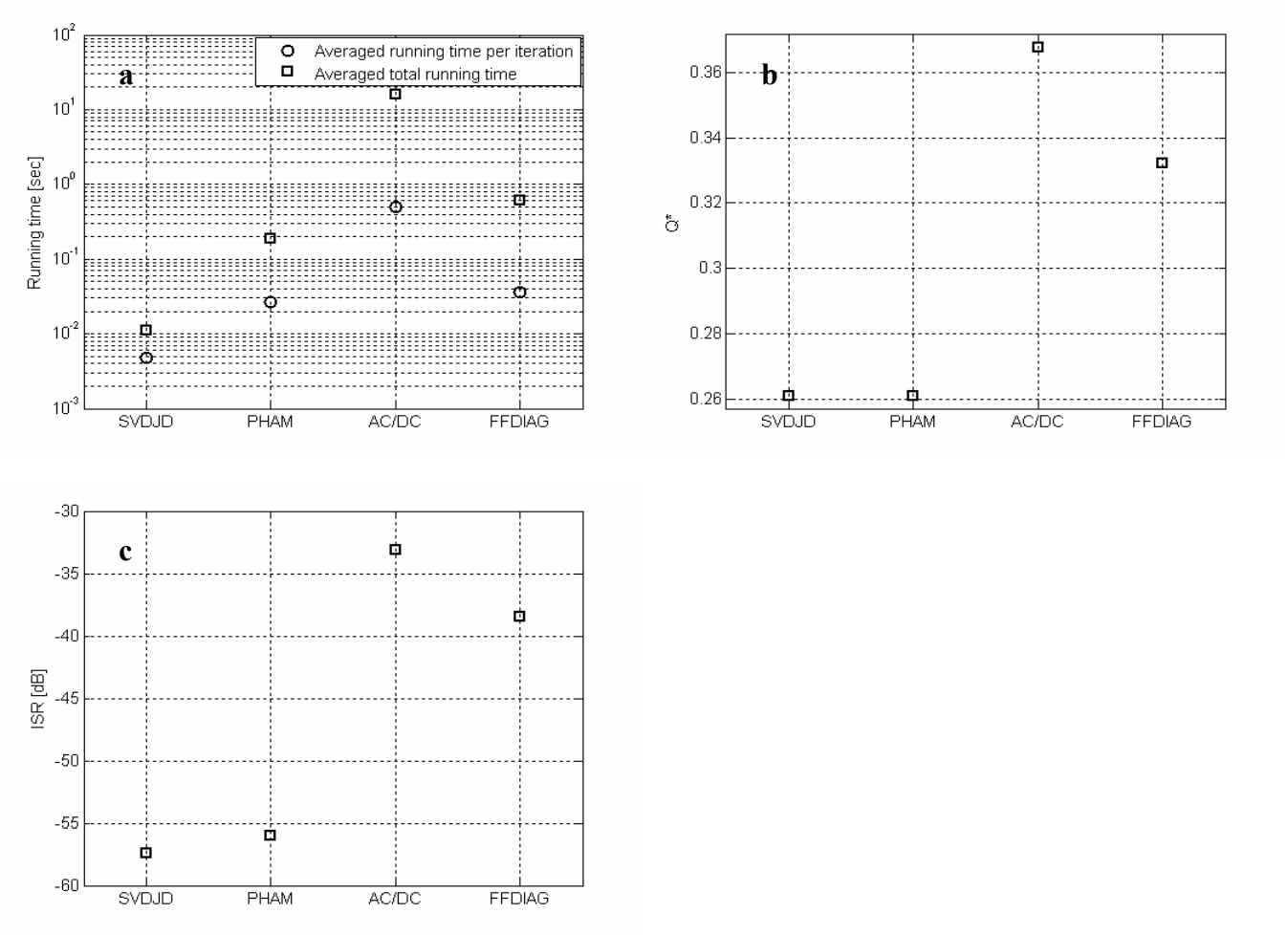

Fig. 19. a) The averaged running time per iteration and the averaged total running time of the compared algorithms. b) The values of the objective function, $Q^*(\mathbf{B})$, obtained by each algorithm. c) The averaged ISR, obtained by each algorithm.

3.7 DISCUSSION AND CONCLUSIONS

A new efficient algorithm, named as SVDJD, for approximate joint diagonalization of positive-definite Hermitian matrices is proposed. The diagonalization matrix, **B**, is not constrained to be unitary and it is estimated by iterative optimization of a ML-based objective function. Each column in **B** is estimated independently using iterative SVDs of a weighted sum of the matrices to be diagonalized.

The fact that each column in **B** is estimated independently enables low computational load which is practical especially in cases of large amount of matrices. The algorithm is easy to implement and demonstrates good diagonalization performance together with low computational load in comparison to existing state-of-the-art algorithms for approximate joint diagonalization. The convergence of the algorithm was studied and a sufficient condition for convergence was derived. Derivation of a necessary condition for convergence is an issue for further research. Finally, the practical use of the proposed algorithm is successfully exemplified in solving an instantaneous linear BSS problem.

4. Summary

4.1 CONCLUSIONS

Two novel ML-based algorithms for BSS of noiseless linear mixture of independent sources are proposed. The algorithms solve the BSS problem by estimation of the sensors distribution parameters via the EM algorithm for GMM parameter estimation, followed by estimation of the separation matrix via approximate joint diagonalization of the estimated GMM covariance matrices.

It was shown that estimation of the sensors distribution parameters via the EM algorithm amounts to obtaining a tight lower bound on the log-likelihood of a function of the separation matrix. It was also shown that joint diagonalization of the estimated GMM covariance matrices amounts to maximization of the obtained tight lower bound w.r.t. the separation matrix.

Generally, the GMMSVDJD and GMMPHAM are preferred upon the GMMFG algorithm since no prewhitening of the observations is performed. However, in cases where the number of observations is small, the use of the GMMFG algorithm is preferred upon the GMMSVDJD and GMMPHAM algorithms, since GMM parameter estimation is less erroneous when prewhitening of the observations is applied.

In comparison to methods like JADE [3] and FastICA [16], which use restrictive assumptions on the sources distribution, the proposed techniques are superior due to the fact that a flexible source density model is applied.

In contrast to our algorithms, the methods described in [4] - [8] utilize an EM algorithm, which jointly estimate the source distribution parameters and the mixing matrix coefficients. This approach has the following disadvantages. First, accurate initialization and order selection of the distribution model of the unobserved source signals is difficult, so the EM algorithm may converge into undesired maxima. Second, implementation of these approaches is complicated.

The learning rate factor in the NIFA algorithm is selected empirically and has a great effect on the separation performance. As observed by simulations, the NIFA algorithm is sensitive to model order selection in each dimension and cannot adapt the number of Gaussians in each direction. This drawback can cause model mismatch, which results in poor separation performance. Since the proposed methods do not estimate the PDF of each unobserved source, they are not affected by this drawback.

In the proposed methods, the distribution parameters of the observations are estimated apart from the separation matrix and therefore, GMM order selection using information theoretic criteria is trivial. In

contrast to this, optimal selection of GMM order for each unobserved source in the methods described in [4]-[8] is much more complicated.

Theoretically, the EM algorithm for GMM parameter estimation of the sensor signals would become intractable as the number of sources increases. This is because the number of Gaussians grows exponentially with the number of sources. For example, K=10 sources with n_k=3 Gaussians for each source, result $M=3^{10}$ Gaussians. However, according to the simulation results, it is observed that due to finite sample size, the determined GMM order in high dimensions is always much smaller than the theoretical number of Gaussians. This property enables the applicability of the proposed methods also for large number of sources.

Finally, according to simulation results, the proposed BSS algorithms demonstrate superior separation performances in comparison to existing methods. However, this superiority comes at the expense of higher computational load, caused by the use of the EM algorithm for GMM parameter estimation [11].

A new efficient algorithm, named as SVDJD, for approximate joint diagonalization of positive-definite Hermitian matrices is proposed. The diagonalization matrix, **B**, is not constrained to be unitary and it is estimated by iterative optimization of an ML-based objective function.

Each column in **B** is estimated independently using iterative SVDs of a weighted sum of the matrices to be diagonalized. The fact that each column in **B** is estimated independently enables low computational load which is practical especially in cases of large amount of matrices.

The algorithm is easy to implement and demonstrates good diagonalization performance together with low computational load in comparison to existing state-of-the-art algorithms for approximate joint diagonalization. The convergence of the algorithm was studied and a sufficient condition for convergence was derived. Derivation of a necessary condition for convergence is an issue for further research. The practical use of the proposed algorithm is successfully exemplified in solving an instantaneous linear BSS problem.

4.2 FUTURE RESEARCH

In this work, the GMM was utilized for blind separation of instantaneous noiseless linear mixture of independent sources, where the number of sensors was equal or greater than the number of sources. Therefore, future research should include utilization of GMM for blind source separation, where the number of sources is greater than the number of sensors.

A new efficient algorithm for approximate joint diagonalization of positive-definite Hermitian matrices was also presented in this work. The convergence of the algorithm was studied and a sufficient condition for convergence was derived. Therefore, further research should include derivation of a necessary condition for convergence.

Appendix A

In this appendix, it is shown that by utilizing the EM algorithm for GMM parameter estimation [11], a tight lower bound on $\log f_{\mathbf{X};\boldsymbol{\theta}^{(\mathbf{x})}}$ is obtained. First, a lower bound on $\log f_{\mathbf{X};\boldsymbol{\theta}^{(\mathbf{x})}}$ is derived. Then, an expression of a tight lower bound on $\log f_{\mathbf{X};\boldsymbol{\theta}^{(\mathbf{x})}}$ is derived. Finally, it is shown that the lower bound can be tightened iteratively by utilizing the EM algorithm [10].

A.1 DERIVATION OF A LOWER BOUND ON THE LOG-LIKELIHOOD FUNCTION

Claim A.1 introduces a lower bound on the log-likelihood function of the observation signals, given their distribution parameters. It is shown that this lower bound is tangent to the log-likelihood function on one point.

Claim A.1:

Let an arbitrary vector of distribution parameters of the observation signals be denoted by $\boldsymbol{\theta}_{arb}^{(\mathbf{x})}$*, then* $\forall\ \boldsymbol{\theta}^{(\mathbf{x})}$ *a lower bound on* $\log f_{\mathbf{X};\boldsymbol{\theta}^{(\mathbf{x})}}$ *is given by*

$$\log f_{\mathbf{X};\boldsymbol{\theta}^{(\mathbf{x})}} \geq \log f_{\mathbf{X};\boldsymbol{\theta}_{arb}^{(\mathbf{x})}} + E_{\mathbf{Y}|\mathbf{X};\boldsymbol{\theta}_{arb}^{(\mathbf{x})}}\left[\log \frac{f_{\mathbf{X},\mathbf{Y};\boldsymbol{\theta}^{(\mathbf{x})}}}{f_{\mathbf{X},\mathbf{Y};\boldsymbol{\theta}_{arb}^{(\mathbf{x})}}}\right] = D\left(\boldsymbol{\theta}^{(\mathbf{x})}, \boldsymbol{\theta}_{arb}^{(\mathbf{x})}\right), \tag{A.1}$$

where $\log f_{\mathbf{X},\mathbf{Y};\boldsymbol{\theta}^{(\mathbf{x})}}$ *is the joint log-likelihood of* $\boldsymbol{\theta}^{(\mathbf{x})}$ *with matrices of observation vectors and of their corresponding hidden indication vectors,* $\mathbf{X} = [\mathbf{x}_1, ..., \mathbf{x}_T]$ *and* $\mathbf{Y} = [\mathbf{y}_1, ..., \mathbf{y}_T]$*, respectively. E denotes the expectation operator.*

Proof A.1:

It is implied by the Bayes theorem that $\log f_{\mathbf{X},\mathbf{Y};\boldsymbol{\theta}^{(\mathbf{x})}} = \log f_{\mathbf{X};\boldsymbol{\theta}^{(\mathbf{x})}} + \log f_{\mathbf{Y}|\mathbf{X};\boldsymbol{\theta}^{(\mathbf{x})}}$*. Hence,*

$$\log f_{\mathbf{X};\boldsymbol{\theta}^{(\mathbf{x})}} = \log f_{\mathbf{X};\boldsymbol{\theta}_{arb}^{(\mathbf{x})}} + E_{\mathbf{Y}|\mathbf{X};\boldsymbol{\theta}_{arb}^{(\mathbf{x})}}\left[\log \frac{f_{\mathbf{X},\mathbf{Y};\boldsymbol{\theta}^{(\mathbf{x})}}}{f_{\mathbf{X},\mathbf{Y};\boldsymbol{\theta}_{arb}^{(\mathbf{x})}}}\right] + E_{\mathbf{Y}|\mathbf{X};\boldsymbol{\theta}_{arb}^{(\mathbf{x})}}\left[\log \frac{f_{\mathbf{Y}|\mathbf{X};\boldsymbol{\theta}_{arb}^{(\mathbf{x})}}}{f_{\mathbf{Y}|\mathbf{X};\boldsymbol{\theta}^{(\mathbf{x})}}}\right]. \tag{A.2}$$

The last term of (A.2) is the Kullback-Leibler (KL) divergence of $f_{\mathbf{Y}|\mathbf{X};\boldsymbol{\theta}^{(\mathbf{x})}}$ *from* $f_{\mathbf{Y}|\mathbf{X};\boldsymbol{\theta}_{arb}^{(\mathbf{x})}}$*, which is always non-negative. Thus, by removing this term from (A.2), the following lower bound is obtained.*

$$\log f_{\mathbf{X};\boldsymbol{\theta}^{(\mathbf{x})}} \geq \log f_{\mathbf{X};\boldsymbol{\theta}_{arb}^{(\mathbf{x})}} + E_{\mathbf{Y}|\mathbf{X};\boldsymbol{\theta}_{arb}^{(\mathbf{x})}}\left[\log \frac{f_{\mathbf{X},\mathbf{Y};\boldsymbol{\theta}^{(\mathbf{x})}}}{f_{\mathbf{X},\mathbf{Y};\boldsymbol{\theta}_{arb}^{(\mathbf{x})}}}\right] = D\left(\boldsymbol{\theta}^{(\mathbf{x})}, \boldsymbol{\theta}_{arb}^{(\mathbf{x})}\right) \square \tag{A.3}$$

According to (A.3), one can notice the $D\left(\boldsymbol{\theta}^{(\mathbf{x})}, \boldsymbol{\theta}_{arb}^{(\mathbf{x})}\right)$ is tangent to $\log f_{\mathbf{X};\boldsymbol{\theta}^{(\mathbf{x})}}$ only when $\boldsymbol{\theta}^{(\mathbf{x})} = \boldsymbol{\theta}_{arb}^{(\mathbf{x})}$.

A.2 DERIVATION OF A TIGHT LOWER BOUND ON THE LOG-LIKELIHOOD FUNCTION

Claim A.2 introduces a *tight* lower bound on the log-likelihood function of the observation signals, given their distribution parameters. It is shown that this lower bound and the log-likelihood function share the same maximum.

Claim A.2:

A tight lower bound on $\log f_{\mathbf{X};\boldsymbol{\theta}^{(\mathbf{x})}}$ *is given by*

$$\log f_{\mathbf{X};\boldsymbol{\theta}^{(\mathbf{x})}} \geq D\left(\boldsymbol{\theta}^{(\mathbf{x})}, \hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right), \tag{A.4}$$

such that $\max_{\boldsymbol{\theta}^{(\mathbf{x})}} D\left(\boldsymbol{\theta}^{(\mathbf{x})}, \hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right) = D\left(\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}, \hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right) = \log f_{\mathbf{X};\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}}$. *The vector* $\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})} = \arg\max_{\boldsymbol{\theta}^{(\mathbf{x})}} \log f_{\mathbf{X};\boldsymbol{\theta}^{(\mathbf{x})}}$ *is the maximum likelihood estimate of* $\boldsymbol{\theta}^{(\mathbf{x})}$.

Proof A.2:

According to the ML estimator and (A.3)

$$\max_{\boldsymbol{\theta}^{(\mathbf{x})}} \log f_{\mathbf{X};\boldsymbol{\theta}^{(\mathbf{x})}} = \log f_{\mathbf{X};\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}} = D\left(\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}, \hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right). \tag{A.5}$$

and

$$\log f_{\mathbf{X};\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}} \geq D\left(\boldsymbol{\theta}^{(\mathbf{x})}, \boldsymbol{\theta}_{ML}^{(\mathbf{x})}\right) \quad \forall \boldsymbol{\theta}^{(\mathbf{x})}. \tag{A.6}$$

Therefore,

$$D\left(\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}, \hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right) \geq D\left(\boldsymbol{\theta}^{(\mathbf{x})}, \hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right) \quad \forall \boldsymbol{\theta}^{(\mathbf{x})}. \tag{A.7}$$

Hence, according to (A.5) and (A.7)

$$\max_{\boldsymbol{\theta}^{(\mathbf{x})}} D\left(\boldsymbol{\theta}^{(\mathbf{x})}, \hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right) = D\left(\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}, \hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right) = \log f_{\mathbf{X};\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}}. \tag{A.8}$$

Thus, $D\left(\boldsymbol{\theta}^{(\mathbf{x})}, \hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)$ *is a tight lower bound on* $\log f_{\mathbf{X};\boldsymbol{\theta}^{(\mathbf{x})}}$ □

A.3 UTILIZATION OF THE EM ALGORITHM FOR LOWER BOUND TIGHTENING

Now, it is shown that the lower bound on $\log f_{\mathbf{X};\boldsymbol{\theta}^{(\mathbf{x})}}$ can be tightened iteratively by utilizing the EM algorithm [10]. Since the entries of $\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}$ are initially unknown, $D\left(\boldsymbol{\theta}^{(\mathbf{x})}, \hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)$ cannot be obtained in one step. Therefore, the lower bound on $\log f_{\mathbf{X};\boldsymbol{\theta}^{(\mathbf{x})}}$ is tightened iteratively. The EM algorithm [11] is an iterative numerical method for maximum likelihood estimation, in which the following two dependent steps are performed in each iteration:

E-step: In this step, $D\left(\boldsymbol{\theta}^{(\mathbf{x})}, \boldsymbol{\theta}_{n-1}^{(\mathbf{x})}\right)$ is calculated according to (A.3), where *n* denotes an iteration index (note that in the first iteration (*n*=1) $\boldsymbol{\theta}_{n-1}^{(\mathbf{x})} = \boldsymbol{\theta}_{0}^{(\mathbf{x})}$ is required to be initialized).

M-step: In this step,

$$\boldsymbol{\theta}_{n}^{(\mathbf{x})} = \arg\max_{\boldsymbol{\theta}^{(\mathbf{x})}} D\left(\boldsymbol{\theta}^{(\mathbf{x})}, \boldsymbol{\theta}_{n-1}^{(\mathbf{x})}\right) \tag{A.9}$$

is estimated.

In the following, we show that each EM iteration tightens the lower bound on $\log f_{\mathbf{X};\boldsymbol{\theta}^{(\mathbf{x})}}$.

Claim A.3:

Let n denote an EM iteration index, then

$$D\left(\boldsymbol{\theta}_{n+1}^{(\mathbf{x})}, \boldsymbol{\theta}_{n}^{(\mathbf{x})}\right) \geq D\left(\boldsymbol{\theta}_{n}^{(\mathbf{x})}, \boldsymbol{\theta}_{n-1}^{(\mathbf{x})}\right). \tag{A.10}$$

Proof A.3:

According to (A.9) $D\left(\boldsymbol{\theta}_{n+1}^{(\mathbf{x})}, \boldsymbol{\theta}_{n}^{(\mathbf{x})}\right) = \max_{\boldsymbol{\theta}^{(\mathbf{x})}} D\left(\boldsymbol{\theta}^{(\mathbf{x})}, \boldsymbol{\theta}_{n}^{(\mathbf{x})}\right)$. Therefore,

$$D\left(\boldsymbol{\theta}_{n+1}^{(\mathbf{x})}, \boldsymbol{\theta}_{n}^{(\mathbf{x})}\right) \geq D\left(\boldsymbol{\theta}_{n}^{(\mathbf{x})}, \boldsymbol{\theta}_{n}^{(\mathbf{x})}\right). \tag{A.11}$$

According to (A.3) $\log f_{\mathbf{X};\boldsymbol{\theta}_n^{(\mathbf{x})}} \geq D\left(\boldsymbol{\theta}_{n}^{(\mathbf{x})}, \boldsymbol{\theta}_{n-1}^{(\mathbf{x})}\right)$ and $\log f_{\mathbf{X};\boldsymbol{\theta}_n^{(\mathbf{x})}} = D\left(\boldsymbol{\theta}_{n}^{(\mathbf{x})}, \boldsymbol{\theta}_{n}^{(\mathbf{x})}\right)$. Therefore,

$$D\left(\boldsymbol{\theta}_{n+1}^{(\mathbf{x})}, \boldsymbol{\theta}_{n}^{(\mathbf{x})}\right) \geq D\left(\boldsymbol{\theta}_{n}^{(\mathbf{x})}, \boldsymbol{\theta}_{n-1}^{(\mathbf{x})}\right) \square \tag{A.12}$$

Thus, each EM iteration tightens the lower bound on $\log f_{\mathbf{X};\boldsymbol{\theta}^{(\mathbf{x})}}$. Fig. 20 illustrates the lower bound tightening process.

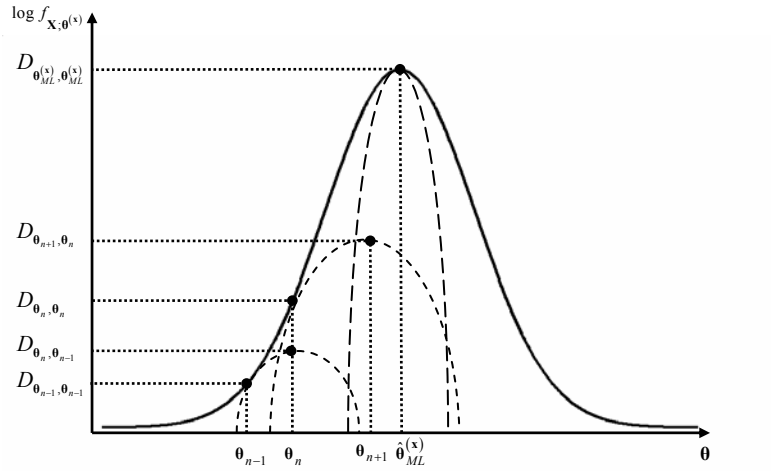

Fig. 20. An illustration of the log-likelihood function (solid line) and its lower bounds tightening (dashed curves).

Appendix B

In this appendix, it is shown that (2.27) can be formulated as described in (2.28).

Claim B.1:

Let $Q'\left(\boldsymbol{\rho}\left(\mathbf{B},\boldsymbol{\theta}^{(\mathbf{s})}\right),\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)=-\frac{1}{T}\sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_{t,m}\log\left(\hat{w}_m N^c\left(\mathbf{x}_t,\mathbf{B}^{-1}\boldsymbol{\mu}_m,\mathbf{B}^{-1}\mathbf{C}_m\mathbf{B}^{-H}\right)\right)$. *Then*

$$Q'\left(\boldsymbol{\rho}\left(\mathbf{B},\boldsymbol{\theta}^{(\mathbf{s})}\right),\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)=\sum_{m=1}^{M}\hat{w}_m\left\{KL_{norm}\left[\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^{\mathrm{T}}\mid\mathbf{C}_m\right]+tr\left[\left(\boldsymbol{\mu}_m-\mathbf{B}\hat{\boldsymbol{\eta}}_m\right)^H\mathbf{C}_m^{-1}\left(\boldsymbol{\mu}_m-\mathbf{B}\hat{\boldsymbol{\eta}}_m\right)\right]\right\}+const. \tag{B.1}$$

Proof B.1:

According to (2.12) and (2.24)

$$Q'\left(\boldsymbol{\theta}^{(\mathbf{x})},\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)=-\frac{1}{T}\sum_{t=1}^{T}\sum_{m=1}^{M}\gamma_{t,m}\log\left(\hat{w}_m N^c\left(\mathbf{x}_t,\boldsymbol{\eta}_m,\mathbf{R}_m\right)\right). \tag{B.2}$$

Therefore,

$$\begin{aligned}&Q\left(\boldsymbol{\theta}^{(\mathbf{x})},\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)=\\&\frac{1}{T}\sum_{m=1}^{M}\left\{\sum_{t=1}^{T}\gamma_{t,m}\log\left(\det\left(\pi\mathbf{R}_m\right)\right)+\sum_{t=1}^{T}\gamma_{t,m}tr\left[\mathbf{R}_m^{-1}\left(\mathbf{x}_t-\boldsymbol{\eta}_m\right)\left(\mathbf{x}_t-\boldsymbol{\eta}_m\right)^H\right]-\sum_{t=1}^{T}\gamma_{t,m}\log\hat{w}_m\right\}.\end{aligned} \tag{B.3}$$

Since trace is a linear operator, the summation w.r.t. t in the mid-term of (B.3) can be inserted into the trace operator and (B.3) can be rewritten in the following manner

$$\begin{aligned}&Q\left(\boldsymbol{\theta}^{(\mathbf{x})},\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)=\\&\frac{1}{T}\sum_{m=1}^{M}\left\{\sum_{t=1}^{T}\gamma_{t,m}\log\left(\det\left(\pi\mathbf{R}_m\right)\right)+tr\left[\mathbf{R}_m^{-1}\sum_{\tau=1}^{T}\gamma_{\tau,m}\left(\mathbf{x}_\tau-\boldsymbol{\eta}_m\right)\left(\mathbf{x}_\tau-\boldsymbol{\eta}_m\right)^H\right]-\sum_{t=1}^{T}\gamma_{t,m}\log\hat{w}_m\right\}.\end{aligned} \tag{B.4}$$

The factor $\sum_{t=1}^{T}\gamma_{t,m}$ *can be extracted out of the main brackets and (B.4) can be formulated as follows*

$$Q\left(\boldsymbol{\theta}^{(\mathbf{x})},\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)=\frac{1}{T}\sum_{m=1}^{M}\sum_{t=1}^{T}\gamma_{t,m}\left\{tr\left[\mathbf{R}_m^{-1}\underbrace{\left(\frac{\sum_{\tau=1}^{T}\gamma_{\tau,m}\left(\mathbf{x}_\tau-\boldsymbol{\eta}_m\right)\left(\mathbf{x}_\tau-\boldsymbol{\eta}_m\right)^H}{\sum_{\tau=1}^{T}\gamma_{\tau,m}}\right)}_{\mathbf{J}_m}\right]+\log\left(\det\left(\pi\mathbf{R}_m\right)\right)-\log\hat{w}_m\right\}. \tag{B.5}$$

The updating equations of the EM algorithm for GMM parameter estimation [11] imply that

$$\hat{\mathbf{R}}_m+\hat{\boldsymbol{\eta}}_m\hat{\boldsymbol{\eta}}_m^H=\frac{\sum_{t=1}^{T}\gamma_{t,m}\mathbf{x}_t\mathbf{x}_t^H}{\sum_{t=1}^{T}\gamma_{t,m}},\ \hat{\boldsymbol{\eta}}_m=\frac{\sum_{t=1}^{T}\gamma_{t,m}\mathbf{x}_t}{\sum_{t=1}^{T}\gamma_{t,m}}\ \text{and}\ \hat{w}_m=\frac{1}{T}\sum_{t=1}^{T}\gamma_{t,m}.$$

Therefore,

$$\mathbf{J}_m=\hat{\mathbf{R}}_m+\hat{\boldsymbol{\eta}}_m\hat{\boldsymbol{\eta}}_m^H-\boldsymbol{\eta}_m\hat{\boldsymbol{\eta}}_m^H-\hat{\boldsymbol{\eta}}_m\boldsymbol{\eta}_m^H+\boldsymbol{\eta}_m\boldsymbol{\eta}_m^H=\hat{\mathbf{R}}_m+\left(\hat{\boldsymbol{\eta}}_m-\boldsymbol{\eta}_m\right)\left(\hat{\boldsymbol{\eta}}_m-\boldsymbol{\eta}_m\right)^H. \tag{B.6}$$

Substitution of (B.6) into (B.5), yields the following expression:

$$Q\left(\boldsymbol{\theta}^{(\mathbf{x})},\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)=\sum_{m=1}^{M}\hat{w}_m\underbrace{\left\{tr\left[\mathbf{R}_m^{-1}\hat{\mathbf{R}}_m\right]+\left(\hat{\boldsymbol{\eta}}_m-\boldsymbol{\eta}_m\right)^H\mathbf{R}_m^{-1}\left(\hat{\boldsymbol{\eta}}_m-\boldsymbol{\eta}_m\right)+\log\left(\det\left(\pi\mathbf{R}_m\right)\right)-\log\hat{w}_m\right\}}_{g_m}. \tag{B.7}$$

Applying that $\boldsymbol{\eta}_m=\mathbf{B}^{-1}\boldsymbol{\mu}_m$ *and* $\mathbf{R}_m=\mathbf{B}^{-1}\mathbf{C}_m\mathbf{B}^{-H}$,

$$\begin{aligned}g_m=&tr\left[\left(\mathbf{B}^{-1}\mathbf{C}_m\mathbf{B}^{-H}\right)^{-1}\hat{\mathbf{R}}_m\right]+\left(\hat{\boldsymbol{\eta}}_m-\mathbf{B}^{-1}\boldsymbol{\mu}_m\right)^H\left(\mathbf{B}^{-1}\mathbf{C}_m\mathbf{B}^{-H}\right)^{-1}\left(\hat{\boldsymbol{\eta}}_m-\mathbf{B}^{-1}\boldsymbol{\mu}_m\right)\\&+K\log\pi-\log\left(\det\left(\left(\mathbf{B}^{-1}\mathbf{C}_m\mathbf{B}^{-H}\right)^{-1}\right)\right)-\log\left(\det\left(\hat{\mathbf{R}}_m\right)\right)+\log\left(\det\left(\hat{\mathbf{R}}_m\right)\right)-K+K-\log\hat{w}_m,\end{aligned} \tag{B.8}$$

where the number of sources is denoted by K. *Let the Kullback-Leibler divergence [20] of* $N^c\left(\mathbf{0},\boldsymbol{\Sigma}_2\right)$ *from* $N^c\left(\mathbf{0},\boldsymbol{\Sigma}_1\right)$ *be expressed as*

$$KL_{norm}\left[\boldsymbol{\Sigma}_1\,|\,\boldsymbol{\Sigma}_2\right]=tr\left[\boldsymbol{\Sigma}_2^{-1}\boldsymbol{\Sigma}_1\right]-\log\left(\det\left(\boldsymbol{\Sigma}_2^{-1}\boldsymbol{\Sigma}_1\right)\right)-K, \tag{B.9}$$

then (B.8) can be formulated in the following manner

$$g_m = \underbrace{tr\left[\mathbf{C}_m^{-1}\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right] - \log\left(\det\left(\mathbf{C}_m^{-1}\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right)\right) - K}_{KL_{norm}\left[\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H|\mathbf{C}_m\right]} + \left(\boldsymbol{\mu}_m - \mathbf{B}\hat{\boldsymbol{\eta}}_m\right)^H \mathbf{C}_m^{-1}\left(\boldsymbol{\mu}_m - \mathbf{B}\hat{\boldsymbol{\eta}}_m\right) \quad \text{(B.10)}$$
$$+ \underbrace{K\log\pi e + \log\left(\det\left(\hat{\mathbf{R}}_m\right)\right) - \log\hat{w}_m}_{q_m},$$

where q_m denotes a constant. Therefore, by inserting (B.10) into (B.7) and by applying that $\boldsymbol{\theta}^{(\mathbf{x})} = \boldsymbol{\rho}\left(\mathbf{B}, \boldsymbol{\theta}^{(\mathbf{s})}\right)$, (B.1) is derived □

Appendix C

In this appendix, it is shown that the objective functions (2.30) and (3.11) can be rewritten in the manner described in (2.31) and (3.12).

Claim C.1:

Let $Q^*\left(\mathbf{B},\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)=\sum_{m=1}^{M}\hat{w}_m KL_{norm}\left[\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H \mid \mathrm{DIAG}\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right)\right]$, *then*

$$Q^*\left(\mathbf{B},\hat{\boldsymbol{\theta}}_{ML}^{(\mathbf{x})}\right)=\sum_{m=1}^{M}\hat{w}_m\left[\log\left(\det\left(\mathrm{DIAG}\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right)\right)\right)-\log\left(\det\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right)\right)\right], \tag{C.1}$$

where $\det(\cdot)$ denotes the determinant operator.

Proof C.1:

According to (B.9)

$$KL_{norm}\left[\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H \mid \mathrm{DIAG}\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right)\right]=tr\left[\left(\mathrm{DIAG}\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right)\right)^{-1}\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right)\right] - \log\left(\det\left(\left(\mathrm{DIAG}\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right)\right)^{-1}\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right)\right)\right)-K. \tag{C.2}$$

Let $\mathbf{G}=\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H$, *then by applying* $tr\left[\left(\mathrm{DIAG}(\mathbf{G})\right)^{-1}\mathbf{G}\right]=K$, *(C.2) is reduced to*

$$KL_{norm}\left[\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H \mid \mathrm{DIAG}\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right)\right]=-\log\left(\det\left(\left(\mathrm{DIAG}\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right)\right)^{-1}\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right)\right)\right). \tag{C.3}$$

Therefore, according to (C.3) , (C.1) is easily verified □

Appendix D

In this appendix, calculation of ISR is presented. Let the mixing and estimated separation matrices be denoted by **A** and **B**, respectively. According to (2.1) and (2.2), the k^{th} element of $\hat{\mathbf{s}}_t = \mathbf{B}\mathbf{x}_t = \mathbf{B}\mathbf{A}\mathbf{s}_t$, contains the signal of interest $s_{k,t}$ at power of $(\mathbf{BA})_{kk}^2 \cdot \mathrm{var}\left[s_{k,t}\right]$ and the j^{th} interfering signal at power of $(\mathbf{BA})_{kj}^2 \cdot \mathrm{var}\left[s_{j,t}\right]$. Therefore, the interference-to-signal ratio (ISR) is calculated in the following manner

$$ISR = 10\log_{10}\left(\frac{1}{K}\sum_{k=1}^{K}\sum_{j\neq k}^{K}\frac{(\mathbf{BA})_{kj}^2\ \mathrm{var}\left[s_{j,t}\right]}{(\mathbf{BA})_{kk}^2\ \mathrm{var}\left[s_{k,t}\right]}\right). \tag{D.1}$$

Appendix E

In this appendix, it is shown that (3.8) can be formulated as described in (3.9).

Claim E.1

Let $Q\left(\mathbf{B},\boldsymbol{\Lambda}_1,\boldsymbol{\Lambda}_2,...,\boldsymbol{\Lambda}_M\right)=\sum_{m=1}^{M}w_m\left[tr\left(\mathbf{B}^H\boldsymbol{\Lambda}_m^{-1}\mathbf{B}\hat{\mathbf{R}}_m\right)-\log\left(\det\left(\mathbf{B}^H\boldsymbol{\Lambda}_m^{-1}\mathbf{B}\right)\right)\right]+\log c$. *Then*

$$Q\left(\mathbf{B},\boldsymbol{\Lambda}_1,\boldsymbol{\Lambda}_2,...,\boldsymbol{\Lambda}_M\right)=\sum_{m=1}^{M}w_m KL_{norm}\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\mid\boldsymbol{\Lambda}_m\right)+\log c, \tag{E.1}$$

where $KL_{norm}\left(\boldsymbol{\Sigma}_1\mid\boldsymbol{\Sigma}_2\right)$ *denotes the Kullback-Leibler divergence of* $N^c\left(\mathbf{0},\boldsymbol{\Sigma}_2\right)$ *from* $N^c\left(\mathbf{0},\boldsymbol{\Sigma}_1\right)$. *The argument c denotes a constant.*

Proof E.1

Equation (E.1) can be formulated in the following manner

$$\begin{aligned}&Q\left(\mathbf{B},\boldsymbol{\Lambda}_1,\boldsymbol{\Lambda}_2,...,\boldsymbol{\Lambda}_M\right)=\\&\sum_{m=1}^{M}w_m\left[tr\left(\mathbf{B}^H\boldsymbol{\Lambda}_m^{-1}\mathbf{B}\hat{\mathbf{R}}_m\right)-\log\left(\det\left(\mathbf{B}^H\boldsymbol{\Lambda}_m^{-1}\mathbf{B}\right)\right)-\log\left(\det\left(\hat{\mathbf{R}}_m\right)\right)+\log\left(\det\left(\hat{\mathbf{R}}_m\right)\right)-K+K\right]\\&+\log c.\end{aligned} \tag{E.2}$$

Since $\left\{\hat{\mathbf{R}}_m\right\}_{m=1}^{M}$ *and* K *are constant w.r.t.* $\mathbf{B}$ *and* $\left\{\boldsymbol{\Lambda}_m\right\}_{m=1}^{M}$,

$$Q\left(\mathbf{B},\boldsymbol{\Lambda}_1,\boldsymbol{\Lambda}_2,...,\boldsymbol{\Lambda}_M\right)=\sum_{m=1}^{M}w_m\left[tr\left(\boldsymbol{\Lambda}_m^{-1}\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right)-\log\left(\det\left(\boldsymbol{\Lambda}_m^{-1}\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\right)\right)-K\right]+c', \tag{E.3}$$

where c' *denotes a constant. Therefore, according to (B.9)*

$$Q\left(\mathbf{B},\boldsymbol{\Lambda}_1,\boldsymbol{\Lambda}_2,...,\boldsymbol{\Lambda}_M\right)=\sum_{m=1}^{M}w_m KL_{norm}\left(\mathbf{B}\hat{\mathbf{R}}_m\mathbf{B}^H\mid\boldsymbol{\Lambda}_m\right)+const \quad \square \tag{E.4}$$

Appendix F

In this appendix the Hessian submatrices, $\mathbf{H}_{1,1}$, $\mathbf{H}_{1,2}$, $\mathbf{H}_{1,3}$, $\mathbf{H}_{2,3}$ and $\mathbf{H}_{3,3}$, described in (3.27), are derived. It is noted that the complex derivatives are defined in [35].

According to (3.27),

$$\frac{\partial \psi}{\partial \mathbf{x}} = \mathbf{x}^H \left(\mathbf{G}\left(\mathbf{y}\right) - \mathbf{I}\right)^2 - \alpha \mathbf{x}^H . \tag{F.1}$$

Therefore,

$$\mathbf{H}_{1,1} = \left.\frac{\partial^2 \psi}{\partial \mathbf{x} \partial \mathbf{x}^H}\right|_{\substack{\mathbf{x}=\tilde{\mathbf{b}}_k^* \\ \mathbf{y}=\tilde{\mathbf{b}}_k^* \\ \alpha=\alpha^*=0}} = \left(\mathbf{G}\left(\tilde{\mathbf{b}}_k^*\right) - \mathbf{I}\right)^2 . \tag{F.2}$$

Equation (3.27) implies that

$$\frac{\partial^2 \psi}{\partial \mathbf{y} \partial \mathbf{x}^H} = \frac{\partial}{\partial \mathbf{y}}\left[\frac{\partial \psi}{\partial \mathbf{x}^H}\right] = \frac{\partial}{\partial \mathbf{y}}\left[\left(\mathbf{G}\left(\mathbf{y}\right) - \mathbf{I}\right)^2 \mathbf{x}\right]. \tag{F.3}$$

According to (3.20)

$$\left(\mathbf{G}\left(\mathbf{y}\right) - \mathbf{I}\right)^2 \mathbf{x} = \sum_{m=1}^{M}\sum_{n=1}^{M} \hat{w}_m \hat{w}_n \frac{\widehat{\mathbf{R}}_m \widehat{\mathbf{R}}_n \mathbf{x}}{\mathbf{y}^H \widehat{\mathbf{R}}_m \mathbf{y} \mathbf{y}^H \widehat{\mathbf{R}}_n \mathbf{y}} - 2\sum_{m=1}^{M} \hat{w}_m \frac{\widehat{\mathbf{R}}_m \mathbf{x}}{\mathbf{y}^H \widehat{\mathbf{R}}_m \mathbf{y}} + \mathbf{x}\widehat{\mathbf{R}} . \tag{F.4}$$

Therefore,

$$\frac{\partial}{\partial \mathbf{y}}\left[\left(\mathbf{G}\left(\mathbf{y}\right) - \mathbf{I}\right)^2 \mathbf{x}\right] = -\sum_{m=1}^{M}\sum_{n=1}^{M} \hat{w}_m \hat{w}_n \frac{\widehat{\mathbf{R}}_m \widehat{\mathbf{R}}_n \mathbf{x}\left(\mathbf{y}^H \widehat{\mathbf{R}}_m \left(\mathbf{y}^H \widehat{\mathbf{R}}_n \mathbf{y}\right) + \mathbf{y}^H \widehat{\mathbf{R}}_n \left(\mathbf{y}^H \widehat{\mathbf{R}}_m \mathbf{y}\right)\right)}{\left(\mathbf{y}^H \widehat{\mathbf{R}}_m \mathbf{y} \mathbf{y}^H \widehat{\mathbf{R}}_n \mathbf{y}\right)^2} + 2\sum_{m=1}^{M} \hat{w}_m \frac{\widehat{\mathbf{R}}_m \mathbf{x} \mathbf{y}^H \widehat{\mathbf{R}}_m}{\left(\mathbf{y}^H \widehat{\mathbf{R}}_m \mathbf{y}\right)^2}. \tag{F.5}$$

Hence,

$$\mathbf{H}_{1,2}=\left.\frac{\partial^2\psi}{\partial\mathbf{y}\partial\mathbf{x}^H}\right|_{\substack{\mathbf{x}=\tilde{\mathbf{b}}_k^*\\ \mathbf{y}=\tilde{\mathbf{b}}_k^*\\ \alpha=\alpha^*=0}}=-\sum_{m=1}^{M}\sum_{n=1}^{M}\hat{w}_m\hat{w}_n\frac{\widehat{\mathbf{R}}_m}{\tilde{\mathbf{b}}_k^{*H}\widehat{\mathbf{R}}_m\tilde{\mathbf{b}}_k^*}\left[\frac{\widehat{\mathbf{R}}_n\tilde{\mathbf{b}}_k^*\tilde{\mathbf{b}}_k^{*H}\widehat{\mathbf{R}}_n}{\left(\tilde{\mathbf{b}}_k^{*H}\widehat{\mathbf{R}}_n\tilde{\mathbf{b}}_k^*\right)^2}+\frac{\widehat{\mathbf{R}}_n\tilde{\mathbf{b}}_k^*\tilde{\mathbf{b}}_k^{*H}\widehat{\mathbf{R}}_m}{\tilde{\mathbf{b}}_k^{*H}\widehat{\mathbf{R}}_m\tilde{\mathbf{b}}_k^*\tilde{\mathbf{b}}_k^{*H}\widehat{\mathbf{R}}_n\tilde{\mathbf{b}}_k^*}\right]$$
$$+2\sum_{m=1}^{M}\hat{w}_m\frac{\widehat{\mathbf{R}}_m\tilde{\mathbf{b}}_k^*\tilde{\mathbf{b}}_k^{*H}\widehat{\mathbf{R}}_m}{\left(\tilde{\mathbf{b}}_k^{*H}\widehat{\mathbf{R}}_m\tilde{\mathbf{b}}_k^*\right)^2}$$
$$=-\underbrace{\sum_{m=1}^{M}\sum_{n=1}^{M}\hat{w}_m\hat{w}_n\frac{\widehat{\mathbf{R}}_m}{\tilde{\mathbf{b}}_k^{*H}\widehat{\mathbf{R}}_m\tilde{\mathbf{b}}_k^*}\frac{\widehat{\mathbf{R}}_n\tilde{\mathbf{b}}_k^*\tilde{\mathbf{b}}_k^{*H}\widehat{\mathbf{R}}_m}{\tilde{\mathbf{b}}_k^{*H}\widehat{\mathbf{R}}_m\tilde{\mathbf{b}}_k^*\tilde{\mathbf{b}}_k^{*H}\widehat{\mathbf{R}}_n\tilde{\mathbf{b}}_k^*}}_{\mathbf{Y}\left(\tilde{\mathbf{b}}_k^*\right)}$$
$$-\underbrace{\sum_{m=1}^{M}\hat{w}_m\frac{\widehat{\mathbf{R}}_m}{\tilde{\mathbf{b}}_k^{*H}\widehat{\mathbf{R}}_m\tilde{\mathbf{b}}_k^*}}_{\mathbf{G}\left(\tilde{\mathbf{b}}_k^*\right)}\underbrace{\sum_{n=1}^{M}\hat{w}_n\frac{\widehat{\mathbf{R}}_n\tilde{\mathbf{b}}_k^*\tilde{\mathbf{b}}_k^{*H}\widehat{\mathbf{R}}_n}{\left(\tilde{\mathbf{b}}_k^{*H}\widehat{\mathbf{R}}_n\tilde{\mathbf{b}}_k^*\right)^2}}_{\mathbf{P}\left(\tilde{\mathbf{b}}_k^*\right)}+2\underbrace{\sum_{m=1}^{M}\hat{w}_m\frac{\widehat{\mathbf{R}}_m\tilde{\mathbf{b}}_k^*\tilde{\mathbf{b}}_k^{*H}\widehat{\mathbf{R}}_m}{\left(\tilde{\mathbf{b}}_k^{*H}\widehat{\mathbf{R}}_m\tilde{\mathbf{b}}_k^*\right)^2}}_{\mathbf{P}\left(\tilde{\mathbf{b}}_k^*\right)}. \tag{F.6}$$

The matrix $\mathbf{Y}\left(\tilde{\mathbf{b}}_k^*\right)$ from (F.6) can be written in the following manner

$$\mathbf{Y}\left(\tilde{\mathbf{b}}_k^*\right)=\sum_{m=1}^{M}\hat{w}_m\frac{\widehat{\mathbf{R}}_m}{\left(\tilde{\mathbf{b}}_k^{*H}\widehat{\mathbf{R}}_m\tilde{\mathbf{b}}_k^*\right)^2}\underbrace{\sum_{n=1}^{M}\hat{w}_n\frac{\widehat{\mathbf{R}}_n}{\tilde{\mathbf{b}}_k^{*H}\widehat{\mathbf{R}}_n\tilde{\mathbf{b}}_k^*}}_{\mathbf{G}\left(\tilde{\mathbf{b}}_k^*\right)}\tilde{\mathbf{b}}_k^*\tilde{\mathbf{b}}_k^{*H}\widehat{\mathbf{R}}_m. \tag{F.7}$$

According to (3.19) and (3.22) $\mathbf{G}\left(\mathbf{b}_k^*\right)\mathbf{b}_k^*=\mathbf{b}_k^*$. Therefore,

$$\mathbf{Y}\left(\tilde{\mathbf{b}}_k^*\right)=\sum_{m=1}^{M}\hat{w}_m\frac{\widehat{\mathbf{R}}_m\tilde{\mathbf{b}}_k^*\tilde{\mathbf{b}}_k^{*H}\widehat{\mathbf{R}}_m}{\left(\tilde{\mathbf{b}}_k^{*H}\widehat{\mathbf{R}}_m\tilde{\mathbf{b}}_k^*\right)^2}=\mathbf{P}\left(\tilde{\mathbf{b}}_k^*\right). \tag{F.8}$$

Substitution of (F.8) into (F.6) yields

$$\mathbf{H}_{1,2}=\left.\frac{\partial^2\psi}{\partial\mathbf{y}\partial\mathbf{x}^H}\right|_{\substack{\mathbf{x}=\tilde{\mathbf{b}}_k^*\\ \mathbf{y}=\tilde{\mathbf{b}}_k^*\\ \alpha=\alpha^*=0}}=-\left(\mathbf{G}\left(\tilde{\mathbf{b}}_k^*\right)-\mathbf{I}\right)\mathbf{P}\left(\tilde{\mathbf{b}}_k^*\right). \tag{F.9}$$

According to (3.27), it is easy to verify that

$$\mathbf{H}_{1,3}=\left.\frac{\partial^2\psi}{\partial\alpha\partial\mathbf{x}^H}\right|_{\substack{\mathbf{x}=\tilde{\mathbf{b}}_k^*\\ \mathbf{y}=\tilde{\mathbf{b}}_k^*\\ \alpha=\alpha^*=0}}=-\tilde{\mathbf{b}}_k^*, \tag{F.10}$$

$$\mathbf{H}_{2,3} = \left.\frac{\partial^2 \psi}{\partial \alpha \partial \mathbf{y}^H}\right|_{\substack{\mathbf{x}=\tilde{\mathbf{b}}_k^* \\ \mathbf{y}=\tilde{\mathbf{b}}_k^* \\ \alpha=\alpha^*=0}} = \mathbf{0},$$

and

$$\mathbf{H}_{3,3} = \left.\frac{\partial^2 \psi}{\partial \alpha^2}\right|_{\substack{\mathbf{x}=\tilde{\mathbf{b}}_k^* \\ \mathbf{y}=\tilde{\mathbf{b}}_k^* \\ \alpha=\alpha^*=0}} = 0.$$

Appendix G

In this appendix, an algorithm which coordinates the order and phases of the vectors in each eigenvectors matrix of $\left\{\widehat{\mathbf{R}}_m = \hat{\mathbf{W}}\hat{\mathbf{R}}_m\hat{\mathbf{W}}^H\right\}_{m=1}^{M}$ is described. Let $\left\{\hat{\mathbf{V}}_m\right\}_{m=1}^{M}$ and $\left\{\hat{\mathbf{\Lambda}}_m\right\}_{m=1}^{M}$ denote the eigenvectors and eigenvalues matrices of $\left\{\widehat{\mathbf{R}}_m\right\}_{m=1}^{M}$, respectively, where $\hat{\mathbf{V}}_m = \left[\hat{\mathbf{v}}_{1,m},...,\hat{\mathbf{v}}_{K,m}\right]$ and $\hat{\mathbf{\Lambda}}_m = diag\left[\hat{\lambda}_{1,m},...,\hat{\lambda}_{K,m}\right]$.

The first step is to determine a pivoting eigenvectors matrix, $\hat{\mathbf{V}}_p$, according to which $\left\{\hat{\mathbf{V}}_m\right\}_{m\neq p}^{M}$ are coordinated. In this work, the eigenvectors matrix corresponding to the minimum approximated averaged mean square error (MSE) is selected as the pivoting matrix. The term approximated averaged MSE (AAMSE) is defined in the following manner. The MSE of $\hat{\mathbf{v}}_{k,m}$ is

$$MSE_{\hat{\mathbf{v}}_{k,m}} = E_{\hat{\mathbf{v}}_{k,m}}\left[\left(\mathbf{v}_{k,m}-\hat{\mathbf{v}}_{k,m}\right)^H\left(\mathbf{v}_{k,m}-\hat{\mathbf{v}}_{k,m}\right)\right] = 2-2\,\mathrm{Re}\left\{\mathbf{v}_{k,m}^H E_{\hat{\mathbf{v}}_{k,m}}\left[\hat{\mathbf{v}}_{k,m}\right]\right\} \tag{G.1}$$

According to [34] the mean vector of $\hat{\mathbf{v}}_{k,m}$ is asymptotically

$$E_{\hat{\mathbf{v}}_{k,m}}[\hat{\mathbf{v}}_{k,m}] \cong \mathbf{v}_{k,m} - \frac{1}{2N_m}\sum_{\substack{j=1\\ j\neq k}}^{K}\frac{\lambda_{k,m}\lambda_{j,m}}{\left(\lambda_{k,m}-\lambda_{j,m}\right)^2}\mathbf{v}_{k,m}, \tag{G.2}$$

where $\mathbf{v}_{k,m}$ and $\lambda_{k,m}$ are the k^{th} unknown eigenvector and eigenvalue of $\mathbf{W}\mathbf{R}_m\mathbf{W}^H$ and N_m denotes the number of observations used for the estimation of $\widehat{\mathbf{R}}_m$. Therefore, according to (G.1) and (G.2), the MSE of $\mathbf{v}_{k,m}$ can be expressed as

$$MSE_{\hat{\mathbf{v}}_{k,m}} = \frac{1}{N_m}\sum_{\substack{j=1\\ j\neq k}}^{K}\frac{\lambda_{k,m}\lambda_{j,m}}{\left(\lambda_{k,m}-\lambda_{j,m}\right)^2}. \tag{G.3}$$

Hence, the averaged MSE of $\hat{\mathbf{V}}_m$ is given by

$$AMSE_{\hat{\mathbf{V}}_m} = \frac{1}{K}\sum_{k=1}^{K}MSE_{\hat{\mathbf{v}}_{k,m}} = \frac{1}{NK}\frac{1}{\hat{w}_m}\sum_{k=1}^{K}\sum_{\substack{j=1\\ j\neq k}}^{K}\frac{\lambda_{k,m}\lambda_{j,m}}{\left(\lambda_{k,m}-\lambda_{j,m}\right)^2}, \tag{G.4}$$

where $N=\sum_{m=1}^{M}N_m$ and $\hat{w}_m=\frac{N_m}{N}$. The eigenvalues $\left\{\lambda_{k,m}\right\}_{k=1,m=1}^{K,M}$ are unknown and therefore, their estimates, $\left\{\hat{\lambda}_{k,m}\right\}_{k=1,m=1}^{K,M}$, are used instead. Thus, the approximate AMSE (AAMSE) of $\hat{\mathbf{V}}_m$ is

$$AAMSE_{\hat{\mathbf{V}}_m}=\frac{1}{NK}\frac{1}{\hat{w}_m}\sum_{k=1}^{K}\sum_{\substack{j=1\\ j\neq k}}^{K}\frac{\hat{\lambda}_{k,m}\hat{\lambda}_{j,m}}{\left(\hat{\lambda}_{k,m}-\hat{\lambda}_{j,m}\right)^2} \tag{G.5}$$

and the index of the pivoting matrix is selected according to

$$p=\arg\min_m AAMSE_{\hat{\mathbf{V}}_m}=\arg\min_m\left[\frac{1}{\hat{w}_m}\sum_{k=1}^{K}\sum_{\substack{j=1\\ j\neq k}}^{K}\frac{\hat{\lambda}_{k,m}\hat{\lambda}_{j,m}}{\left(\hat{\lambda}_{k,m}-\hat{\lambda}_{j,m}\right)^2}\right]. \tag{G.6}$$

Selection of the pivoting matrix, $\hat{\mathbf{V}}_p$, is followed by order and signs coordination of the vectors in each $\hat{\mathbf{V}}_m$ $\left(m\neq p\right)$ according to $\hat{\mathbf{V}}_p$. This process is performed in the following manner. Let

$$P_{k,j}^{(m)}=\left\langle\hat{\mathbf{v}}_{k,p},\hat{\mathbf{v}}_{j,m}\right\rangle=\hat{\mathbf{v}}_{k,p}^{H}\cdot\hat{\mathbf{v}}_{j,m}\quad\left(j,k=1,...,K;\ m=1,...,M\right), \tag{G.7}$$

where $\left\langle\cdot,\cdot\right\rangle$ denotes the scalar product operator. The index, which corresponds to extreme scalar product between $\hat{\mathbf{v}}_{k,p}$ and $\hat{\mathbf{v}}_{j,m}$, is

$$l_j^{(m)}=\arg\max_k\left|P_{k,j}^{(m)}\right|. \tag{G.8}$$

We note that in order to avoid singularities, the index $l_j^{(m)}$ can not be selected twice for the same $\hat{\mathbf{V}}_m$. The phase of $P_{l_j^{(m)},j}^{(m)}$ is

$$\phi_j^{(m)}=\angle P_{l_j^{(m)},j}^{(m)}, \tag{G.9}$$

where $\angle$ denotes the phase operator. After calculating $l_j^{(m)}$ and $\phi_j^{(m)}$ $\forall\, j=1,...,K$ the coordinated version of $\hat{\mathbf{V}}_m$ is

$$\tilde{\hat{\mathbf{V}}}_m=permute\left(\left[\hat{\mathbf{v}}_{1,m}\cdot\exp\left(-i\phi_1^{(m)}\right),...,\hat{\mathbf{v}}_{K,m}\cdot\exp\left(-i\phi_K^{(m)}\right)\right],\left[l_1^{(m)},...,l_K^{(m)}\right]\right), \tag{G.10}$$

where *permute* denotes the permutation operator.

In summary, the synchronization algorithm comprises the following steps:

1. **Calculate** $AAMSE_{\hat{\mathbf{V}}_m} \ \forall\, m = 1,...,M$ **, according to (G.5).**
2. **Select the pivoting matrix,** $\hat{\mathbf{V}}_p$**, according to (G.6).**
3. **Calculate** $l_j^{(m)}$ **and** $\phi_j^{(m)} \ \forall\, m = 1,...,M$ **and** $\forall\, j = 1,...,K$ **according to (G.8) and (G.9), respectively.**
4. **Coordinate the vectors of** $\hat{\mathbf{V}}_m \ \forall\, m \neq p$ **according to (G.10).**

Appendix H

Let $\left\{\widehat{\mathbf{R}}_m\right\}_{m=1}^{M}=\left\{\hat{\mathbf{W}}\hat{\mathbf{R}}_m\hat{\mathbf{W}}^H\right\}_{m=1}^{M}$, where $\hat{\mathbf{W}}$ is a whitening matrix of $\hat{\mathbf{R}}=\sum_{m=1}^{M}\hat{w}_m\hat{\mathbf{R}}_m$. In this appendix it is shown that the diagonalization matrix of $\left\{\widehat{\mathbf{R}}_m\right\}_{m=1}^{M}$ is approximately unitary and that the eigenvectors of $\left\{\widehat{\mathbf{R}}_m\right\}_{m=1}^{M}$ can be partitioned into *K* distinct clusters.

First, a transformation matrix **W**, for which each matrix in the set $\left\{\mathbf{W}\mathbf{R}_m\mathbf{W}^H\right\}_{m=1}^{M}$ has the same eigenvectors, is derived.

Claim H.1:

Let

$$\mathbf{R}_m=\mathbf{A}\boldsymbol{\Lambda}_m\mathbf{A}^H\ \ \forall\ m=1,...,M\ , \tag{H.1}$$

where $\mathbf{A}=\mathbf{B}^{-1}$ *is not necessarily unitary. If*

$$\mathbf{C}=\sum_{m=1}^{M}w_m\boldsymbol{\Lambda}_m=\mathbf{I}_K\ , \tag{H.2}$$

where $\left\{w_m\right\}_{m=1}^{M}$ *are the weights of* $\left\{\boldsymbol{\Lambda}_m\right\}_{m=1}^{M}$ *and* $\mathbf{I}_K$ *denotes a* $K\times K$ *identity matrix, then there exists a transformation matrix* **W***, such that the matrices in the set* $\left\{\mathbf{W}\mathbf{R}_m\mathbf{W}^H\right\}_{m=1}^{M}$ *have the same eigenvectors.*

In the context of BSS, the physical meaning of the assumption in (H.2) is that the covariance matrix of zero-mean statistically independent sources is the identity matrix.

Proof H.1:

The SVD of **A** *is given by*

$$\mathbf{A}=\underbrace{\mathbf{U}}_{\text{orthonormal}}\cdot\underbrace{\mathbf{D}}_{\text{diagonal}}\cdot\underbrace{\mathbf{V}^H}_{\text{orthonormal}}\ . \tag{H.3}$$

According to (H.1)-(H.3)

$$\mathbf{R} \triangleq \sum_{m=1}^{M} w_m \mathbf{R}_m = \mathbf{A} \underbrace{\left(\sum_{m=1}^{M} w_m \mathbf{\Lambda}_m \right)}_{\mathbf{I}_K} \mathbf{A}^H = \mathbf{A}\mathbf{A}^H = \mathbf{U}\mathbf{D}^2\mathbf{U}^H . \tag{H.4}$$

Let

$$\mathbf{W} = \mathbf{D}^{-1}\mathbf{U}^H , \tag{H.5}$$

such that

$$\mathbf{W}\mathbf{R}\mathbf{W}^H = \mathbf{I}_K . \tag{H.6}$$

According to (H.1) and (H.3) $\forall\ m = 1,...,M$

$$\mathbf{W}\mathbf{R}_m\mathbf{W}^H = \mathbf{W}\mathbf{A}\mathbf{\Lambda}_m\mathbf{A}^H\mathbf{W}^H = \mathbf{W}\mathbf{U}\mathbf{D}\mathbf{V}^H\mathbf{\Lambda}_m\mathbf{V}\mathbf{D}\mathbf{U}^H\mathbf{W}^H . \tag{H.7}$$

Therefore, substituting (H.5) into (H.7) implies that

$$\mathbf{W}\mathbf{R}_m\mathbf{W}^H = \underbrace{\mathbf{D}^{-1}\mathbf{U}^H}_{\mathbf{W}} \underbrace{\mathbf{U}\mathbf{D}\mathbf{V}^H}_{\mathbf{A}} \mathbf{\Lambda}_m \underbrace{\mathbf{V}\mathbf{D}\mathbf{U}^H}_{\mathbf{A}^H} \underbrace{\mathbf{U}\mathbf{D}^{-1}}_{\mathbf{W}^H} = \mathbf{V}^H\mathbf{\Lambda}_m\mathbf{V} \tag{H.8}$$

where $\mathbf{V}$ *is a unitary rotation matrix* □

Let $\mathbf{W}$ and $\mathbf{R}_m$ in (H.8) be substituted by $\hat{\mathbf{W}}$ and $\hat{\mathbf{R}}_m$, respectively, where $\hat{\mathbf{W}}$ denotes the estimate of $\mathbf{W}$, the following can be concluded: 1) The diagonalization matrix of $\left\{\hat{\mathbf{W}}\hat{\mathbf{R}}_m\hat{\mathbf{W}}^H\right\}_{m=1}^{M}$ is approximately unitary; 2) The eigenvectors of $\left\{\hat{\mathbf{W}}\hat{\mathbf{R}}_m\hat{\mathbf{W}}^H\right\}_{m=1}^{M}$ can be partitioned into *K* distinct clusters of eigenvectors.

The estimation of $\mathbf{W}$ is carried out in the following manner. Let

$$\mathrm{SVD}(\hat{\mathbf{R}}) = \hat{\mathbf{U}}\hat{\mathbf{D}}^2\hat{\mathbf{U}}^H , \tag{H.9}$$

where in similar to (H.4)

$$\hat{\mathbf{R}} = \sum_{m=1}^{M} \hat{w}_m \hat{\mathbf{R}}_m . \tag{H.10}$$

Equation (H.5) implies that the estimate of $\mathbf{W}$ is

$$\hat{\mathbf{W}} = \hat{\mathbf{D}}^{-1}\hat{\mathbf{U}}^H . \tag{H.11}$$

References

[1] J. F. Cardoso, "Blind signal separation: statistical principles," *Proceedings of the IEEE*, vol. 86, no. 10, pp. 2009-2025, 1998.

[2] D. T. Pham and J. F. Cardoso, "Blind separation of instantaneous mixtures of nonstationary sources," *IEEE Trans. on Signal Processing*, vol. 49, no. 9, pp. 1837-1848, 2001.

[3] J. F. Cardoso, "High-order contrasts for independent component analysis," *Neural Computation*, vol. 11, pp. 157-192, 1999.

[4] E. Moulines, J. F. Cardoso, and E. Gassiat, "Maximum likelihood for blind separation and de-convolution of noisy signals using mixture models," *in Proceedings of ICASSP*, vol. 5, pp. 3617-3620, 1997.

[5] H. Attias, "Independent factor analysis," *Neural Computation*, vol. 11, pp. 803-851, 1999.

[6] H. Attias, "Independent factor analysis with temporally structured sources," *in Advances in Neural Information Processing Systems*, vol, 12. MIT Press, 2000.

[7] M. Welling and M. Weber, "A constraint EM algorithm for independent component analysis," *Neural Computation*, vol. 13, no. 3, pp. 677-689, 2001.

[8] M. Davis and N. Mitianoudis, "Simple mixture model for sparse overcomplete ICA," *IEE Proc. Vis. Image Signal Process.*, vol. 151, no. 1, February 2004.

[9] T. W. Lee, M. S. Lewicki, and T. J. Sejnowsi, "ICA mixture models for unsupervised classification of non-Gaussian classes and automatic context switching in blind signal separation," *IEEE Trans. on Pattern Analysis and Machine Learning*, vol. 22, no. 10, pp. 1078-1089, 2000.

[10] A. P. Dempster, N. M. Laird, and D. B Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society*, vol. 39 B, pp. 1-38, 1977.

[11] J. Bilmes, "A gentle tutorial on the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden markov models," *Technical Report*, University of Berkely, ICSI-TR-97-021, 1997. Available http://www.citeseer.nj.nec.com/bilmes98gentle.html

[12] J. J. Verbeek, N. Vlassis, and B. Kröse, "Efficient greedy learning for Gaussian mixture models," *Neural Computation*, vol. 15, pp. 469-485, 2003.

[13] B. N. Flury and W. G. Gautschi, "An algorithm for orthogonal transformation of several positive-definite symmetric matrices to nearly diagonal form," *Siam J. Sci. Stat. Comp.*, vol. 7, no. 1, pp. 169-184, 1984.

[14] Q. Li and A. Barron, "Mixture density estimation," *Advances in Neural Information Processing Systems*, vol. 12, pp. 279-285, MIT Press, 2000.

[15] S. I. Amari, "Differential-geometrical methods in statistics," *Lect. Notes in Stat.*, vol. 28, Springer-Verlag, 1985.

[16] A. Hyvarnien and E. Oja, "A fast-fixed-point algorithm for independent component analysis," *Neural Computation*, vol. 9, pp. 1483-1492, 1997.

[17] G. Schwarz, "Estimating the dimension of a model," *Annals of Statistics*, vol. 6, pp. 461-464, 1978.

[18] A. Barron, J. Rissanen, and B. Yu, "The minimum description length principle in coding and modeling," *IEEE Trans. on Information Theory*, vol. 44, no. 6, pp. 2743-2760, 1998.

[19] H. Akaike, "A new look at the statistical model identification," *IEEE Trans. on Automatic Control*, vol. 19, pp. 716-723, 1974.

[20] S. Kullback and R.A. Leibler, "On information and sufficiency," *Ann. Math. Stat.,* vol. 22, pp. 79-86, 1951.

[21] G. Box and G. Tiao, Bayesian inference in statistical analysis. John Wiley and Sons, 1973.

[22] G. H. Golub and C. F. van Loan, Matrix computation. The John Hopkins University Press , 1996, pp. 254.

[23] M. Abramowitz and I. A. Stegun (Eds.), Handbook of mathematical functions with formulas, graphs and mathematical tables. 9th printing, New York: Dover, p. 928, 1972.

[24] J. F. Cardoso and A. Souloumiac, "Jacobi angles for simultaneous diagonalization," *SIAM J. Mat. Anal. Appl.*, vol. 17, 1, pp. 161–164, January 1996.

[25] D. T. Pham, "Joint approximate diagonalization of positive-definite matrices," *SIAM J. Matrix Anal. Appl.*, vol 22, 4, pp. 1136–1152, 2001.

[26] A. J. van der Veen, "Joint diagonalization via subspace fitting techniques," *Proc. ICASSP*, vol. 5, pp. 2773-2776, 2001.

[27] A. Yeredor, "Nonorthogonal joint diagonalization in the least-squares sense with application in blind source separation," *IEEE Trans. on Signal Processing*., vol. 50, 7, pp. 1545–1553, July 2002.

[28] M. Joho and K. Rahbar, "Joint diagonalization of correlation matrices by using Newton methods with application to blind signal separation," *Proc. of IEEE Sensor Array and Multichannel Signal Processing Workshop SAM*, pp. 403–407, 2002.

[29] A. Ziehe, P. Laskov, G. Nolte, and K.-R Müller, "A fast algorithm for joint diagonalization with nonorthogonal transformations and its application to blind source separation," *Journal of Machine Learning Research*, vol. 5, pp. 777-800, July 2004.

[30] K. Todros and J. Tabrikian, "Blind separation of independent sources using Gaussian mixture model," *IEEE Trans. on Signal Processing,* in press.

[31] K. Todros and J. Tabrikian, "Fast approximate joint diagonalization of positive-definite Hermitian matrices," submitted to the *IEEE Trans. on Signal Processing*.

[32] A. J. van der Veen and A. Paulraj, "An analytical constant modulus algorithm," *IEEE Trans. Signal Processing*, vol. 44, pp. 1136–1155, May 1996.

[33] T. Ratnarajah, R. Vaillancourt, and M. Alvo, "Complex random matrices and rayleigh channel," *Communications in Information and Systems*, vol. 3, 2, pp. 119-138, 2003.

[34] M. Kaveh, and A. Barabell, "The statistical performance of the MUSIC and the minimum norm algorithms in resolving plane waves in noise," *IEEE Trans. Acoust. Speech Signal Process*. vol. 34, 2, pp. 331-341, 1986.

[35] S. M. Kay, *Fundamentals of statistical signal processing: estimation theory,* Prentice Hall, pp. 521, 1993.