3dc+ model SIGPRO : 3398 Prod.Type:FLP pp.1–14(col.fig.:NIL) ED:BabuB. PAGN:Bhaskara SCAN:

ARTICLE IN PRESS

Available online at www.sciencedirect.com

ScienceDirect

Signal Processing ▮ (▮▮▮▮) ▮▮▮–▮▮▮

SIGNAL PROCESSING

www.elsevier.com/locate/sigpro

Recursive complex BSS via generalized eigendecomposition and application in image rejection for BPSK

Puskal P. Pokharel[a,*], Umut Ozertem[b], Deniz Erdogmus[b], Jose C. Principe[a]

[a]*Computational NeuroEngineering Laboratory, ECE Department, University of Florida, Gainesville, FL 32611, USA*
[b]*Adaptive Systems Laboratory, CSEE Department, Oregon Health and Science University, OR 97006, USA*

Received 11 December 2006; received in revised form 29 October 2007; accepted 7 December 2007

Abstract

Under the assumptions of non-Gaussian, non-stationary, or non-white independent sources, linear blind source separation can be formulated as generalized eigenvalue decomposition. Here we provide an elegant method of doing this on-line, instead of waiting for a sufficiently large batch of data. This is done through a recursive generalized eigendecomposition algorithm that tracks the optimal solution that one would obtain using all the data observed. The algorithms proposed in this paper follow the well-known recursive least squares (RLS) algorithm in spirit. We also propose to employ this on-line approach for joint image rejection in separating audio signals with the linear mixing varying with time and in slow fading wireless receivers with successful results.

Keywords: Recursive ICA; Recursive BSS; Generalized eigendecomposition; Cumulants; Image rejection

1. Introduction

Independent component analysis (ICA) is an important statistical tool in signal processing and machine learning, both as a solution to the problem of blind source separation (BSS) [1–9] and as a preprocessing step that complements a more comprehensive solution as in dimensionality reduction and feature extraction [9–12]. To implement these applications feasibly on contemporary digital signal processors (DSPs), on-line learning algorithms are required. Currently, the on-line ICA solutions are obtained using algorithms designed with the stochastic gradient concept (e.g., Infomax [13]). The drawbacks of stochastic gradient algorithms in on-line learning include difficulty in selecting the step size for optimal speed misadjustment trade off and suboptimal estimates of the weights given all the samples seen at any given iteration.

Recursive least squares (RLS) is an on-line algorithm for supervised adaptive filter training, which has the desirable property that the estimated weights correspond to the optimal least squares solution that one would obtain using all the data observed so far, provided that initialization is done properly [14]. This benefit comes at a cost of additional computational requirements compared to LMS. Nevertheless, it would be beneficial in certain ICA applications to track at each step the

*Corresponding author. Tel.: +1 352 392 2682; fax: +1 352 392 0044.
E-mail addresses: pokharel@cnel.ufl.edu (P.P. Pokharel), ozertemu@csee.ogi.edu (U. Ozertem), derdogmus@ieee.org (D. Erdogmus), principe@cnel.ufl.edu (J.C. Principe).

0165-1684/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.sigpro.2007.12.001

Please cite this article as: P.P. Pokharel, et al., Recursive complex BSS via generalized eigendecomposition and application in image rejection for BPSK, Signal Process. (2008), doi:10.1016/j.sigpro.2007.12.001

optimal solution given all the data up to the step. The joint diagonalization of higher order statistics have been known to solve the ICA problem under the assumed linear mixing model and have lead to popular algorithms like JADE [1,4]. This motivates the derivation of a recursive generalized eigendecomposition (GED) based ICA algorithm that is similar to RLS in principle, but solved by the simultaneous diagonalization of the second and fourth order joint statistics of the observed mixtures. This can be done in three major ways, assuming the sources are non-stationary and decorrelated [8], non-white and decorrelated [9], or non-Gaussian and independent [15].

GED is an extremely useful statistical tool in many applications like feature extraction, pattern, classification, signal estimation and detection [16,17]. Alternatively, GED is also sometimes referred to as oriented principal component analysis (OPCA) [18,19]. PCA which finds enormous real-world applications is a special case of GED and analytical techniques have been developed in the linear algebra literature [20]. These numerical techniques are computationally inconvenient and moreover require blocks of data. Only fast on-line algorithms can adapt quickly to the changing environment while block techniques lack this feature. The past decade or so has produced plenty of on-line algorithms and efficient computation techniques for PCA and subspace tracking [18,21,22]. These algorithms can be used to solve GED. For instance, a two-step PCA approach has been proposed [17]. But the application of these GED methods using PCA can be limited due to computational requirements. Compared to PCA, there are fewer on-line algorithms to directly solve GED. The traditional power series method for PCA has been revisited recently with improvements [22,23]. We shall employ a similar method for solving GED, such as [24].

The on-line implementation of ICA through GED also requires a sample-by-sample estimation of the correlation and cumulant matrices. Certain efficient methods for on-line and adaptive estimation of cumulants for a white stochastic process have been proposed [25,26]. These methods are very useful for estimating the fourth order cumulant for a scaler process but require careful choice of certain parameters for consistent convergence. Moreover, their extension for a vector time series is non-trivial. Instead we shall start with first principles and derive a very fast and reliable estimation technique following the ideas used in the RLS algorithm for optimal filtering.

An area where these types of on-line algorithms would be useful in wireless receivers. Here we also present an efficient method of image rejection in diversity wireless receivers. Since the algorithm would work in a DSP with digital base band data, it could eliminate highly selective RF band pass filters (BPF) in the front end thus making the front end less costly and more compact. We shall follow the approach presented in [27] to formulate the problem as one of BSS, where the authors present a variant of the Fast-ICA algorithm which is used on batches of data rather than being on-line.

The next section presents the recursive BSS (RBSS) algorithms. Section 3 demonstrates their use in separating linearly mixed speech with a constant mixing matrix and a time varying one, while Sections 4 and 4.1 present the approach for image rejection in wireless receivers and the

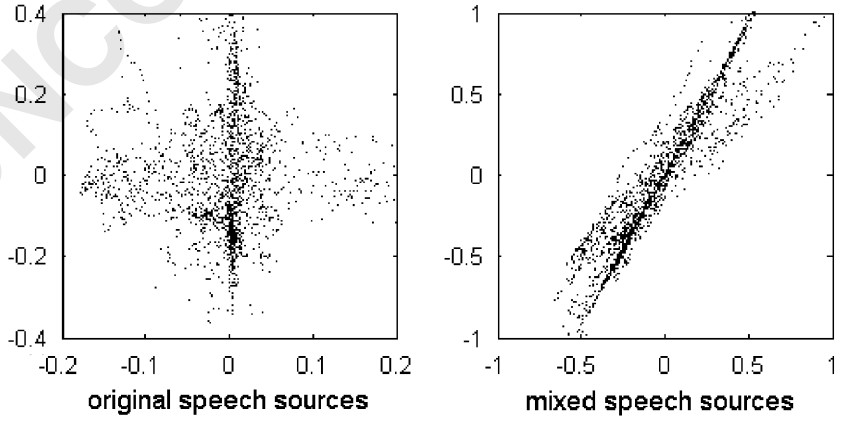

Fig. 1. Scatter plots for the two speech sources before and after linear mixing.

Please cite this article as: P.P. Pokharel, et al., Recursive complex BSS via generalized eigendecomposition and application in image rejection for BPSK, Signal Process. (2008), doi:10.1016/j.sigpro.2007.12.001

corresponding simulation results, respectively (Fig. 1).

2. Recursive BSS algorithms

The square linear ICA problem is expressed as

$$\mathbf{X} = \mathbf{AS}, \tag{1}$$

where $\mathbf{X}$ is the $n \times N$ observations matrix, $\mathbf{A}$ is the $n \times n$ mixing matrix, and $\mathbf{S}$ is the $n \times N$ independent sources matrix. If we consider each column as a sample in time, (1) becomes

$$\mathbf{x}_t = \mathbf{A}\mathbf{s}_t. \tag{2}$$

The joint diagonalization of the covariance matrices and higher order cumulant matrices can be compactly formulated in the form of a GED problem that gives the ICA solution an analytical form [28]. This formulation follows the fact that the matrix $\mathbf{A}$ appears conveniently in the expansion of various cross-statistics of the observations $\mathbf{x}_t$ in terms of the corresponding statistics of the source. More specifically for the covariance matrix $\mathbf{R}_\mathbf{X} = E[\mathbf{x}_t\mathbf{x}_t^\mathrm{H}]$:

$$\mathbf{R}_\mathbf{X} = \mathbf{A}\mathbf{R}_\mathbf{s}\mathbf{A}^\mathrm{H}, \tag{3}$$

where $\mathbf{R}_\mathbf{s}$ is the diagonal covariance matrix for the independent or decorrelated sources. Here we consider the general case of complex variables and the superscript H is the Hermitian transpose operator. Other cross-statistics $\mathbf{Q}_\mathbf{s}$ have the same diagonalization property when the sources are non-Gaussian, non-stationary, or non-white such that

$$\mathbf{Q}_\mathbf{X} = \mathbf{A}\mathbf{Q}_\mathbf{s}\mathbf{A}^\mathrm{H}. \tag{4}$$

For source separation, a demixing matrix $\mathbf{W}^\mathrm{H}$ is to be found such that $\mathbf{W}^\mathrm{H}\mathbf{A} = \mathbf{I}$. This gives

$$\mathbf{s}(t) = \mathbf{W}^\mathrm{H}\mathbf{A}\mathbf{s}(t) = \mathbf{W}^\mathrm{H}\mathbf{x}(t). \tag{5}$$

Let us assume the diagonal matrix $\mathbf{Q}_\mathbf{s}$ is non-singular. Right multiplying (3) and (4) with $\mathbf{W}$ and (4) again with $\mathbf{Q}_\mathbf{s}^{-1}$, and combining the results gives

$$\mathbf{R}_\mathbf{X}\mathbf{W} = \mathbf{Q}_\mathbf{X}\mathbf{W}\Lambda, \tag{6}$$

where $\Lambda = \mathbf{Q}_\mathbf{s}^{-1}\mathbf{R}_\mathbf{s}$ is diagonal since $\mathbf{Q}_\mathbf{s}$ and $\mathbf{R}_\mathbf{s}$ are both diagonal from the assumptions. Eq. (6) is a generalized eigenvalue equation, where the demixing matrix $\mathbf{W}^\mathrm{H}$ is fully determined for distinct eigenvalues specifying n column vectors to corresponding to at most n sources. The recovered sources are arbitrary up to scale and permutations, since the order and scale of the eigenvectors are arbitrary (generally the eigenvectors are scaled to unit norm).

To properly separate the sources the choice of the two cross-statistics matrices in (6) is important and are decided according to the assumptions that are most appropriate for the sources. $\mathbf{R}_\mathbf{X}$ is usually the covariance matrix and $\mathbf{Q}_\mathbf{X}$ is a cumulant matrix for non-Gaussian and independent sources, the covariance matrix at a different time instant for non-stationary and decorrelated sources, or the cross-correlation matrix with a certain time delay for non-white and decorrelated sources. Here the recursive algorithms for all these cases are presented.

2.1. Non-Gaussian and independent sources

For non-Gaussian independent sources that are stationary and white more than second order statistics is required for proper separation. For this case, $\mathbf{Q}_\mathbf{X}$ is chosen as the cumulant matrix estimated using sample averages. While any order of cumulants can be employed, lower orders are more robust to outliers and small sample sizes, so we focus on the fourth order cumulant matrix

$$\mathbf{Q}_\mathbf{X} = E[(\mathbf{x}^\mathrm{H}\mathbf{x})(\mathbf{x}\mathbf{x}^\mathrm{H})] - \mathbf{R}_\mathbf{X}\,\mathrm{trace}(\mathbf{R}_\mathbf{X}) - E[\mathbf{x}\mathbf{x}^\mathrm{T}]E[\mathbf{x}^*\mathbf{x}^\mathrm{H}] - \mathbf{R}_\mathbf{X}\mathbf{R}_\mathbf{X}, \tag{7}$$

where $\mathbf{x}^\mathrm{H}$, $\mathbf{x}^\mathrm{T}$ and $\mathbf{x}^*$, respectively, represent Hermitian transpose, transpose, and complex conjugate of $\mathbf{x}$. Since the fourth order cumulant vanishes for the Gaussian distribution this strategy is only valid for non-Gaussian sources. With iid samples, expectations reduce to sample averages for covariance and cumulant matrices. Then one can define recursive update rules for the estimates of the covariance and cumulant matrices, $\mathbf{R}$ and $\mathbf{Q}$. Also throughout this paper we shall assume that all the signals are zero mean (which otherwise can be trivially obtained by subtracting the mean from the signal). Since the recursive updates are performed with each new incoming data sample, we can introduce a forgetting factor λ $(0<\lambda\leqslant 1)$ in the update equations (similar to that used in RLS). This effectively introduces a decaying window on the data with the ith sample in the past being weighted by $\lambda^{(i-1)/2}$. This can be clearly understood by an inspection of the update equations that follow.

The recursive update rule for the covariance matrix at instant t can be written as

$$\mathbf{R}_t = \frac{t-1}{t}\lambda\mathbf{R}_{t-1} + \frac{1}{t}\mathbf{x}_t\mathbf{x}_t^\mathrm{H} \tag{8}$$

and the update rule for the cumulant matrix is given by

Please cite this article as: P.P. Pokharel, et al., Recursive complex BSS via generalized eigendecomposition and application in image rejection for BPSK, Signal Process. (2008), doi:10.1016/j.sigpro.2007.12.001

$$\mathbf{Q}_t = \mathbf{C}_t - \mathbf{B}_t\mathbf{B}_t^* - \mathbf{R}_t\,\mathrm{trace}(\mathbf{R}_t) - \mathbf{R}_t^2. \tag{9}$$

$\mathbf{C}$ is $E[(\mathbf{x}^H\mathbf{x})(\mathbf{x}\mathbf{x}^H)]$ and its estimate can be updated as

$$\mathbf{C}_t = \frac{t-1}{t}\lambda^2\mathbf{C}_{t-1} + \frac{1}{t}(\mathbf{x}_t^H\mathbf{x}_t)\mathbf{x}_t\mathbf{x}_t^H. \tag{10}$$

$\mathbf{B}$ is $E[\mathbf{x}\mathbf{x}^T]$ and its estimate can be updated as

$$\mathbf{B}_t = \frac{t-1}{t}\lambda\mathbf{B}_{t-1} + \frac{1}{t}\mathbf{x}_t\mathbf{x}_t^T. \tag{11}$$

Thus $\mathbf{B}_t\mathbf{B}_t^*$ can be updated recursively with

$$\mathbf{B}_t\mathbf{B}_t^* = \left(\frac{(t-1)\lambda}{t}\right)^2\mathbf{B}_{t-1}\mathbf{B}_{t-1}^* + \frac{1}{t^2}(\mathbf{x}^T\mathbf{x}^*)\mathbf{x}_t\mathbf{x}_t^H + \frac{(t-1)\lambda}{t^2}(\mathbf{x}_t\mathbf{z}_t^H + \mathbf{z}_t\mathbf{x}_t^H), \tag{12}$$

where $\mathbf{z} = \mathbf{B}_{t-1}\mathbf{x}^*$. The following recursive update of $\mathbf{R}^2$ can be obtained by squaring (8):

$$\mathbf{R}_t^2 = \frac{(t-1)^2}{t^2}\lambda^2\mathbf{R}_{t-1}^2 + \frac{1}{t^2}(\mathbf{x}_t^H\mathbf{x}_t)\mathbf{x}_t\mathbf{x}_t^H + \frac{t-1}{t^2}\lambda[\mathbf{v}_t\mathbf{x}_t^H + \mathbf{x}_t\mathbf{v}_t^H]. \tag{13}$$

For further computational savings we introduce the vector $\mathbf{v}_t$ as $\mathbf{v}_t = \mathbf{R}_{t-1}\mathbf{x}_t$ and we can obtain $\mathbf{R}^{-1}$ and $\mathbf{R}^{-1}\mathbf{Q}$ by iterating to avoid matrix multiplications and inversions having $O(n^3)$ computational load. These two matrices are required for the fixed point algorithm that solves for the GED which is discussed later. Employing the matrix inversion lemma [14], the recursion rule for $\mathbf{R}^{-1}$ becomes

$$\mathbf{R}_t^{-1} = \frac{t}{\lambda(t-1)}\mathbf{R}_{t-1}^{-1} - \frac{t}{\lambda(t-1)\alpha_t}\mathbf{u}_t\mathbf{u}_t^H, \tag{14}$$

where α_t and $\mathbf{u}_t$ are defined as

$$\alpha_t = \lambda(t-1) + \mathbf{x}_t^H\mathbf{u}_t, \quad \mathbf{u}_t = \mathbf{R}_{t-1}^{-1}\mathbf{x}_t. \tag{15}$$

Here we also define the matrix $\mathbf{D}$ as

$$\mathbf{D}_t = \mathbf{R}_t^{-1}\mathbf{Q}_t. \tag{16}$$

To obtain a recursion rule, (16) can be written using (14) and (9) as

$$\mathbf{D}_t = \mathbf{R}_t^{-1}\mathbf{C}_t - \mathbf{R}_t^{-1}\mathbf{B}_t\mathbf{B}_t^* - \mathrm{trace}(\mathbf{R}_t)\mathbf{I} - \mathbf{R}_t. \tag{17}$$

Recursive rule for $\mathbf{R}_t^{-1}\mathbf{C}_t$ can be obtained as

$$\mathbf{R}_t^{-1}\mathbf{C}_t = \lambda(\mathbf{R}_{t-1}^{-1}\mathbf{C}_{t-1} - \mathbf{u}_t(\mathbf{u}_t^H\mathbf{C}_{t-1})) + \frac{\mathbf{x}_t^H\mathbf{x}_t}{\lambda(t-1)}\left(1 + \frac{\mathbf{u}_t^H\mathbf{x}_t}{\alpha_t}\right)\mathbf{u}_t\mathbf{x}_t^H. \tag{18}$$

Now the recursive rule for $\mathbf{R}_t^{-1}\mathbf{B}_t\mathbf{B}_t^*$ can be obtained by using (12) and (14) as

$$\mathbf{R}_t^{-1}\mathbf{B}_t\mathbf{B}_t^* = \frac{(t-1)\lambda}{t}\left[\mathbf{R}_{t-1}^{-1}\mathbf{B}_{t-1}\mathbf{B}_{t-1}^* - \frac{1}{\alpha}\mathbf{u}_t(\mathbf{u}_t^H(\mathbf{B}_{t-1}\mathbf{B}_{t-1}^*))\right] + \frac{1}{t}\left(\left(1 - \frac{\mathbf{u}_t^H\mathbf{x}_t}{\alpha_t}\right)\mathbf{u}_t\mathbf{z}_t^H + (\mathbf{R}_{t-1}^{-1}\mathbf{z}_t)\mathbf{x}_t^H\right) + \left(\frac{\mathbf{x}_t^T\mathbf{x}_t^*}{t}\left(\frac{1}{(t-1)\lambda} - \frac{\mathbf{u}_t^H\mathbf{x}_t}{\alpha_t}\right) + \frac{1}{t}\mathbf{u}_t^H\mathbf{z}_t\right)\mathbf{u}_t\mathbf{x}_t^H. \tag{19}$$

Thus the calculation of $\mathbf{D}_t$ has been reduced to vector–matrix multiplications instead of matrix–matrix multiplications, saving computation time. Of course, this will be appreciable when the number of sources is large. Following the algebraic manipulations presented so far, the overall computational complexity of computing the cumulant matrix is reduced to $O(n^2)$, since all the matrix–matrix multiplication operations have been reduced to a set of vector–matrix operations.

2.2. Non-stationary and decorrelated sources

When the source signals are non-stationary (more specifically in power), the source covariance matrix $\mathbf{R}_\mathbf{s}$ becomes a function of time $\mathbf{R}_\mathbf{s}(t)$. Moreover if they are uncorrelated with each other $\mathbf{R}_\mathbf{s}(t)$ becomes diagonal. Thus, $\mathbf{A}$ is a transformation that expands the diagonal covariance of the sources into the observed covariances at all times as given by (3) despite the signals being non-stationary. In this case, $\mathbf{Q}_\mathbf{x}$ is set as the covariance matrix $E[\mathbf{x}_k\mathbf{x}_k^H]$ where k is at a different time than that used to calculate $\mathbf{R}_\mathbf{x}$. This will give the diagonal cross-statistics required for the generalized eigenvalue equation (6). For computation, we can estimate the expectations of the two covariance matrices by sample averages of the data points in non-overlapping windows, both with lengths close to the stationarity time of the signals. We can also introduce a forgetting factor as before to help in cases where the mixing matrix may change slowly. Then the update rule for $\mathbf{Q}_\mathbf{x}$ is

$$\mathbf{Q}_t = \frac{t-1}{t}\lambda\mathbf{Q}_{t-1} + \frac{1}{t}\mathbf{x}_{k+t}\mathbf{x}_{k+t}^H, \tag{20}$$

and that for $\mathbf{R}_\mathbf{x}$ is

$$\mathbf{R}_t = \frac{t-1}{t}\lambda\mathbf{R}_{t-1} + \frac{1}{t}\mathbf{x}_t\mathbf{x}_t^H, \tag{21}$$

where k is chosen to be greater than the stationarity time of the data. Also, it can be shown that the

Please cite this article as: P.P. Pokharel, et al., Recursive complex BSS via generalized eigendecomposition and application in image rejection for BPSK, Signal Process. (2008), doi:10.1016/j.sigpro.2007.12.001

matrix $\mathbf{D}_t = \mathbf{R}_t^{-1}\mathbf{Q}_t$ can be found recursively by

$$\mathbf{D}_t = \mathbf{D}_{t-1} + \frac{1}{\lambda(t-1)}\left[(\mathbf{R}_{t-1}^{-1}\mathbf{x}_{t+k})\mathbf{x}_{t+k}^{\mathrm{H}} - \frac{\mathbf{u}_t^{\mathrm{H}}\mathbf{x}_{t+k}}{\alpha_t}\mathbf{u}_t\mathbf{x}_{t+k}^{\mathrm{H}}\right] - \frac{1}{\alpha_t}\mathbf{u}_t(\mathbf{u}_t^{\mathrm{H}}\mathbf{Q}_{t-1}), \quad (22)$$

where α_t and $\mathbf{u}_t$ are given by (15).

2.3. Non-white and decorrelated sources

When the sources are non-white and decorrelated, one can use second order statistics in the form of cross-correlations for different time lags τ:

$$\mathbf{R}_{\mathbf{x}}(\tau) = E[\mathbf{x}_t\mathbf{x}_{t+\tau}^{\mathrm{H}}] = \mathbf{A}E[\mathbf{s}_t\mathbf{s}_{t+\tau}^{\mathrm{H}}]\mathbf{A}^{\mathrm{H}} = \mathbf{A}\mathbf{R}_{\mathbf{s}}(\tau)\mathbf{A}^{\mathrm{H}}. \quad (23)$$

More details on simultaneous diagonalization of cross-correlations can be found in [9]. Of course, with the assumptions, $\mathbf{R}_s(\tau)$ will be diagonal and (6) can be employed. For simplicity, let us take

$$\mathbf{R}_{\mathbf{x}} = [\mathbf{R}_{\mathbf{x}}(\tau)]_{\tau=0}. \quad (24)$$

Thus the update rule for $\mathbf{R}_{\mathbf{x}}$ in (3) will remain the same as in (20). $\mathbf{Q}_{\mathbf{x}}$ in (3) can be taken as the symmetric cross-correlation matrix with a non-zero time delay, i.e.,

$$\mathbf{Q}_{\mathbf{x}} = E[\mathbf{x}_t\mathbf{x}_{t+\tau}^{\mathrm{H}} + \mathbf{x}_{t+\tau}\mathbf{x}_t^{\mathrm{H}}]/2. \quad (25)$$

This can be estimated on-line with the forgetting factor λ using

$$\mathbf{Q}_t = \frac{t-1}{t}\lambda\mathbf{Q}_{t-1} + \frac{1}{t}(\mathbf{x}_t\mathbf{x}_{t+\tau}^{\mathrm{H}} + \mathbf{x}_{t+\tau}\mathbf{x}_t^{\mathrm{H}})/2. \quad (26)$$

τ is chosen so that autocorrelation terms in $\mathbf{Q}_{\mathbf{x}}$ are non-zero. The matrix $\mathbf{D}_t = \mathbf{R}_t^{-1}\mathbf{Q}_t$ can be found recursively by

$$\mathbf{D}_t = \mathbf{D}_{t-1} - \frac{1}{\alpha_t}\mathbf{u}_t(\mathbf{u}_t^{\mathrm{H}}\mathbf{Q}_{t-1}) + \frac{1}{2\lambda(t-1)}\left[\mathbf{u}_t\mathbf{x}_{t+\tau}^{\mathrm{H}} + (\mathbf{R}_{t-1}^{-1}\mathbf{x}_{t+\tau})\mathbf{x}_t^{\mathrm{H}} - \frac{\mathbf{u}_t^{\mathrm{H}}\mathbf{x}_t}{\alpha_t}\mathbf{u}_t\mathbf{x}_{t+\tau}^{\mathrm{H}} - \frac{\mathbf{u}_t^{\mathrm{H}}\mathbf{x}_{t+\tau}}{\alpha_t}\mathbf{u}_t\mathbf{x}_t^{\mathrm{H}}\right], \quad (27)$$

where $\mathbf{u}_t$ and α_t are the same as given by (15). Again the overall complexity of computing these matrices as presented in these two subsections is $O(n^2)$.

2.4. Deflation procedure

Having the update equations, the aim is to find the optimal solution for the eigendecomposition for the updated correlation and cumulant matrices in each iteration. As given by (3) we need to solve for the weight matrix $\mathbf{W}$. We will employ the deflation procedure to determine each generalized eigenvector sequentially. Every generalized eigenvector $\mathbf{w}$ that is a column of $\mathbf{W}$ is a stationary point of the function

$$J(\mathbf{w}) = \frac{\mathbf{w}^{\mathrm{H}}\mathbf{R}\mathbf{w}}{\mathbf{w}^{\mathrm{H}}\mathbf{Q}\mathbf{w}}. \quad (28)$$

This fact can easily be seen by taking the derivative of the expression on the right of (21) with respect to $\mathbf{w}$ and equating it to zero which will result in

$$\mathbf{R}\mathbf{w} = \frac{\mathbf{w}^{\mathrm{H}}\mathbf{R}\mathbf{w}}{\mathbf{w}^{\mathrm{H}}\mathbf{Q}\mathbf{w}}\mathbf{Q}\mathbf{w}. \quad (29)$$

This is the equation for GED, the eigenvalues being the value of the objective function $J(\mathbf{w})$ given in (28) evaluated at its stationary points. Thus the fixed point algorithm becomes

$$\mathbf{w} \leftarrow \frac{\mathbf{w}^{\mathrm{H}}\mathbf{R}\mathbf{w}}{\mathbf{w}^{\mathrm{H}}\mathbf{Q}\mathbf{w}}\mathbf{R}^{-1}\mathbf{Q}\mathbf{w}. \quad (30)$$

A similar update rule for the standard eigendecomposition or PCA is traditionally referred to as the power iteration method and hence appropriately we shall use the same name for (30). This procedure converges to the principal GED vector (the one corresponding to the largest generalized eigenvalue) of $\mathbf{R}$ and $\mathbf{Q}$, and the deflation procedure is employed to manipulate the matrices such that they have the same generalized eigenvalue and eigenvector pairs except for the ones that have been determined previously. This method has been described in detail in [24]. The larger eigenvalues are replaced by zeros in each deflation step. Thus a power iteration step after the dth deflation step would result in the GED vector corresponding to the dth largest GED eigenvalue. Note that in this subsection the time index is implicit and omitted for notational convenience.

$$\mathbf{Q}_d = \left[\mathbf{I} - \frac{\mathbf{Q}_{d-1}\mathbf{w}_{d-1}\mathbf{w}_{d-1}^{\mathrm{H}}}{\mathbf{w}_{d-1}^{\mathrm{H}}\mathbf{Q}_{d-1}\mathbf{w}_{d-1}}\right]\mathbf{Q}_{d-1}, \quad \mathbf{R}_d = \mathbf{R}_{d-1}. \quad (31)$$

The deflated matrices are initialized to $\mathbf{Q}_1 = \mathbf{Q}$ and $\mathbf{R}_1 = \mathbf{R}$. Obtaining the new matrices, we employ the same fixed point iteration procedure given in (26) to find the corresponding eigenvector.

Given (30), it is clear that iterating $\mathbf{R}^{-1}$ and $\mathbf{D}$ as suggested will result in computational savings. The deflation rules for these matrices can be deduced easily. The deflation of $\mathbf{R}^{-1}$ is

Please cite this article as: P.P. Pokharel, et al., Recursive complex BSS via generalized eigendecomposition and application in image rejection for BPSK, Signal Process. (2008), doi:10.1016/j.sigpro.2007.12.001

$$\mathbf{R}_d^{-1} = \mathbf{R}_{d-1}^{-1}. \tag{32}$$

Similarly, the deflation rule for $\mathbf{D}$ can be obtained by combining (16), (31) and (32) resulting in

$$\mathbf{D}_d = \mathbf{D}_{d-1}\left[\mathbf{I} - \frac{\mathbf{w}_{d-1}\mathbf{w}_{d-1}^{\mathrm{H}}\mathbf{Q}_{d-1}}{\mathbf{w}_{d-1}^{\mathrm{H}}\mathbf{Q}_{d-1}\mathbf{w}_{d-1}}\right]. \tag{33}$$

For each generalized eigenvector, the corresponding power iteration rule becomes

$$\mathbf{w}_d \leftarrow \frac{\mathbf{w}_d^{\mathrm{H}}\mathbf{R}_d\mathbf{w}_d}{\mathbf{w}_d^{\mathrm{H}}\mathbf{Q}_d\mathbf{w}_d}\mathbf{D}_d\mathbf{w}_d. \tag{34}$$

Employing this power iteration for each dimension and solving for the eigenvectors sequentially, one can update the $\mathbf{W}$ matrix and proceed to the next time update step. The computational burden for computing these GED vectors is $O(n^2)$ for each eigenvector extracted. This is generally true for other power iteration type algorithms [21,29]. Of course, the computational burden could possibly be reduced if the matrices involved have some especial properties, like if they are Toeplitz or Hankel. But that is not the case here. Nevertheless this approach results in extremely fast and accurate convergence. A detailed analysis of the convergence properties can be found in [24]. Even with the vector $\mathbf{w}$ initialized randomly (with unit norm), the power iteration converges within 10–15 steps. After successive time steps accurate convergence can be reached in less than five steps, since one can start the power iteration with $\mathbf{W}$ initialized to what it converged to in the previous time step.

The combination of these weight updates, matrix deflation procedures, and recursive covariance/cumulant updates give us the RBSS algorithms for the three sets of assumptions. The algorithms are summarized in Table 1. In theory, these recursive algorithms are expected to track the batch GED solutions that one would obtain at any given time point using all the data available up to that instant. In practice, random initialization and numerical errors (in updates and power iterations) culminate in some deviation. Still this approach results in extremely fast convergence to the original GED BSS in batch mode [28]. To demonstrate this fact, Fig. 2 shows the comparison of the proposed RBSS with its batch mode counterpart. One thousand Monte Carlo (MC) runs were performed with random unit-norm initialized vectors and random mixing matrices with a condition number of 20. The source signals were iid and uniformly distributed with unit variance. To show the quick decay of the effect of initialization, one set of simulations were performed

Table 1
Summary of recursive blind source separation (RBSS) algorithms

Source assumption	Cross-statistics ($\mathbf{Q}_{\mathbf{x}}$)	Recursion rules
Non-Gaussian independent	$E[\mathbf{x}^{\mathrm{H}}\mathbf{x}\mathbf{x}\mathbf{x}^{\mathrm{H}}]$ $-\mathbf{R}_{\mathbf{X}}\,\mathrm{trace}(\mathbf{R}_{\mathbf{X}})$ $-E[\mathbf{x}\mathbf{x}^{\mathrm{T}}]E[\mathbf{x}^{*}\mathbf{x}^{\mathrm{H}}]$ $-\mathbf{R}_{\mathbf{X}}\mathbf{R}_{\mathbf{X}}$	$\mathbf{R}_t = \frac{t-1}{t}\lambda\mathbf{R}_{t-1} + \frac{1}{t}\mathbf{x}_t\mathbf{x}_t^{\mathrm{H}}$; $\mathbf{B}_t = \frac{t-1}{t}\lambda\mathbf{B}_{t-1} + \frac{1}{t}\mathbf{x}_t\mathbf{x}_t^{\mathrm{T}}$ $\mathbf{Q}_t = \mathbf{C}_t - \mathbf{B}_t\mathbf{B}_t^{*}$ $-\mathbf{R}_t\,\mathrm{trace}(\mathbf{R}_t) - \mathbf{R}_t^2$; $\mathbf{R}_t^2 = \frac{(t-1)^2}{t^2}\lambda^2\mathbf{R}_{t-1}^2$ $+\frac{1}{t^2}(\mathbf{x}_t^{\mathrm{H}}\mathbf{x}_t)\mathbf{x}_t\mathbf{x}_t^{\mathrm{H}}$ $\mathbf{C}_t = \frac{t-1}{t}\lambda^2\mathbf{C}_{t-1}$ $+\frac{1}{t}(\mathbf{x}_t^{\mathrm{H}}\mathbf{x}_t)\mathbf{x}_t\mathbf{x}_t^{\mathrm{H}}$; $+\frac{t-1}{t^2}\lambda[\mathbf{v}_t\mathbf{x}_t^{\mathrm{H}} + \mathbf{x}_t\mathbf{v}_t^{\mathrm{H}}]$ $\mathbf{v}_t = \mathbf{R}_{t-1}\mathbf{x}_t$ ($\mathbf{Q}$ is fourth order cumulant matrix)
Non-stationary decorrelated	$E[\mathbf{x}_k\mathbf{x}_k^{\mathrm{H}}]$	$\mathbf{R}_t = \frac{t-1}{t}\lambda\mathbf{R}_{t-1} + \frac{1}{t}\mathbf{x}_t\mathbf{x}_t^{\mathrm{H}}$; $\mathbf{Q}_t = \frac{t-1}{t}\lambda\mathbf{Q}_{t-1} + \frac{1}{t}\mathbf{x}_{k+t}\mathbf{x}_{k+t}^{\mathrm{H}}$ (k chosen greater than stationarity time of sources)
Non-white decorrelated	$E[\mathbf{x}_t\mathbf{x}_{t+\tau}^{\mathrm{H}} + \mathbf{x}_{t+\tau}\mathbf{x}_t^{\mathrm{H}}]$	$\mathbf{R}_t = \frac{t-1}{t}\lambda\mathbf{R}_{t-1} + \frac{1}{t}\mathbf{x}_t\mathbf{x}_t^{\mathrm{H}}$; $\mathbf{Q}_t = \frac{t-1}{t}\lambda\mathbf{Q}_{t-1}$ $+\frac{1}{t}(\mathbf{x}_t\mathbf{x}_{t+\tau}^{\mathrm{H}} + \mathbf{x}_{t+\tau}\mathbf{x}_t^{\mathrm{H}})$ (τ chosen for non-zero autocorrelation in sources)
Matrix inversion $\mathbf{R}_t^{-1} = \frac{t}{\lambda(t-1)}\mathbf{R}_{t-1}^{-1} - \frac{t}{\lambda(t-1)\alpha}\mathbf{u}_t\mathbf{u}_t^{\mathrm{H}}$; $\alpha_t = \lambda(t-1) + \mathbf{x}_t^{\mathrm{H}}\mathbf{u}_t$; $\mathbf{u}_t = \mathbf{R}_{t-1}^{-1}\mathbf{x}_t$		
Deflation and eigendecomposition steps $\mathbf{D} = \mathbf{R}^{-1}\mathbf{Q}$; $\mathbf{D}_d = \mathbf{D}_{d-1}\left[\mathbf{I} - \frac{\mathbf{w}_{d-1}\mathbf{w}_{d-1}^{\mathrm{H}}\mathbf{Q}_{d-1}}{\mathbf{w}_{d-1}^{\mathrm{H}}\mathbf{Q}_{d-1}\mathbf{w}_{d-1}}\right]$; $\mathbf{Q}_d = \left[\mathbf{I} - \frac{\mathbf{Q}_{d-1}\mathbf{w}_{d-1}\mathbf{w}_{d-1}^{\mathrm{H}}}{\mathbf{Q}_{d-1}\mathbf{w}_{d-1}\mathbf{w}_{d-1}^{\mathrm{H}}}\right]\mathbf{Q}_{d-1}$; $\mathbf{R}_d = \mathbf{R}_{d-1}$; $\mathbf{w}_d \leftarrow \frac{\mathbf{w}_d^{\mathrm{H}}\mathbf{R}_d\mathbf{w}_d}{\mathbf{w}_d^{\mathrm{H}}\mathbf{Q}_d\mathbf{w}_d}\mathbf{D}_d\mathbf{w}_d$ (fixed point iteration for each dimension, d)	

Please cite this article as: P.P. Pokharel, et al., Recursive complex BSS via generalized eigendecomposition and application in image rejection for BPSK, Signal Process. (2008), doi:10.1016/j.sigpro.2007.12.001

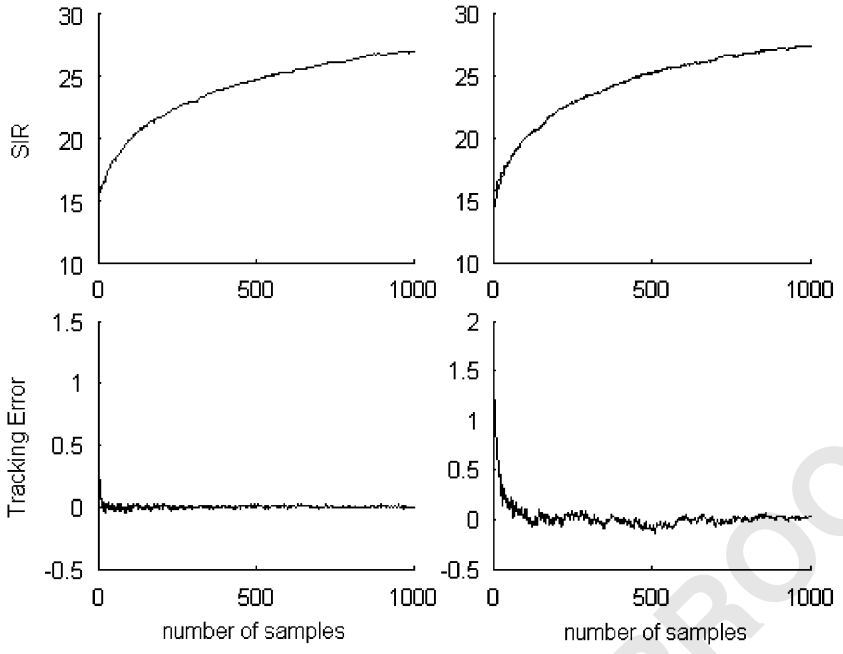

Fig. 2. Performance comparison of the original GED BSS and that of proposed RBSS using an on-line sample by sample update.

with the correlation and fourth order cumulant matrices initialized to $10^{-4}\mathbf{I}$ (top and bottom left plots) and another set with the matrices initialized to $2\mathbf{I}$ (top and bottom right plots). Obviously, it is always suitable to choose a small initialization for these algorithms. In both cases the tracking error between the RBSS approach and that of the batch mode GED BSS is very minimal. Of course, the tracking convergence with a small initialization is much faster than that with a large one. This is demonstrated by the bottom two curves. These properties are the same as is observed for the RLS update rules for optimal filtering.

To summarize the computation burden for each time step, the power iteration is $O(n^2)$ for each GED vector and the matrix updates are overall $O(n^2)$. So, major part of the computation is for the GED solution. So the overall complexity is dominated by $O(n^3 + n^2)$ especially if n is large. But this approach provides fast and reliable convergence which is crucial for any on-line method. Moreover, unlike less complex stochastic algorithms, dependence on the initialization and adaptation parameters is very minimal. In contrast, a batch mode algorithm would have to wait for a sufficiently large data size before it can be run and may require an extra prewhitening stage which again cannot be performed on-line and would require extra computation burden. Moreover these methods cannot be employed in a changing environment. We shall show one simple example where the RBSS algorithm shall work for separating speech signals with the linear mixing varying with time.

3. Experiments for speech separation

We will demonstrate performance of the RBSS algorithms we have proposed to separate linearly mixed speech sources. We will present two basic experiments in this section:

(a) Comparison of the original GED-BSS algorithms [28] with the results of the proposed RBSS algorithms.

Here the forgetting factor was taken as unity. For reference, Fast-ICA [29] results are included in the comparisons. Although we initially attempted to include comparisons with the stochastic gradient based Infomax [13] algorithm, finding a large stable stepsize for each individual run proved to be challenging, therefore these results are omitted. The experiments include the separation of speech signals from instantaneous linear mixtures. The database consists of 10 clips of acoustic signals (five male, four female, one symphony). We selected nine random pairs from this set and run 10 MC simulations for each pair. In each MC run, a mixing matrix with constant condition number is generated, and the RBSS algorithms were randomly

Please cite this article as: P.P. Pokharel, et al., Recursive complex BSS via generalized eigendecomposition and application in image rejection for BPSK, Signal Process. (2008), doi:10.1016/j.sigpro.2007.12.001

initialized to small diagonal correlation matrices and random weight matrices. Original GED-BSS and Fast-ICA algorithms both ran on a batch of data, with the batch size increasing by one sample in each iteration. The RBSS algorithms operate on-line updating matrices and weights using one new sample at a time. All RBSS algorithms and Fast-ICA were allowed five fixed-point updates per new sample using their respective update rules. This means, Fast-ICA implements five fixed point iterations over the whole available data set at any given time. Comparisons are provided using the standard average signal-to-interference ratio (SIR) measure in decibels (dB) [30]. Fig. 3 shows the performance of Fast-ICA, original GED-BSS and the three RBSS methods for mixture condition number of 40. The results were very similar for condition numbers from small values like 5 to moderately large values like 100, as we would expect. In these cases, the convergence speed was not affected by the mixture condition number. The two algorithms using non-Gaussianity and independence assumptions and Fast-ICA performed worse than the RBSS algorithms using the more suitable non-stationarity and non-whiteness assumptions for speech. It is known that Fast-ICA shows degraded performance when the sources are non-white and when the data size is small [31]. Fig. 4 shows the tracking error between the RBSS algorithms and their corresponding GED-BSS algorithms. The asymptotic tracking error could be made arbitrarily small by letting RBSS algorithms iterate more per sample.

(b) Performance of the RBSS algorithms for a time varying mixing matrix.

Here we design an example that simulates a time varying mixing for speech signals. Our scheme is depicted in Fig. 5. Speech source S1 remains fixed at its position, whereas source S2 moves long the circle with radius r and center at O. The audio sensors are fixed at P1 and P2 collinear and at a distance l from the center. S2 is normal at O to the line between P1 and P2. θ is the angular separation between S1 and S2. So, as S2 moves θ increases. Here we will assume that the attenuation of each speech signal as received by an audio sensor is inversely related to the distance between the source and the sensor. So we shall take the time varying mixing matrix to be

$$\mathbf{A}(n) = \begin{bmatrix} a_{11}(n) & a_{12}(n) \\ a_{21}(n) & a_{22}(n) \end{bmatrix}, \tag{35}$$

such that

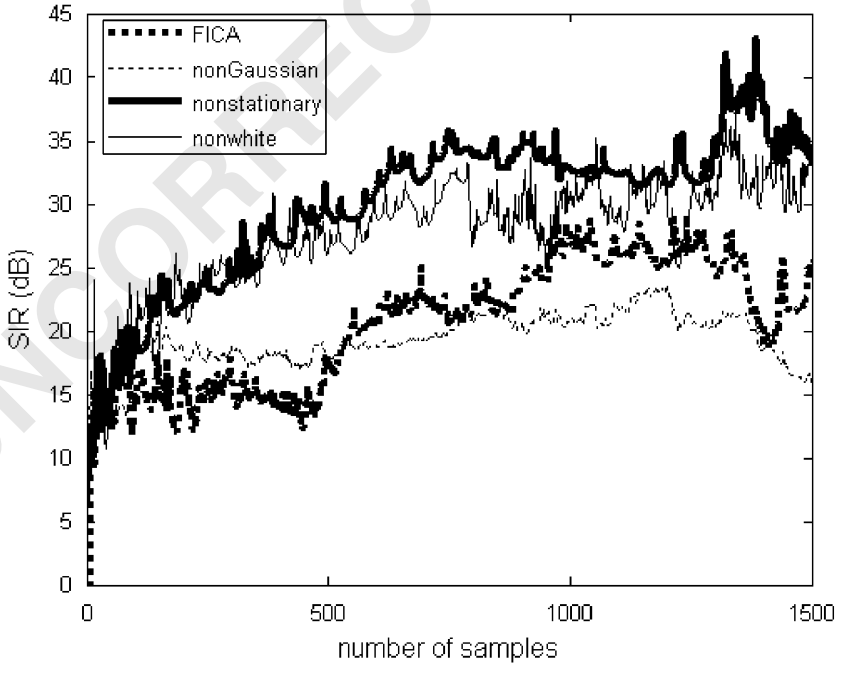

Fig. 3. SIR (dB) for Fast-ICA and recursive BSS using the assumptions of non-Gaussianity, non-stationarity and non-whiteness with the condition number of the mixing matrix as 40.

Please cite this article as: P.P. Pokharel, et al., Recursive complex BSS via generalized eigendecomposition and application in image rejection for BPSK, Signal Process. (2008), doi:10.1016/j.sigpro.2007.12.001

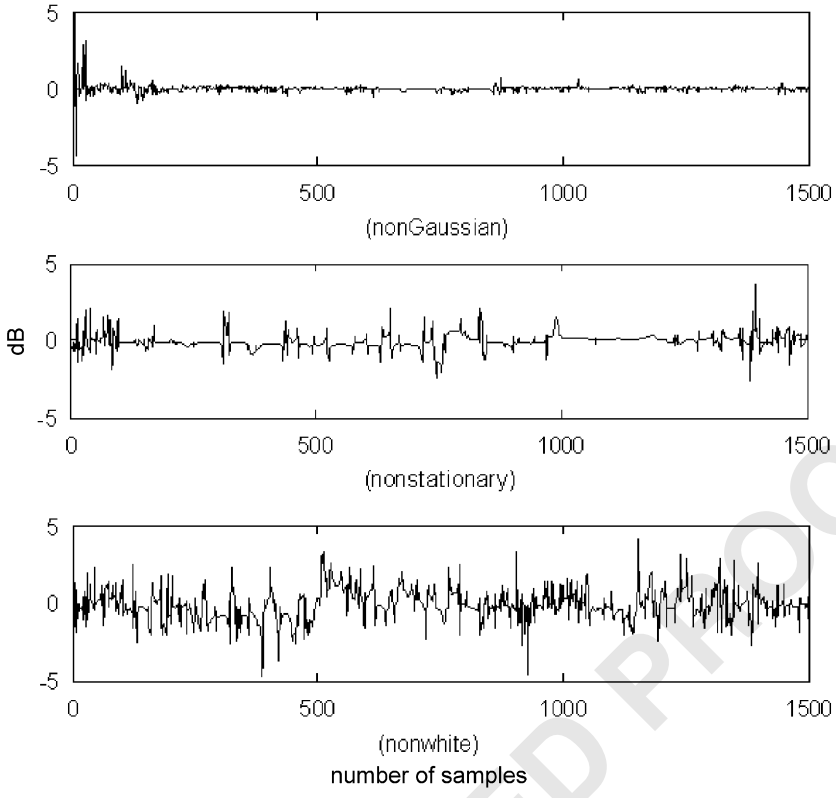

Fig. 4. Performance difference between the original GED-BSS and the proposed RBSS for the three different assumptions of non-Gaussianity, non-stationarity and non-whiteness.

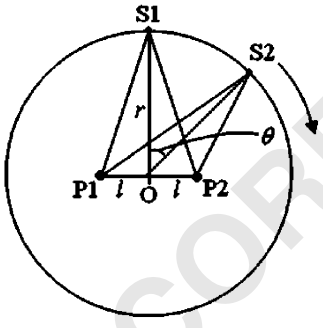

Fig. 5. The arrangement of two speech sources (with one moving) to simulate time varying linear mixing.

$$a_{11}(n) = a_{21}(n) = \frac{1}{\sqrt{l^2 + r^2}}, \tag{36}$$

$$a_{12}(n) = \frac{1}{\sqrt{l^2 + r^2 + 2rl\sin\theta(n)}}, \tag{37}$$

and

$$a_{22}(n) = \frac{1}{\sqrt{l^2 + r^2 - 2rl\sin\theta(n)}}. \tag{38}$$

For our results we used $l = 1$ and $r = 2$ and $\theta(n)$ increased (starting at $\theta(0) = 0$) at a rate of $4\pi/9000$ radians per sample. With a sampling rate of 16 kHz that would translate to approximately 0.0071π radians per second. And accordingly, the mixing matrix $\mathbf{A}(n)$ would change. Fig. 6 shows the corresponding SIR plots. Since for speech, the RBSS algorithms with the assumption of non-stationarity and that with the assumption of non-whiteness work the best, we have only shown the results for these cases. We used a forgetting factor of $\lambda = 0.995$ in both cases. We selected nine random pairs from the audio data set mentioned earlier and the plots show the average performance. Note that in the plots the steep valleys occur when θ is an integer multiple of π (since the mixing matrix would be singular). The assumption that the sources are non-stationary does not work as well as the assumption that they are non-white since the former requires the covariance matrices to be computed at

Please cite this article as: P.P. Pokharel, et al., Recursive complex BSS via generalized eigendecomposition and application in image rejection for BPSK, Signal Process. (2008), doi:10.1016/j.sigpro.2007.12.001

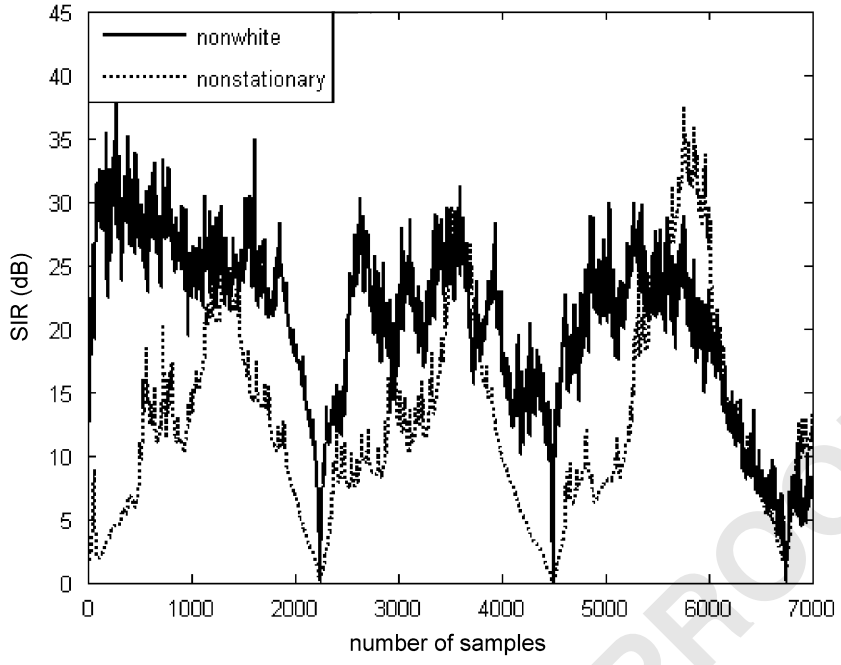

Fig. 6. SIR for a time varying mixing matrix as a speech source moves along a circle around audio sensors.

two time instances separated well enough to exploit the non-stationarity of the data resulting in two different mixing matrices (since the mixing is time variant).

4. Joint image rejection

One major advantage of RBSS algorithms will be for problems where the observations are obtained sequentially with time. These situations are also prevalent in communication receivers. For more integration and control, DSPs are required to perform most of the radio receiver functionalities and more efficient means of performing front end operations are required. One challenging task for wireless receivers is to reject the radio frequency (RF) image that is generated by the wireless process. Highly selective RF BPF can be used but this increases the cost and the difficulty in integration [32]. Low intermediate frequency (IF) systems mitigate the image problem, but the mismatch between the in- and quadrature-phase down conversion paths also requires additional processing and more hardware [33,34]. We shall follow the approach presented in [27] to formulate the problem as one of BSS. We shall frame the source separation problem for BPSK receivers. The image signal (that interferes at the receiver, usually through heterodyning [35]) can be considered as an additive interference. In fact, the same procedure could be followed in a slow fading multichannel communication system to remove unwanted signal components from other channels. As is well known, BPSK or its variants are used in many applications for reasons of robustness and link budget enhancement [36–38]. This is usually very crucial for the control channel of a wireless network. So we believe the following presentation could contribute for such cases.

Fig. 7 shows the simple receiver structure employing the usual steps to convert the received signal to digital base band [35]. Two antennas are used thus resulting in two base band observations. The antennas receive two signals each consisting of the desired signal $s(t)$ and image signal $i(t)$. Each of the signals has its fading coefficients defined as

$$f_{sk} = \alpha_{sk} e^{j\psi_{sk}}, \tag{39}$$

$$f_{ik} = \alpha_{ik} e^{j\psi_{sk}}, \tag{40}$$

where $k(=1,2)$ is the antenna index; f_{sk} and f_{ik} are the fading coefficients for the desired and image signals, respectively. Likewise, α_{sk} and α_{ik} are the channel's amplitude responses, and ψ_{sk} and ψ_{ik} are the channel's phase responses. As we shall see later the type of fading (Raleigh or otherwise) is not important for this approach of image suppression. Using simple mathematical manipulations we can

Please cite this article as: P.P. Pokharel, et al., Recursive complex BSS via generalized eigendecomposition and application in image rejection for BPSK, Signal Process. (2008), doi:10.1016/j.sigpro.2007.12.001

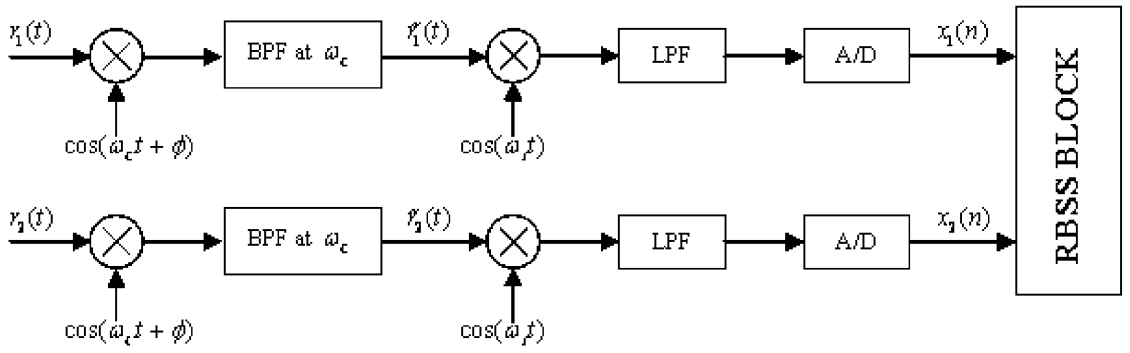

Fig. 7. Block diagram of the BPSK diversity receiver employing RBSS.

show that a proper BSS algorithm can remove the image component [27] without any prior knowledge of the transmitted signal.

The received signal of the kth antenna $r_k(t)$ can be expressed as

$$\begin{aligned} r_k(t) &= 2\,\mathrm{Re}[s(t)f_{sk}e^{j(\omega_C+\omega_I)t} + i(t)f_{ik}e^{j(\omega_C-\omega_I)t}] \\ &= s(t)f_{sk}e^{j(\omega_C+\omega_I)t} + i(t)f_{ik}e^{j(\omega_C-\omega_I)t} \\ &\quad + s^*(t)f_{sk}^*e^{-j(\omega_C+\omega_I)t} + i^*(t)f_{ik}^*e^{-j(\omega_C-\omega_I)t}, \quad (41) \end{aligned}$$

where Re[·] gives the real part of the complex argument ω_C is the oscillator frequency of the first mixer and ω_I is the frequency of the second local oscillator. The receiver employs the basic mixing and filtering scheme to retrieve the base band signals from (41). At the receiver a constant frequency oscillator (CFO) is present to generate the mixer signal with the appropriate frequency. It is given by

$$x_{LO}(t) = 2\cos(\omega_C t+\phi) = e^{j(\omega_C t+\phi)} + e^{-j(\omega_C t+\phi)}, \quad (42)$$

where ϕ is the CFO phase offset. Next the signals are down converted to IF followed by BPF with the center frequency at ω_I for band selection. The output of the BPF is then

$$\begin{aligned} \tilde{r}_k(t) &= s(t)f_{sk}e^{j(\omega_I t-\phi)} + i(t)f_{ik}e^{-j(\omega_I t+\phi)} \\ &\quad + s^*(t)f_{sk}^*e^{-j(\omega_I t-\phi)} + i^*(t)f_{ik}^*e^{j(\omega_I t+\phi)}. \quad (43) \end{aligned}$$

At the final stage the IF signals are further down converted to base band and A/D conversion is performed resulting in the discrete time signal given by

$$x_k(n) = \mathrm{Re}[s(n)f_{sk}e^{-j\phi} + i^*(n)f_{ik}^*e^{j\phi}]. \quad (44)$$

Since $s(n)$ and $i(n)$ are real for BPSK signals, we can write (6) as

$$x_k(n) = a_k s(n) + b_k i(n), \quad (45)$$

where $a_k = \mathrm{Re}[f_{sk}e^{-j\phi}]$ and $b_k = \mathrm{Re}[f_{ik}^*e^{j\phi}]$. Thus using the two antennas in the receivers, Eq. (28) can be in vector form as

$$\mathbf{x}(n) = \begin{bmatrix} a_1 & b_1 \\ a_2 & b_2 \end{bmatrix} \begin{bmatrix} s(n) \\ i(n) \end{bmatrix} = \mathbf{A}\mathbf{s}(n). \quad (46)$$

Eq. (46) is of the same form as (2) and hence BSS can be applied to recover $s(n)$. Of course the mixing matrix $\mathbf{A}$ must be non-singular for proper source separation. This is usually not a problem due to the randomness of the wireless channel. In order to mitigate the inherent ambiguity in the order of the separated sources, we shall assume that reference sequences are present in transmitted signals, which are available in most communication standards. Also notice that the type of fading in the channel does not affect the final form of (46). In next section we shall present the simulation results using the proposed RBSS algorithm using the assumption of non-Gaussianity as explained in Section 2.1.

4.1. Experimental results

Here we present some simulation results for image rejection in a wireless receiver as explained in Section 4. We have performed 100 MC simulations with random fading coefficients and BPSK symbols transmitted. The amplitudes and phase for the fading coefficients are generated as Rayleigh distributed with parameter value of 1 and uniform distributed in $[0, 2\pi)$, respectively. Likewise, the phase offset in (42) is also chosen to be uniformly distributed in $[0, 2\pi)$. The measure of performance is SIR as before. We have employed the RBSS algorithm assuming non-Gaussian distribution and using the fourth order cumulant as described in Section 2.1. As before we have also included the performance of using the Fast-ICA algorithm,

Please cite this article as: P.P. Pokharel, et al., Recursive complex BSS via generalized eigendecomposition and application in image rejection for BPSK, Signal Process. (2008), doi:10.1016/j.sigpro.2007.12.001

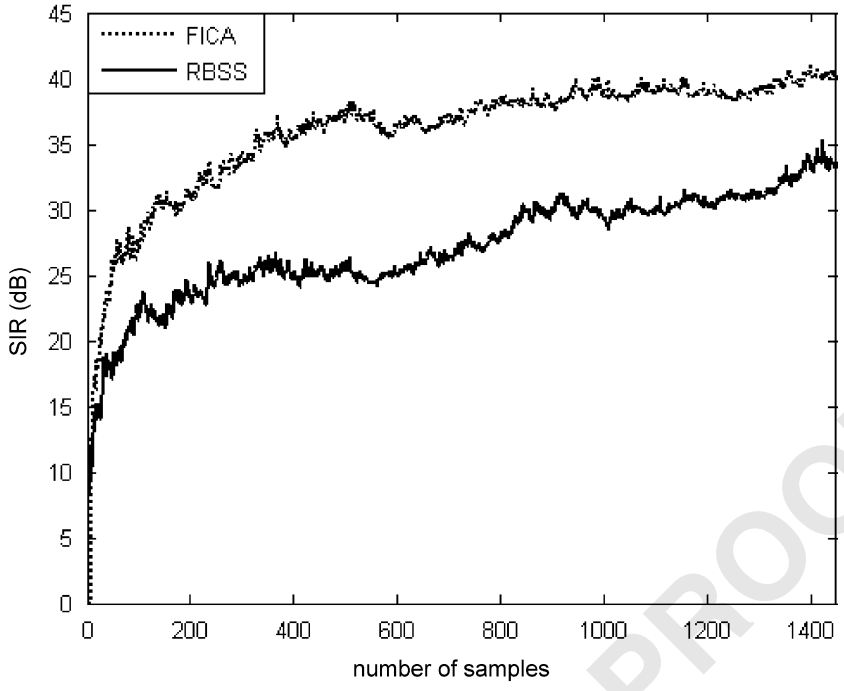

Fig. 8. Performance of Fast-ICA using a batch of data and that of RBSS using an on-line sample by sample update.

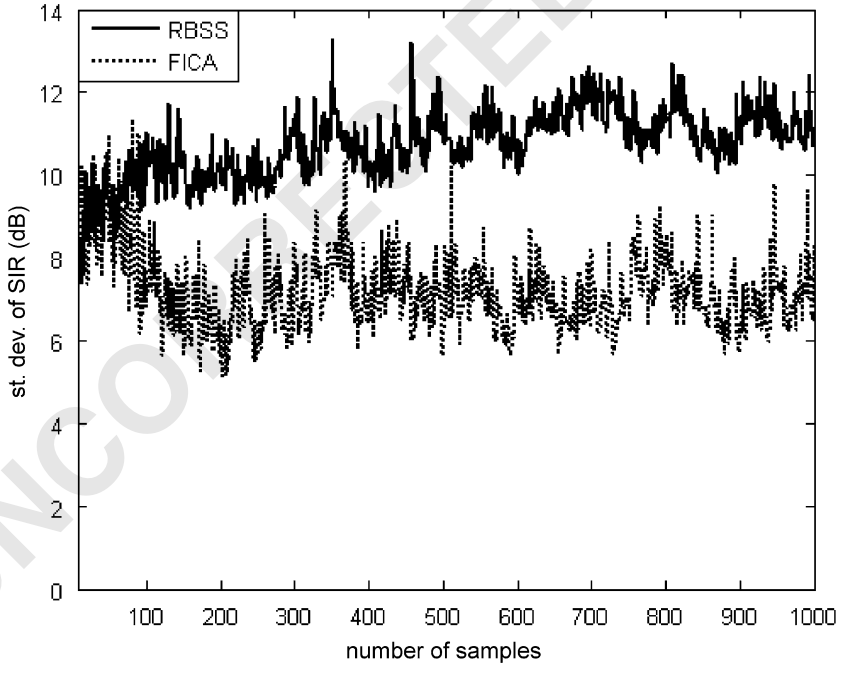

Fig. 9. The standard deviations of the two methods: Fast-ICA using a batch of data and RBSS using an on-line sample by sample update.

which unlike the proposed method works with a batch of data and hence would incur tremendous computational burden if employed to work on-line with the batch size increasing by one with each sample. Also this method would require the system to store all the past observations. The RBSS algorithm and Fast-ICA are allowed five fixed-point

Please cite this article as: P.P. Pokharel, et al., Recursive complex BSS via generalized eigendecomposition and application in image rejection for BPSK, Signal Process. (2008), doi:10.1016/j.sigpro.2007.12.001

updates per new sample using their respective update rules.

Figs. 8 and 9 show the performance of these two approaches. Fast-ICA [29] has a slight advantage over RBSS with an improvement of about 5 dB on an average as the curves settle. Nonetheless, RBSS also achieves image rejection successfully with an added advantage of it being on-line. Again results obtained using Infomax [13] which is an on-line method has been omitted since choosing a proper nonlinearity for this problem (the source has a bimodal distribution with BPSK and a simple sigmoid would not work) was challenging and performance was far inferior.

5. Conclusion

On-line ICA/BSS algorithms are essential for many signal processing and machine learning applications, where the ICA solution acts as a front-end preprocessor, a feature extractor, or a portion of a solution to a larger problem. Though stochastic gradient based algorithms motivated by various ICA criteria is used in such situations with the advantage of yielding computationally simple weight update rules, they do not offer an optimal solution at every iteration and choosing an appropriate step size is still an inconvenience. In this paper we presented recursive BSS algorithms based on the joint diagonalization of various cross-statistics based on three standard assumption sets regarding source signals: non-Gaussianity, non-stationarity, and non-whiteness. The derivation employs the use of the matrix inversion lemma and the update rules for the expectations approximated by sample averages. The resulting algorithm, of course, is computationally more expensive than stochastic gradient type algorithms per update. However, it converges to and tracks the optimal solution based on its separation criterion in a small number of samples/iterations, even with random initialization.

To demonstrate the tracking ability of RBSS we applied this for linearly mixed speech sources with one source moving circularly around the audio sensors. Thus the mixing matrix became a function of time. Results show that these set of algorithms are appropriate for such scenarios as well. We also presented an elegant application of the RBSS algorithm in image rejection for wireless receivers. The method is independent of the channel fading type and can be effectively incorporated in the DSP. This works with the digital base band signal, removing the need of economically and spatially expensive frequency selective band pass analog filters in the receiver front end.

References

[1] P. Comon, Independent component analysis, a new concept?, Signal Processing 36 (1994) 287–314.
[2] C. Jutten, J. Herault, Blind separation of sources, part I: an adaptive algorithm based on neuromimatic architecture, Signal Processing 24 (1) (1991) 1–10.
[3] P. Comon, C. Jutten, J. Herault, Blind separation of sources, part II: problems statement, Signal Processing 24 (1) (1991) 11–20.
[4] J. Cardoso, Blind signal separation: statistical principles, Proc. IEEE 86 (1998) 2009–2025 (special issue).
[5] A. Belouchrani, K. Abed-Meraim, J.-F. Cardoso, A blind source separation technique using second-order statistics, IEEE Trans. Signal Process. 45 (2) (1997) 434–444.
[6] A. Hyvarinen, J. Karhunen, E. Oja, Independent Component Analysis, Wiley, New York, 2001.
[7] A. Cichoki, S.I. Amari, Adaptive Blind Signal and Image Processing: Learning Algorithms and Applications, Wiley, New York, 2002.
[8] L. Molgedey, H.G. Schuster, Separation of a mixture of independent signals using time delayed correlations, Phys. Rev. Lett. 71 (1994) 3634–3637.
[9] E. Weinstein, M. Feder, A. Oppenheim, Multi-channel signal separation by decorrelation, IEEE Trans. Speech Audio Process. 1 (1993) 405–413.
[10] A. Hyvarinen, E. Oja, P. Hoyer, J. Hurri, Image feature extraction by sparse coding and independent component analysis, in: ICPR'98, 1998, pp. 1268–1273.
[11] T. Lan, D. Erdogmus, A. Adami, M. Pavel, Feature selection by independent component analysis and mutual information maximization in EEG signal classification, in: IJCNN 2005, 2005, pp. 3011–3016.
[12] A. Hyvarinen, E. Oja, Independent component analysis: algorithms and applications, Neural Networks 13 (2000) 411–430.
[13] A. Bell, T. Sejnowski, An information-maximization approach to blind separation and blind deconvolution, Neural Comput. 7 (1995) 1129–1159.
[14] S. Haykin, Adaptive Filter Theory, Prentice-Hall, Englewood Cliffs, NJ, 1996.
[15] J. Cardoso, A. Souloumiac, Blind beamforming for non-Gaussian signals, IEE Proc.-F 140 (1993) 362–370.
[16] R.O. Duda, P.E. Hart, Pattern Classification and Scene Analysis, Wiley, New York, 1973.
[17] J. Mao, A.K. Jain, Artificial neural networks for feature extraction and multivariate data projection, IEEE Trans. Neural Networks 6(2).
[18] E.F. Deprettere (Ed.), SVD and Signal Processing. Algorithms, Applications and Architectures, North-Holland, Amsterdam, 1988.
[19] K.I. Diamantaras, S.Y. Kung, An unsupervised neural model for oriented principal component extraction, in: Proceedings of the International Conference on Acoustics, Speech, and Signal Processing, 1991, pp. 1049–1052.

Please cite this article as: P.P. Pokharel, et al., Recursive complex BSS via generalized eigendecomposition and application in image rejection for BPSK, Signal Process. (2008), doi:10.1016/j.sigpro.2007.12.001

[20] G.H. Golub, C.F.V. Loan, Matrix Computations, The John Hopkins University Press, Baltimore, MD, 1991.
[21] P. Comon, G.H. Golub, Tracking a few extreme singular values and vectors in signal processing, Proc. IEEE 78 (8) (1990) 1327–1343.
[22] Y. Hua, Y. Xiang, T. Chen, K. Abed-Meraim, Y. Miao, Natural power method for fast subspace tracking, in: Proceedings of the IEEE Workshop on Neural Networks for Signal Processing IX, 1999, pp. 176–185.
[23] Y.N. Rao, J.C. Principe, A fast on-line algorithm for PCA and its convergence characteristics, in: Proceedings of the IEEE Workshop on Neural Networks for Signal Processing X, 2000, pp. 299–308.
[24] Y.N. Rao, J.C. Principe, T.F. Wong, Fast RLS-like algorithm for generalized eigendecomposition and its applications, J. VLSI Signal Process. 37 (2004) 333–344.
[25] A. Benveniste, M. Metivier, P. Priouret, Adaptive Algorithms and Stochastic Approximations, Springer, New York, 1990.
[26] P.O. Amblard, J.M. Brossier, Adaptive estimation of the fourth-order cumulant of a white stochastic process, IEEE Trans. Neural Networks 42 (1995) 37–43.
[27] T. Yang, W.B. Mikhael, Practical image rejection for single branch BPSK receivers, IEEE Electron. Lett. 40 (2004) 1453–1454.
[28] L. Parra, P. Sajda, Blind source separation via generalized eigenvalue decomposition, J. Machine Learning Res. 4 (2003) 1261–1269.
[29] A. Hyvarinen, E. Oja, A fast fixed-point algorithm for independent component analysis, Neural Comput. 9 (1997) 1483–1492.
[30] K. Hild, D. Erdogmus, J. Principe, Blind source separation using Renyi's mutual information, IEEE Signal Process. Lett. 8 (6) (2001) 174–176.
[31] P. Tichavsky, Z. Koldovsky, E. Oja, Performance analysis for the FastICA algorithm and Cramer–Rao bounds for linear independent component analysis, IEEE Trans. Signal Process. 54 (4) (2006) 1189–1203.
[32] W. Namgoong, T. Meng, Direct-conversion RF receiver design, IEEE Trans. Commun. 49 (3) (2001) 518–529.
[33] J.P.F. Glas, Digital I/Q imbalance compensation in a low-IF receiver, in: IEEE Global Communications Conference, vol. 3, 1998, pp. 1461–1466.
[34] J. Crols, M.S.J. Steyaert, A single-chip 900 MHz CMOS receiver front-end with a high performance low-IF topology, IEEE J. Solid State Circuits 30 (12) (1995) 1483–1492.
[35] J.G. Proakis, Digital is Communications, fourth ed., McGraw-Hill, New York, 2001.
[36] Y. Hu, M. Sawan, A fully-integrated low-power BPSK based wireless inductive link for implantable medical devices, in: Proceedings of the IEEE Midwest Symposium on Circuits and Systems, vol. 3, 2004.
[37] M.E. Fox, M.W. Marcellin, Shaped BPSK and the 5 kHz UHF SATCOM channels, in: Proceedings of the IEEE Military Communications Conference, vol. 1, 1991, pp. 326–332.
[38] F.C. Zheng, S.K. Barton, Orthogonal on-off BPSK: a convenient signaling scheme for near-far resistant detection in DS/CDMA, IEEE Trans. Vehicular Technol. 47 (1998) 969–976.

Please cite this article as: P.P. Pokharel, et al., Recursive complex BSS via generalized eigendecomposition and application in image rejection for BPSK, Signal Process. (2008), doi:10.1016/j.sigpro.2007.12.001