A FAST ALGORITHM FOR BLIND SEPARATION OF NON-GAUSSIAN AND TIME-CORRELATED SIGNALS

Germán Gómez-Herrero[1], *Zbyněk Koldovský*[2], *Petr Tichavský*[2], *Karen Egiazarian*[1]

[1] Institute of Signal Processing, Tampere University of Technology, P.O.Box 553, 33101 Tampere, Finland
[2]Institute of Information Theory and Automation, P.O.Box 18, 18208 Prague, Czech Republic

ABSTRACT

In this article we propose a computationally efficient method (termed FCOMBI) to combine the strengths of non-Gaussianity-based Blind Source Separation (BSS) and cross-correlations-based BSS. This is done by fusing the separation abilities of two well-known BSS algorithms: EFICA and WASOBI. Simulations show that our approach is at least as accurate and often more accurate that other state-of-the-art approaches which also aim to separate simultaneously non-Gaussian and time-correlated components. However, in terms of computational efficiency and stability, FCOMBI is the clear winner which makes it specially suitable for the analysis of very high-dimensional datasets like high-density Electroencephalographic (EEG) or Magnetoencephalographic (MEG) recordings.

1. INTRODUCTION

In this article we consider the most common BSS problem in which the sources are assumed to be *independent* and the mixing is assumed to be *linear* and *instantaneous*. Such BSS problem can be solved by recovering independence in the source estimates through Independent Component Analysis (ICA). The BSS model can be compactly expressed as:

$$\mathbf{x}(t) = \sum_{j=1}^{d} \mathbf{a}_j s_j(t) = \mathbf{A}\mathbf{s}(t) \qquad (1)$$

where $\mathbf{A} = [\mathbf{a}_1, ..., \mathbf{a}_d]$ is an unknown mixing matrix, $\mathbf{s}(t) = [s_1(t), ..., s_d(t)]^T$ are the original unobserved sources and $\mathbf{x}(t) = [x_1(t), ..., x_d(t)]^T$ the observed linear and instantaneous mixtures. The BSS problem consists in estimating a separating matrix $\hat{\mathbf{B}} \approx \mathbf{A}^{-1} = \mathbf{B}$ such that the mixing process $\mathbf{A}$ can be inverted and the sources $\mathbf{s}$ recovered: $\hat{\mathbf{s}} = \hat{\mathbf{B}}\mathbf{x} = \hat{\mathbf{B}}\mathbf{A}\mathbf{s} \approx \mathbf{s}$. Different methods to solve the BSS problem usually differ in (1) the statistics measuring the *independence* of the source signals and (2) the estimators of those statistics. Indeed, the accuracy of the estimator is less important than the selection of an appropriate independence measure. The optimal choice for this measure depends upon the underlying model generating the source signals. Two of the most common choices for measuring independence are:

1. *Non-Gaussianity*. Maximizing non-Gaussianity of the estimated sources is a good choice when the original source signals are independent and identically distributed (i.i.d.) processes with non-Gaussian distribution. Non-Gaussianity can be measured using marginal entropy for which several accurate estimators have been proposed in the BSS framework (e.g. [1, 2]). Two of the best algorithms in terms of speed and accuracy are FastICA [1] and EFICA [3].
2. *Cross-correlations*. This is a simple and effective independence contrast when the sources are time series with non-zero autocorrelations for time lags greater than zero. In this case, the true sources can be identified by minimizing cross-correlations and maximizing autocorrelations in the estimated sources. Algorithms in this group include SOBI/TDSEP [4, 5] and Weights-Adjusted SOBI (WASOBI [6, 7, 8]).

In real applications, it is common to find mixtures of non-Gaussian i.i.d. sources with Gaussian non-white sources. In such scenarios each of the independence contrasts above will be (at best) able to separate just some of the sources but never all of them. A compromise solution is to try to optimize a weighted average of both types of contrasts. Several BSS algorithms have been proposed in this direction, including JADE_{TD} [9], JCC [10], Thin ICA [11] and the unifying model of [12]. A probably more accurate approach is to successively use each of the contrasts above to separate the sub-sets of sources for which they are more suitable. The algorithms EFWS [13], COMBI [13] and MCOMBI [14] implement this idea and effectively combine the strengths of EFICA and WASOBI. A practical limitation of all these combination approaches is that their computational cost is unaffordable for high-dimensional mixtures like the ones found in high-density electroencephalography (EEG) and magnetoencephalography (MEG). In this article we propose a new version of MCOMBI that overcomes this limitation. Using simulations, we show that the new version (termed FCOMBI) achieves a performance similar to that of MCOMBI with just a small fraction of computational load.

2. MULTIDIMENSIONAL INDEPENDENT COMPONENTS

Standard BSS assumes that the one-dimensional unknown sources in Eq. 1 are mutually independent according to the independency contrast used. A straightforward generalization of this principle assumes that not all the d sources are mutually independent but they form M higher dimensional independent components [15, 16]. Let d_l denote the dimensionality of the lth multidimensional component that groups together the one-dimensional source signals with indexes $l_1, ..., l_{d_l}$. Then, the lth multidimensional compo-

[0]This work was supported by the Academy of Finland, project No. 213462 (Finnish Centre of Excellence program 2006 - 2011), by the Czech Science Foundation (GA ČR) through the projects 102/05/0278 and 102/07/P384 and by Ministry of Education, Youth and Sports of the Czech Republic through the project 1M0572.

nent is given by $\mathbf{S}_l = [\mathbf{s}_{l_1}, ..., \mathbf{s}_{l_{d_l}}]^\dagger$, where $l = 1, ..., M$ and $d_1 + d_2 + ... + d_M = d$. Therefore, we can rewrite the sources data matrix $\mathbf{S}$ in Eq. 1 as $\mathbf{S} = [\mathbf{s}_1, ..., \mathbf{s}_d]^\dagger = \mathbf{Q}[\mathbf{S}_1, ..., \mathbf{S}_M]^\dagger$ where $\mathbf{Q}$ is a permutation matrix. Using the notation above and dropping matrix $\mathbf{Q}$ under the permutation indeterminacy of ICA, we can reformulate Eq. 1 as:

$$\mathbf{S} = \mathbf{B}\mathbf{X} = [\mathbf{B}_1\mathbf{X}, ..., \mathbf{B}_M\mathbf{X}]^\dagger = [\mathbf{S}_1, ..., \mathbf{S}_M]^\dagger \quad (2)$$

The goal of multidimensional BSS is to estimate the submatrices $\{\mathbf{B}_l\}_{l=1,...,M}$ each of which is of dimension $d_l \times d$. Since the sub-components of a multidimensional independent component are arbitrarily mixed we can recover $\{\mathbf{B}_l\}_{l=1,...,M}$ only up to an invertible matrix factor [16].

A multidimensional component according to certain independency contrast (e.g. non-Gaussianity) might be separable into one-dimensional components using an alternative independency measure (e.g. cross-correlations). This suggests a procedure for combining complementary independency criteria [14]:

1. Try BSS using certain independency criterion.
2. Detect the presence of multidimensional components in the source signals estimated in step (1).
3. Try BSS using an alternative independency contrast in each multidimensional component found in step (2).

This is the basic idea underlying FCOMBI which combines the complementary strengths of the non-Gaussianity criterion of EFICA and the criterion based on cross-correlations of WASOBI.

3. DETECTION OF MULTIDIMENSIONAL INDEPENDENT COMPONENTS

A common way of evaluating the accuracy of the separation produced by any BSS algorithm is the matrix of Interference-To-Signal Ratios (**ISR** matrix). Element-wise, the **ISR** matrix is defined as $\mathbf{ISR}_{kl} = \mathbf{G}_{kl}^2/\mathbf{G}_{kk}^2$ where $\mathbf{G} = \hat{\mathbf{B}}\mathbf{A}$. $\hat{\mathbf{B}}$ is the estimated separating matrix and $\mathbf{A}$ is the true mixing matrix. $\mathbf{ISR}_{kl}$ measures the level of residual interference between the kth and lth estimated components. The total ISR of the kth estimated source is defined as $\mathbf{isr}_k = \sum_{l=1,l\neq k}^{d} \mathbf{ISR}_{kl}$.

EFICA and WASOBI share the rare feature of allowing the estimation of the obtained **ISR** matrix through simple empirical estimate of E[**ISR**] using the estimated sources $\hat{\mathbf{s}}$. This means that EFICA and WASOBI permit us to estimate $\widehat{\mathbf{ISR}} \approx \mathrm{E}[\mathbf{ISR}]$. It has been shown that the estimations $\widehat{\mathbf{ISR}}$ obtained by WASOBI and EFICA are quite accurate even when the respective assumptions about the sources are only partially fulfilled [14]. The information provided by $\widehat{\mathbf{ISR}}$ is crucial for detecting the presence of multidimensional components within the estimated sources which is the reason for us to choose EFICA and WASOBI in our combined BSS method.

If the **ISR** matrix is known, or if it can be estimated, we can easily assess the presence of multidimensional independent components by grouping together components with high mutual interference. This is done by defining a symmetric distance measure between two estimated components $D(\hat{\mathbf{s}}_k, \hat{\mathbf{s}}_l) = D_{kl} = 1/(\mathbf{ISR}_{kl} + \mathbf{ISR}_{lk}) \geq 0\ \forall l \neq k$. We also define $D_{kk} = 0\ \forall k$. Using the distance metric D, we cluster together the estimated components whose distance from each other is small. For this task we use agglomerative hierarchical clustering with single linkage. By single linkage we mean that the distance between clusters of components is defined as the distance between the closest pair of components. The output of this clustering algorithm is a set of $i = 1, 2, .., d$ possible partition levels of the estimated sources. At each particular level the method joins together the two clusters from the previous level which are closest in distance. Therefore, in level $i = 1$ each source forms a cluster whereas in level $i = d$ all the sources belong to the same cluster. For assessing the goodness-of-fit of the $i = 2, ..., d-1$ partition levels we propose using the validity index $I_i = D_i^{intra}/D_i^{inter}$ where D_i^{intra} and D_i^{inter} roughly measure, respectively, the average intra-cluster and inter-cluster distances. They are defined, for $1 < i < d$, as follows:

$$D_i^{intra} = \frac{\sum_{j=1,\mathrm{Card}(\Gamma_{i,j})>1}^{d-i+1} \mathrm{Card}(\Gamma_{i,j})(\mathrm{Card}(\Gamma_{i,j}) - 1)/2}{\sum_{j=1,\mathrm{Card}(\Gamma_{i,j})>1}^{d-i+1} \sum_{k\in\Gamma_{i,j}, l\in\Gamma_{i,j}} \mathbf{ISR}_{kl}} \quad (3)$$

$$D_i^{inter} = \frac{\sum_{j=1}^{d-i+1} \mathrm{Card}(\Gamma_{i,j})(d - \mathrm{Card}(\Gamma_{i,j}))}{\sum_{j=1}^{d-i+1} \sum_{k\in\Gamma_{i,j}, l\notin\Gamma_{i,j}} \mathbf{ISR}_{kl}} \quad (4)$$

where $\Gamma_{i,j}$ is the set of indexes of the sources belonging to the jth cluster at the ith partition level. We also define $I_1 = 1/ISR_{max}$ where ISR_{max} is the maximum entry in the **ISR** matrix. We set $I_d = 10$. Finally we choose the best cluster partition to be that one corresponding to the maximum of all local maxima of the cluster validity index I. By setting $I_d = 10$ we consider that the separation failed completely (there is just one d-dimensional cluster) if $D_i^{inter} < 10 \cdot D_i^{intra}\ \forall i = 2, ..., d-1$. The definition $I_1 = 1/ISR_{max}$ means that the estimated sources will be considered to be 1-dimensional (perfect separation) if $ISR_{max} < min_{i>2}(1/I_i)$. Therefore, since $I_d = 10$, we require the maximum ISR between two 1-dimensional components to be in any case below -10 dB. In Fig. 1 we can see the results of clustering the ISR estimated by WASOBI and EFICA for a simulated dataset.

In order to ease the explanation of FCOMBI in the next section we will use the following Matlab notation to refer to the hierarchical clustering algorithm described in this section: `[i,I] = hclus(ISR)`, where the input parameter is the estimated **ISR** matrix, the first output parameter is the selected partition level and the second output parameter is a $1 \times (d - i + 1)$ cell array such that `I{k}` is a vector containing the indexes of the sources belonging to the kth cluster.

4. PROPOSED ALGORITHM: FCOMBI

FCOMBI is described below using Matlab notation:

```
function [B] = FCOMBI(X,ARorder)
[d, L] = size(X);
[B, ISRwa] = WASOBI(X,ARorder);
[iwa, Iwa] = hclus(ISRwa);
if iwa == 1, return; end
for i = 1:(d-iwa+1),
    if length(Iwa{i})==1, continue; end
    index = Iwa{i}; di = length(index);
```

```
    [Bef,ISRef] = EFICA(B(index,:)*X);
    [ief, Ief] = hclus(ISRef);
    if (ief < di) || ...
     (min(sum(ISR(index,index),2)) > ...
      min(sum(ISRef,2))),
          B(index,:) = Bef*B(index,:);
    end
end
```

FCOMBI starts by applying WASOBI on the input data. The reason for using WASOBI first instead of EFICA is that the former is considerably faster than the latter for high dimensional mixtures, which is the target application of FCOMBI. Subsequently, EFICA is applied on each multidimensional component of sources found in the output of WASOBI. Finally, we decide whether EFICA was able to improve the separation of the sources within the cluster or not. In our implementation of the algorithm we include a third step (not shown in the Matlab code above) that consists on running WASOBI again on the cluster of unresolved components in the output of EFICA (if such a cluster exists). This last step is helpful only in the rare cases when, in the first run of WASOBI, we were not able to detect the correct clusters. If EFICA was able to separate some non-Gaussian sources we expect the accuracy of WASOBI to improve by applying it only to the cluster of Gaussian components that was not correctly separated by EFICA. WASOBI requires the user to specify the order of the AR model that best fits the unobserved sources. However, the performance is not critically dependent on this parameter and it is enough to select an order high enough to model appropriately the source signals.

5. SIMULATIONS

In this section we use simulations to compare FCOMBI to other state-of-the-art approaches that also aim to combine non-Gaussianity and cross-correlations to measure independence of the source signals, namely EFWS, COMBI, MCOMBI, JADE_{TD}, ThinICA, and JCC. The implementations of these algorithms were obtained from their respective author's public web-pages or provided directly to us by the authors (JCC). The only exception was JADE_{TD} which we implemented using the publicly available implementations of JADE [2] and TDSEP. Unfortunately, the implementation of the unifying model of [12] kindly provided by his author did not allow the separation of AR sources of order greater than 1. Due to the lack of time to implement the necessary changes in the code we decided not to include it in the comparison.

Both JADE_{TD} and JCC are based on joint diagonalization of quadricovariance eigen matrices and time-delayed correlation matrices. In both algorithms we selected the time lags of the cross-correlations to be $0,1,...,K$ where K denotes the maximum AR order of the source signals (see Table 1). In the contrast function of ThinICA we included second order cross-correlations at lags $0,1,...,K$, third order statistics at lag 0 (skewness) and fourth order cumulants at lags $0,1,...,K$. For the algorithms that made use of WASOBI (EFWS, COMBI, MCOMBI and FCOMBI) we set the order of the AR models employed to be equal to K. A Matlab implementation of FCOMBI as well as the Matlab scripts necessary for repeating the figures shown in this paper can be downloaded from the web-page of the first author of this article [1].

The source signals were generated by feeding Auto-Regressive (AR) filters with random i.i.d. samples with different distributions. The characteristics of the sources are summarized in Table 1. We can see that the simulated dataset consists of $d = 4\cdot K + G$ sources. The first K sources have a Gaussian distribution and therefore cannot be separated by means of the non-Gaussianity ICA contrast. By contrary sources $K+1$ to $4\cdot K$ are all are easily separated by exploiting their non-Gaussianity. It can also be observed that for a fixed value of m, the sources with indexes $n\cdot K+m$ for $n = 1,2,3,4$ have the same spectrum and therefore cannot be separated by means of SOBI, WASOBI or other algorithms exploiting different spectra of the source signals. Sources $4\cdot K+1$ to $4\cdot K+G$ are Gaussian i.i.d. which means that they cannot be separated by any of the tested algorithms. The multidimensional structure of the simulated dataset for $K = 5$, $G = 0$ can be observed in Fig. 1.

Source #	distribution	AR filter coefficients
1	Gaussian	$[1,\rho]$
2	Gaussian	$[1,0,\rho]$
$\vdots$	$\vdots$	$\vdots$
K	Gaussian	$[1,0,\cdots,0,\rho]$
$K+1$	BPSK	$[1,\rho]$
$\vdots$	$\vdots$	$\vdots$
$2\cdot K$	BPSK	$[1,0,\cdots,0,\rho]$
$2\cdot K+1$	Laplacian	$[1,\rho]$
$\vdots$	$\vdots$	$\vdots$
$3\cdot K$	Laplacian	$[1,0,\cdots,0,\rho]$
$3\cdot K+1$	Uniform	$[1,\rho]$
$\vdots$	$\vdots$	$\vdots$
$4\cdot K$	Uniform	$[1,0,\cdots,0,\rho]$
$4\cdot K+1$	Gaussian	$[1]$
$\vdots$	$\vdots$	$\vdots$
$4\cdot K+G$	Gaussian	$[1]$

Table 1: Characteristics of the source signals

To evaluate the overall separation performance we used the average of the ISR obtained for the individual sources, i.e.: $ISR_{avg} = \frac{1}{d}\sum_{k=1}^{d}\mathbf{isr}_k$. In Fig. 2(a) we show the average ISR obtained for different number of data samples of the sources. The computation times for different sample sizes are in Fig. 3(a). The accuracy of FCOMBI for values of ρ between 0.2 and 0.4 is lower than the accuracy of MCOMBI. The reason is that both WASOBI and EFICA were able to produce a rather accurate separation of the sources, and therefore FCOMBI chose WASOBI to separate them even if EFICA would have been even more accurate. COMBI and EFWS performance decreases with increasing number of data samples which is due to the fact that those three algorithms are not able to separate the Gaussian sources (sources 1-5). For small sample sizes, the sample distribution of those sources is not exactly Gaussian which explains the better performance. From Fig. 3(b) we can observe that, for low-dimensional problems, JADETD_{TD} is the fastest algorithm and is still able to separate all the sources (although not as

[1] http://www.cs.tut.fi/-gomezher/

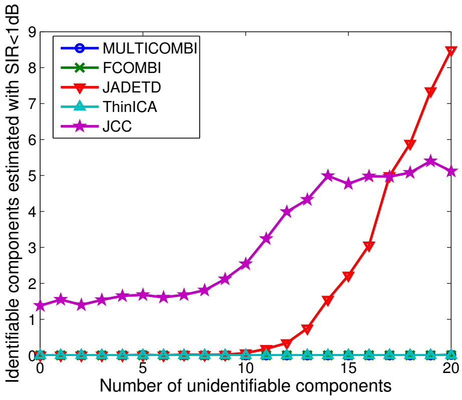


Figure 4: Number of identifiable components that were estimated with SIR $<$ 1 dB versus the number of Gaussian i.i.d. (unidentifiable) components (parameter G in Table 1). The settings of the simulation were $\rho = 0.6$, $K = 5$, $N = 5000$.

accurately as MCOMBI and FCOMBI). Therefore, we can conclude that, for low-dimensional mixtures, FCOMBI and MCOMBI offer the best trade-off between computational complexity and accuracy but they are closely followed by JADE$_{TD}$.

The major advantage of FCOMBI is the possibility of using it with very high dimensional datasets. This can be easily checked from Fig. 3(b) where we show the computation times for different dimensionalities of the mixture. The computational cost of JADE$_{TD}$ grows exponentially with increasing dimensionality which makes it completely unsuitable for analysis of high-dimensional datasets. FCOMBI performs much faster than any other algorithm and is among the most accurate (accuracy results for different values of K are not shown for lack of space). This makes FCOMBI the best choice for the analysis of high-dimensional mixtures.

Finally, we tested the robustness of the algorithm when we keep constant the number of data samples and we increase the number of unresolvable Gaussian i.i.d. components in the mixture, i.e. when we increase the value of parameter G. The robustness was assessed by counting the number of identifiable components (sources 1 to $4 \cdot K$) that obtained an ISR of less than 1 dB. The average results for 100 Monte-Carlo repetitions of the sources are shown in Fig. 4. FCOMBI, MCOMBI and ThinICA are clearly more stable than JCC and JADE$_{TD}$.

6. CONCLUSIONS

We proposed a BSS algorithm (FCOMBI) that simultaneously separates non-Gaussian and time-correlated sources. FCOMBI is almost as accurate as the closely related algorithm MCOMBI and more accurate than JADE$_{TD}$, JCC and ThinICA. However, due to its low computational cost, FCOMBI is the only realistic choice for the analysis of high dimensional mixtures like the ones found in high-density EEG and MEG.

REFERENCES

[1] A. Hyvärinen, "Fast and robust fixed-point algorithms for Independent Component Analysis," *IEEE T. on Neural Networks*, vol. 10, no. 3, pp. 626–634, 1999.

[2] J.-F. Cardoso, "High-order contrasts for Independent Component Analysis," *Neural Computation*, pp. 157–192, 1999.

[3] Z. Koldovský, P. Tichavský, and E. Oja, "Efficient variant of algorithm fastica for independent component analysis attaining the cramer-rao lower bound," *IEEE T. Neural Networks*, vol. 17, no. 5, pp. 1265–1277, 2006.

[4] A. Belouchrani, K. Abed Meraim, J.-F. Cardoso, and E. Moulines, "A blind source separation technique based on second order statistics," *IEEE T. on Signal Processing*, vol. 45, no. 2, pp. 434–444, 1997.

[5] A. Ziehe and K.-R. Müller, "TDSEP—an efficient algorithm for blind separation using time structure," in *Proc. ICANN'98*, 1998, pp. 675–680.

[6] A. Yeredor, "Blind separation of Gaussian sources via second-order statistics with asymptotically optimal weighting," *IEEE Signal Processing Letters*, vol. 7, no. 7, pp. 197–200, 2000.

[7] E. Doron and A. Yeredor, "Asymptotically optimal blind separation of parametric Gaussian sources," in *Proc. ICA'04, Granada, Spain*, 2004.

[8] P. Tichavský, E. Doron, and A. Yeredor, "A computationally affordable implementation of an asymptotically optimal BSS algorithm for AR sources," in *Proc. EUSIPCO'06, Florence, Italy*, 2006.

[9] K.-R. Müller, P. Philips, and A. Ziehe, "JADE$_{TD}$: Combining higher-order statistics and temporal information for blind source separation (with noise)," in *Proc. ICA'99, Aussois, France*, 1999, pp. 87–92.

[10] I. F. Gorodnitsky and A. Belouchrani, "Joint cumulant and correlation based signal separation with application to EEG data," in *Proc. ICA'01, San Diego, CA, USA*, 2001, pp. 475–480.

[11] S. A. Cruces-Álvarez and A. Cichocki, "Combining blind source extraction with joint approximate diagonalization: Thin algorithms for ICA," in *Proc. ICA'03, Nara, Japan*, 2003, pp. 463–469.

[12] A. Hyvärinen, "A unifying model for blind separation of independent sources," *Signal Processing*, vol. 85, no. 7, pp. 1419–1427, 2005.

[13] P. Tichavský, Z. Koldovský, E. Doron, A. Yeredor, and G. Gómez-Herrero, "Blind signal separation by combining two ICA algorithms: HOS-based EFICA and time structure-based WASOBI," in *Proc. EUSIPCO'06, Florence, Italy*, 2006.

[14] P. Tichavský, Z. Koldovský, A. Yeredor, G. Gómez-Herrero, and E. Doron, "A hybrid technique for blind separation of non-Gaussian and time-correlated sources using a multicomponent approach," *Submitted to IEEE Trans. Neural Networks*, 2007.

[15] L. De Lathauwer, D. Callaerts, B. De Moor, and J. Vandewalle, "Fetal electrocardiogram extraction by source subspace separation," in *Proc. IEEE Signal Processing workshop on higher-order statistics, Girona, Spain, June 12–14*, 1995, pp. 134–138.

[16] J.-F. Cardoso, "Multidimensional independent component analysis," in *Proc. ICASSP'98*, Seattle, WA, 1998.

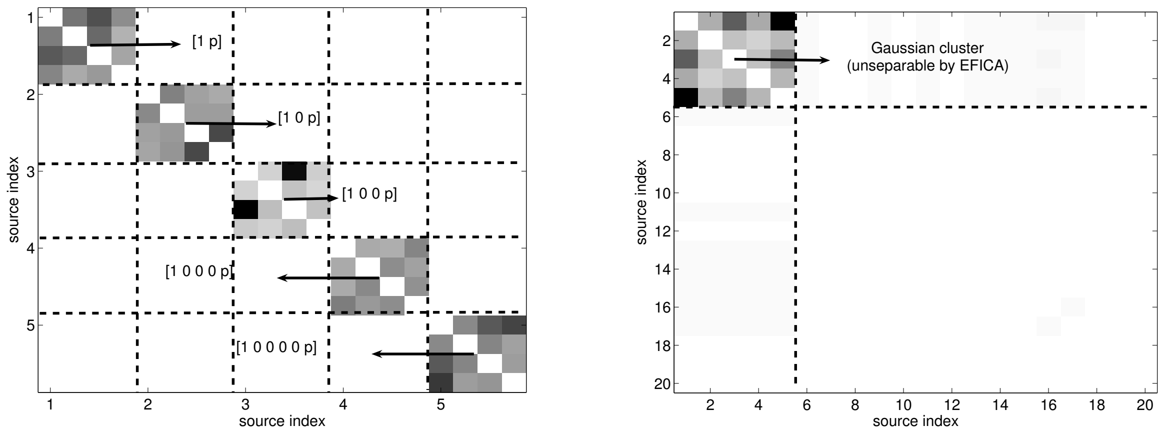
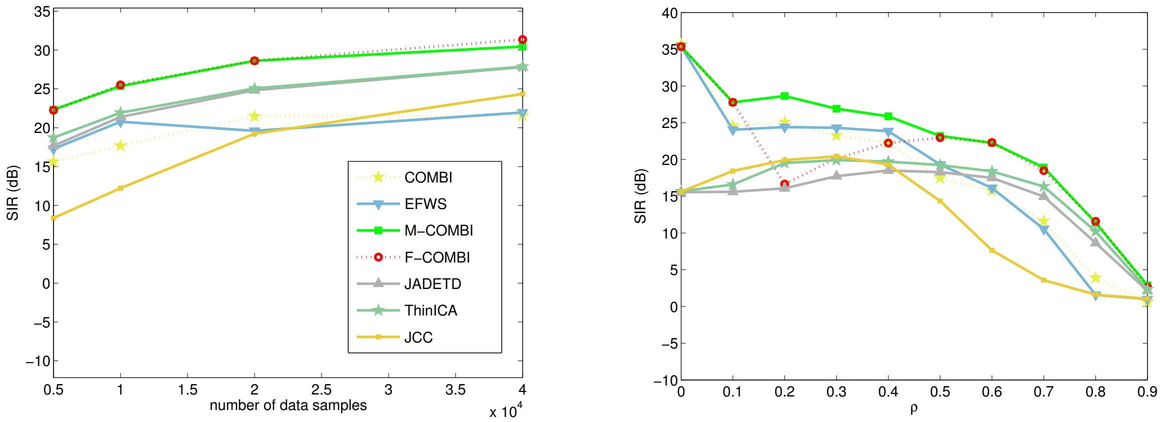


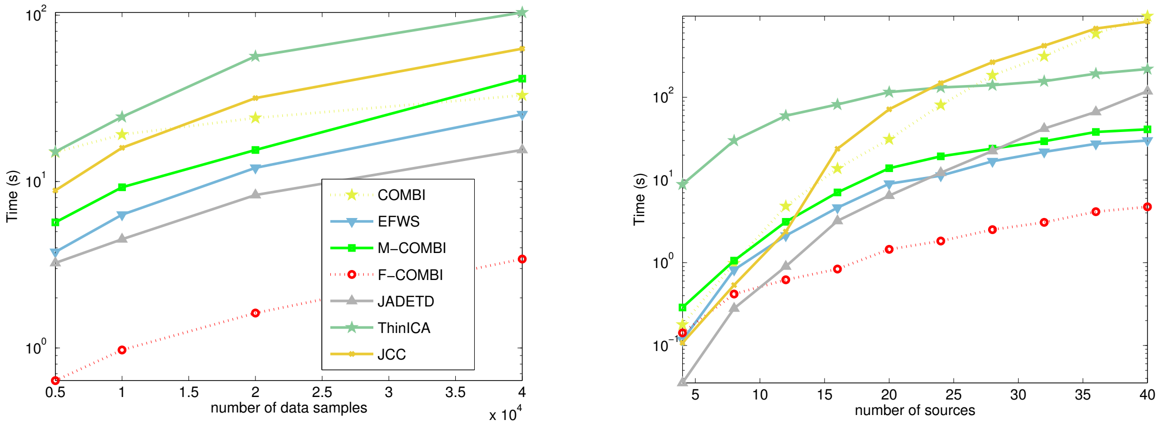
Figure 1: The results of clustering the ISR matrices estimated by WASOBI (left) and EFICA (right) for the simulated dataset when $K = 5$, $G = 0$, $N = 5000$, and $\rho = 0.6$. The best validity index for WASOBI was obtained at level $i = 16$, i.e. the clustering algorithm found $d - i + 1 = 5$ clusters. In the case of the ISR matrix estimated by EFICA, the best clustering level was $i = 5$, i.e. the clustering algorithm found $d - 5 + 1 = 16$ clusters.



(a) SIR versus N ($\rho = 0.6$, $K = 5$, $G = 0$)

(b) SIR versus ρ ($N = 5000$, $K = 5$, $G = 0$)

Figure 2: Average SIR when varying the number of data samples N and when varying the value of the coefficient of the AR filters ρ. The values shown are the average results for 100 Monte-Carlo repetitions of the sources.



(a) Time versus N ($\rho = 0.6$, $K = 5$, $G = 0$)

(b) Time versus $d = 4 \cdot K$ ($N = 15000$, $\rho = 0.6$, $G = 0$)

Figure 3: Average computational time when varying the number of observed data samples N and the number of sources $d = 4 \cdot K$. Note the different scale of the y-axes. The values shown are the average results for 100 Monte-Carlo repetitions of the sources.