COMPARISON OF ICA/BSS ALGORITHMS FOR NOISY DATA

Sang-Hoon OH[†], *Andrzej CICHOCKI*[†], *Seungjin CHOI*[§], *Shun-ichi AMARI*[†], *and Soo-Young LEE*[‡]

[†]Lab for Advanced Brain Signal Processing , Brain Science Institute, RIKEN, Japan
[§]Department of Electrical Engineering, Chungbuk National University, Korea
[‡]Division of Electrical Engineering, Korea Advanced Institute of Science and Technology, Korea

ABSTRACT

Various blind source separation (BSS) and independent component analysis (ICA) algorithms have been developed. However, comparison study for BSS/ICA algorithms has not been extensively carried out yet. The main objective of this paper is to compare various promising BSS/ICA algorithms in terms of several factors such as robustness to sensor noise, computational complexity, and the conditioning of the mixing matrix. We propose several benchmarks which are useful for the evaluation of the algorithm. This comparison study will be useful for real-world applications, especially EEG/MEG analysis and future VLSI implementation of these algorithms.

1. INTRODUCTION AND PROBLEM FORMULATION

The mixing and filtering processes of unknown input sources $\{s_j(t)\}$ $(j = 1, 2, ..., n)$ may have different mathematical or physical models, depending on specific applications. In this paper we will focus mainly on the simplest cases when m mixed signals $x_i(t)$ are a linear combination of the n (typically $m \geq n$) unknown statistically independent, zero mean source signals $s_j(t)$, and are noise-contaminated. They are written as

$$x_i(t) = \sum_{j=1}^{n} a_{ij} s_j(t) + \nu_i(t) \qquad i = 1, 2, ..., m \tag{1}$$

or in the matrix notation (see Fig. 1)

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) + \nu(t), \tag{2}$$

where $\mathbf{x}(t) = [x_1(t) \cdots x_m(t)]^T$ is a noisy sensor vector, $\mathbf{s}(t) = [s_1(t) \cdots s_n(t)]^T$ is a source signal vector, $\nu(t) = [\nu_1(t) \cdots \nu_m(t)]^T$ is a noise vector, and $\mathbf{A}$ is an unknown full rank $m \times n$ mixing matrix. In this paper, it is assumed that the number of source signals n is, in general, unknown unless stated otherwise. It is assumed that only the sensor vector $\mathbf{x}(t)$ is available and it is necessary to design a feed-forward or recurrent neural network and an associated adaptive learning algorithm that enables estimation of sources and/or identification of the mixing matrix $\mathbf{A}$ with good tracking abilities.

The above problems are often referred to as noisy BSS (blind source separation) or ICA (independent component analysis).

There are several definitions of ICA and BSS. In this paper, we employ the following definitions :

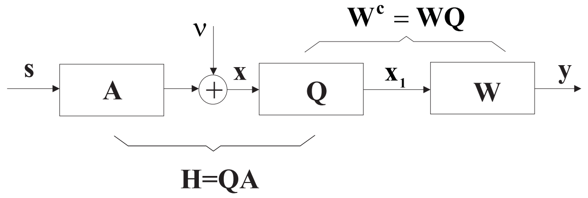


Figure 1: Mixing ($\mathbf{A}$), prewhitening ($\mathbf{Q}$), and blind separation model ($\mathbf{WQ}$).

Definition 1: The ICA of a noisy random vector $\mathbf{x}(t) = [x_1(t) \cdots x_m(t)]^T$ is obtained by finding an $n \times m$, (with $m \geq n$), full rank, linear transformation separating matrix $\mathbf{W}^c$ such that the output signal vector $\mathbf{y}(t) = [y_1(t) \cdots y_n(t)]^T$ defined as

$$\mathbf{y}(t) = \mathbf{W}^c \mathbf{x}(t), \tag{3}$$

contains the source components that are as independent as possible, evaluated by an information-theoretic cost function such as the minimum Kullback–Leibler divergence.

Definition 2: The BSS of $\mathbf{x}(t)$ is to estimate simultaneously all the source signals and their numbers and/or identify (possibly unbiased by noise) a mixing matrix $\widehat{\mathbf{A}}$ or its pseudo inverse separating matrix $\mathbf{W}^c = \widehat{\mathbf{A}}^+$ assuming only the stochastic independence of the primary sources, linear independence of columns of $\mathbf{A}$, and $m \leq n$.

Similarly, we can define Blind Extraction of Sources.

Definition 3: The BSE (blind source extraction) of $\mathbf{x}(t)$ is to estimate one independent component or a selected number of the components sequentially one by one or estimate a specified group of sources and identify the corresponding vector(s) $\widehat{\mathbf{a}}_j$ of a mixing matrix $\widehat{\mathbf{A}}$ and their pseudo inverses $\mathbf{w}^c{}_j$ which are rows of the demixing matrix $\mathbf{W}^c = \widehat{\mathbf{A}}^+$ assuming only the stochastic independence of its primary sources and linear independence of columns of $\mathbf{H}$.

Remark: It is worth emphasizing that in the literature the definitions of BSS/BSE and ICA are often confused, although, they refer to the same model usually with the same assumptions, pursue similar tasks and are solved with the same algorithms if primary sources are independent. However, generally for real-world problems the objective and results can be completely different. The objective of blind source separation is to estimate the original source signals even if they are not completely statistically independent or they are correlated, while the objective of ICA is to determine such a transformation which assures that the output signals are as independent as possible.

It should be noted that, in many cases, ICA methods use higher order statistics (HOS) while BSS ones use only second order statistics (SOS) if sources are temporally correlated or their variances are time varying. Thus HOS based ICA methods can not be applied to the case where more than one Gaussian sources exist, whrereas SOS based BSS methods do not have such constraints. In fact, BSS methods do not really replace ICA and vice versa , since each approach is based on different assumptions and often different objectives.

Recently, several promising and efficient algorithms has been proposed for ICA/BSS problems. However, it is not clear which of these algorithms is the best or most promising regarding robustness to large noise and efficiency (computation time and complexity of implementation), especially for huge and noisy biomedical data like MEG/EEG. Such comparison is the primary objective of this paper. The second objective is to propose several modifications of existing algorithms and a sub optimal choice of their parameters in order to improve their performance for data corrupted by white additive noise. The third objective is to propose several difficult benchmarks which can be used for testing and comparison of existing and future algorithms.

In fact, recently we observe explosion and development of new algorithms and each author claims some advantages over previous algorithms. Some flexible and universal benchmarks are in our opinion necessary in order to make fair comparison.

2. SOS BASED BSS AND ICA

In this section, we propose a robust prewhitening and briefly review ICA and BSS algorithms.

2.1. Robust prewhitening

In many cases sensor signals are prewhitened before separation for faster convergence and better performance. A standard whitening is a procedure to find such a linear transformation $\mathbf{x}_1(t) = \mathbf{Q}\mathbf{x}(t)$ that

$$E\{\mathbf{x}_1(t){\mathbf{x}_1}^T(t)\} = \mathbf{I}. \tag{4}$$

The simple batch (block) method for prewhitening is to use the eigenvalue decomposition of the covariance matrix

$$\mathbf{R}_{xx} = E\{\mathbf{x}(t)\mathbf{x}^T(t)\} = \mathbf{V}\Lambda\mathbf{V}^T, \tag{5}$$

where $\mathbf{V}$ is the orthogonal matrix of eigenvectors of $\mathbf{R}_{xx}$ and Λ is a diagonal matrix of its eigenvalues. Prewhitening can be done by

$$\mathbf{x}_1 = \mathbf{Q}\mathbf{x} = \mathbf{V}\Lambda^{-1/2}\mathbf{V}^T\mathbf{x} \tag{6}$$

or

$$\mathbf{x}_1 = \mathbf{Q}\mathbf{x} = \Lambda^{-1/2}\mathbf{V}^T\mathbf{x}. \tag{7}$$

It is easy to check that now $\mathbf{R}_{x_1x_1} = E\{\mathbf{x}_1(t){\mathbf{x}_1}^T(t)\} = \mathbf{I}$. Unfortunately, such a standard prewhitening is biased under white additive noise $\nu(t)$.

In order to obtain unbiased prewhitening by noise, we perform eigenvalue decomposition for the modified covariance matrix

$$\tilde{\mathbf{R}}_{xx}(\tau) = \frac{1}{2}[E\{\mathbf{x}(t)\mathbf{x}^T(t-\tau)\} + E\{\mathbf{x}(t-\tau)\mathbf{x}^T(t)\}] \tag{8}$$

where τ is small time delay typically one or two samples. Alternatively, for a very ill conditioned problem, we can apply the learning algorithm with equivalent properties

$$\begin{aligned}\Delta\mathbf{Q}(k) = \quad & \eta\{\mathbf{I} - \tfrac{1}{2}[< \mathbf{x}_1(t){\mathbf{x}_1}^T(t-\tau) > \\ & + < \mathbf{x}_1(t-\tau){\mathbf{x}_1}^T(t) >]\}\mathbf{Q}(k).\end{aligned} \tag{9}$$

Remark: It should be noted that such robust prewhitening can be applied only to non i.i.d.(independent, identically distributed) source signals, i.e. for sources with temporal structures.

2.2. Fast ICA algorithm (Hyvarinen & Oja)

The fast ICA algorithm proposed by Hyvarinen and Oja[6] minimizes or maximizes kurtosis of output signals $\mathbf{y} = \mathbf{w}^T\mathbf{x}$ based on the fact that finding the local extrema of kurtosis is equivalent to estimating non-Gaussian independent components. They proposed an objective function

$$J(\mathbf{w}) =< (\mathbf{w}^T\mathbf{x})^4 > -3||\mathbf{w}||^4 + F(||\mathbf{w}||^2) \tag{10}$$

where F is a term for the constraint $||\mathbf{w}||^2 = 1$. From this objective function, they developed the simple and efficient learning rule

$$\mathbf{w}^+ =< \mathbf{x}(\mathbf{w}^T(k)\mathbf{x})^3 > -3\mathbf{w}(k) \tag{11}$$

where k denotes a iteration number and $\mathbf{w}(k+1) = \mathbf{w}^+/||\mathbf{w}^+||$.

2.3. NG (Amari et al.) and Flexible ICA algorithm (Choi, Cichocki, & Amari)

The natural gradient (NG) developed rigorously by Amari et al.[1, 2] can be written in a general form

$$\Delta\mathbf{W}(k) = \eta\mathbf{F}(\mathbf{y},\mathbf{W})\mathbf{W}(k), \tag{12}$$

where the nonlinear function $\mathbf{F}(\mathbf{y},\mathbf{W})$ can take different forms for example

$$\mathbf{F}(\mathbf{y},\mathbf{W}) = [\Lambda - < \varphi(\mathbf{y})\mathbf{y}^T >] \tag{13}$$

or

$$\begin{aligned}\mathbf{F}(\mathbf{y},\mathbf{W}) = \quad & [\Lambda - < \mathbf{y}\mathbf{y}^T > - < \varphi(\mathbf{y})\mathbf{y}^T > \\ & + < \mathbf{y}\varphi^T(\mathbf{y}) >]\end{aligned} \tag{14}$$

where Λ is a diagonal matrix whose diagonal elements are the same with ones in $< \varphi(\mathbf{y})\mathbf{y}^T >$.

The flexible ICA algorithm[3] based on generalized Gaussian model of probability density function of source signals estimates nonlinear activation functions as follows

$$\varphi(y_i) = |y_i|^{\alpha_i - 1}\mathrm{sgn}(y_i) \tag{15}$$

where α_i can be estimated, e.g. as follows[3]

$$\alpha_i = \begin{cases} 0.8, & \text{if } \kappa(y_i) > 20, \\ 1, & \text{if } 5 < \kappa(y_i) \le 20, \\ 1.3, & \text{if } 0 < \kappa(y_i) \le 5, \\ 4, & \text{otherwise} \end{cases} \tag{16}$$

and $\kappa(y_i) = E\{y_i^4\}/E^2\{y_i^2\} - 3$ is a normalized kurtosis.

2.4. Cumulant-based iterative inversion(CII) algorithm (Cruces, Cichocki, & Castedo)

As a generalization of the fast ICA algorithm, Cruces et al. proposed a cumulant-based algorithm to estimate the mixing matrix[4]. This algorithm can work in the presence of additive Gaussian noise since the high-order cumulants of joint Gaussian processes are null. In this algorithm, we can decompose the mixing matrix $\mathbf{A} = [\mathbf{A}_L|\mathbf{A}_R]$ and extract a part of sources related to $\mathbf{A}_L$. Defining $\mathbf{B}_L = \mathbf{Q}\mathbf{A}_L$ as the left part of the overall estimated mixing after pre-whitening, $\mathbf{B}_L$ is iteratively estimated as

$$\mathbf{B}_L(k+1) = \mathbf{B}_L(k) + \mu(k)(\mathbf{C}^{1,3}_{z,y}\mathbf{S}^3_y - \mathbf{B}_L(k)) \tag{17}$$

where $\mathbf{C}^{1,3}_{a,b}$ denotes the fourth order cross-cumulant matrix of elements

$$[\mathbf{C}^{1,3}_{a,b}]_{ij} = Cum(a_i[n], b_j^*[n], b_j[n], b_j^*[n]) \tag{18}$$

and $\mathbf{S}_a$ denotes a diagonal matrix of cumulants signs

$$[\mathbf{S}_a]_{ii} = sign([\mathbf{C}^{1,3}_{a,a}]_{ii}). \tag{19}$$

And alternatively we update

$$\mathbf{W}(k) = (\mathbf{B}_L(k))^+. \tag{20}$$

In simulations, we extract sources one-by-one using this algorithm.

2.5. Nonstationary source separation based on joint diagonalization (Choi & Cichocki)

The NSS-TD-JD algorithm was proposed as a second ordered BSS algrithm incoporating nonstationarity and temporal information of sources[5]. This is based on joint diagonalization and only needs multiple time-delayed correlation matrices of sensor signals at several time-windowed frames. The noise robustness of this algorithm is due to the time-delayed correlation since the time-delayed correlation of white noise is zero. At time frame t_k, the τ_j delayed correlation matrix is given by

$$\mathbf{M}_x(t_k,\tau_j) = \frac{1}{2}[\mathbf{R}_x(t_k,\tau_j) + \mathbf{R}_x^T(t_k,\tau_j)] \tag{21}$$

where $\mathbf{R}_x(t_k,\tau_j) = E\{\mathbf{x}(t_k)\mathbf{x}^T(t_k - \tau_j)\}$ is evaluated using the samples in the kth time-windowed frame. The $\{\mathbf{M}_x(t_k,\tau_j)\}$ displays different statistics for nonstationary signals. Using joint diagonalization algorithm[7], we can find a joint diagonalizer $\mathbf{W}$ of $\{\mathbf{M}_x(t_k,\tau_j)\}, (k = 1,2,\ldots,K, j = 1,2,\ldots,J)$ such that

$$\mathbf{W}\mathbf{M}_x(t_k,\tau_j)\mathbf{W}^T = \Lambda_{k,j} \tag{22}$$

where $\{\Lambda_{k,j}\}$ is a set of diagonal matrices.

3. SIMULATIONS

Here we evaluated the performance of 4 different BSS algorithms: (1) fast ICA; (2) flexible ICA; (3) CII algorithm; (4) NSS-TD-JD. In the experimental comparison we focused on the following:

- robustness to sensor noise
- computational complexity (in terms of flops)
- performance with respect to the number of sources
- performance with respect to the conditioning of the mixing matrix.

The performance index (PI) is defined by

$$\begin{aligned}\mathrm{PI} \quad = \quad & \frac{1}{2n(n-1)}\sum_{i=1}^{n}\left\{\left(\sum_{k=1}^{n}\frac{|g_{ik}|^2}{\max_j |g_{ij}|^2} - 1\right)\right. \\ & \left. + \left(\sum_{k=1}^{n}\frac{|g_{ki}|^2}{\max_j |g_{ji}|^2} - 1\right)\right\},\end{aligned} \tag{23}$$

where g_{ij} is the (i,j)-element of the global system matrix $\mathbf{G} = \mathbf{W}\mathbf{A}$ and $\max_j g_{ij}$ represents the maximum value among the elements in the ith row vector of $\mathbf{G}$, $\max_j g_{ji}$ does the maximum value among the elements in the ith column vector of $\mathbf{G}$. The performance index defined in (23) tells us how far the global system matrix $\mathbf{G}$ is from a generalized permutation matrix.

In the first experiment, natural speech signals of four different persons were mixed using a randomly generated mixing matrix and zero-mean Gaussian noises were added to the sensor signals. The size of the source matrix is $4 \times 10,000$.

In the NSS-TD-JD algorithm, the observation data is divided into 50 non-overlapping frames ($K = 50$). In the fast ICA algorithm, the stopping criterion for convergence is $(1 - |\mathbf{w}^T(k)\mathbf{w}(k+1)|) < 1 \times 10^{-5}$. In the flexible ICA algorithm, the learning rate is 0.3 and the stopping criterion for convergence is $(1 - \mathrm{Avg}[|\mathbf{w}^T(k)\mathbf{w}(k+1)|]) < 1 \times 10^{-5}$ where Avg[.] is an average operator with respect to row vectors of $\mathbf{W}$.

For robust prewhitening, we used a time-delayed correlation with the time lag $\tau = 1$. With the choice of $\tau = 1$, the performance of NSS-TD-JD and CII algorithms were improved. However, the fast ICA and flexible ICA algorithms attained better performance with the choice of $\tau = 0$ than $\tau = 1$. With better choice of τ, those algorithms were tested and their PI vs. SNR are shown in Fig. 2. At each SNR, 50 different runs were carried out.

The flexible ICA and fast ICA algorithms show worse performance, compared to the CII and NSS-TD-JD algorithms. The poor performance of the flexible ICA and fast ICA was expected because they did not consider the additive noise effect. The CII algorithm is based on the 4th-order cross-cumulant, so it is expected to have some robustness to the additive Gaussian noise. In a wide range of SNR, the NSS-TD-JD algorithm shows the best performance. This means that the second order methods based on time-delayed correlation matrices are good techniques to reduce noise effects.

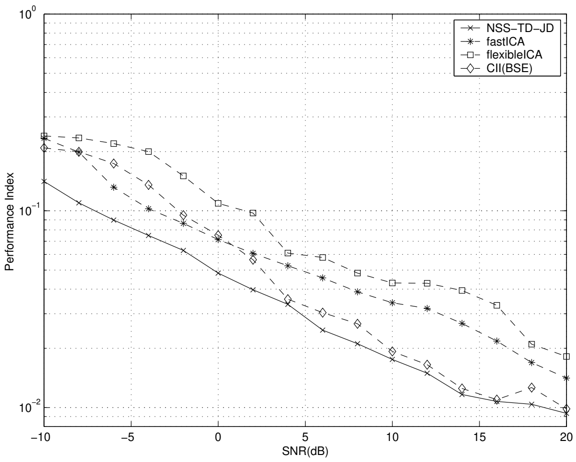
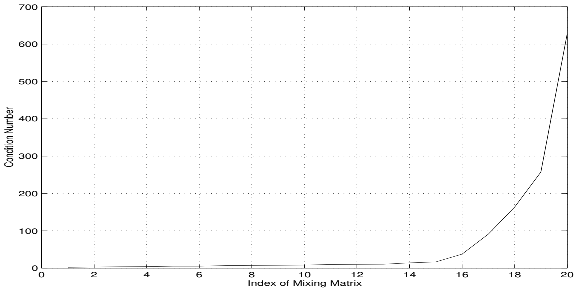
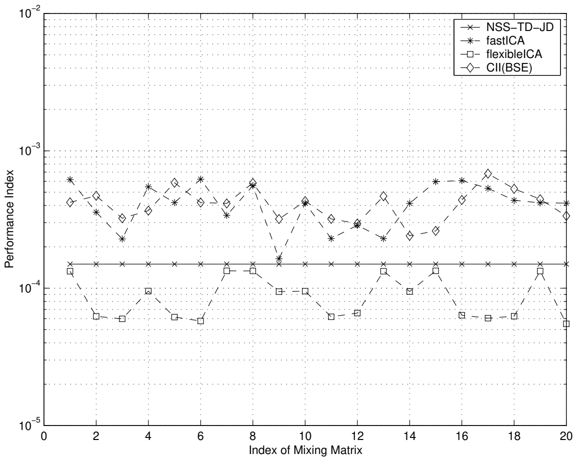


Figure 2: Performance Index vs. SNR in the speech separation experiment.

Next we evaluate how much the performance of algorithm is affected according to the condition number of the mixing matrix. For this investigation, we generated 20 mixing matrices ($\mathbf{A}$) randomly and sorted them according to the condition number. As shown in Fig. 3 (a), the condition number of mixing matrices ranges from 2.22 to 627.62. After simulations using these mixing matrices without additive noise, we drew the PI vs. the index of $\mathbf{A}$ in Fig. 3 (b). Since the flexible ICA algorithm belongs to the class of the natural gradient based ICA algorithms, it has equivariant property which guarantees uniform performance regardless of the mixing condition. In practical implementation, we observe some fluctations of performance in the flexible ICA algorithm, but it shows the best performance. The PI of the NSS-TD-JD algorithm does not vary according to the condition number of mixing matrix (at least in the range that we consider here). Although all the algorithms show reasonable performance in this simulation, it is not always true. More difficult environment will be considered next.



(a)



(b)

Figure 3: The relationship between condition number of mixing matrix and Performance Index in separation of speech signals without additive noise ((a) Condition number of mixing matrix, (b) Performance Index).

Next, we investigate PI and GFLOPS (giga FLOPS) versus the number of sources in BSS problems without noise. In this experiment, ten sets of n source signals were generated using the AR(auto-regressive) model

$$s_i(t) = \xi_i(t) - c_{i1}s_i(t-1) - c_{i2}s_i(t-2) \qquad (24)$$

where $\xi_i(t), c_{i1}$, and c_{i2} are randomly generated for each source, respectively. The distributions of c_{i1} and c_{i2} are uniform on $[0, 1]$ and $\xi_i(t)$ is uniform on $[-0.5, 0.5]$. Each source signal $s_i(t)$ was normalized to have unit variance. Furthermore, we excluded $s_i(t)$ with $|\kappa(s_i)| < 0.01$ since Gaussian-like sources can not be extracted by higher-order statistics based algorithms such as the flexible ICA, fast ICA, and CII algorithms. The size of source matrix was $n \times 5,000$. Exemplary source signals are shown in Fig. 4.

In this experiment, the frame size for NSS-TD-JD algorithm is 200 ($K = 25$). We used $\tau = 0$ for pre-whitening since we considered the noise-free data. The rest of parameters are the same as those in the first experiment.

Fig. 5 (a) and (b) show the PI vs. the number of sources and the GFLOPS vs. number of sources, respectively. These are the averaged results of 20 runs $\times$ 10 source

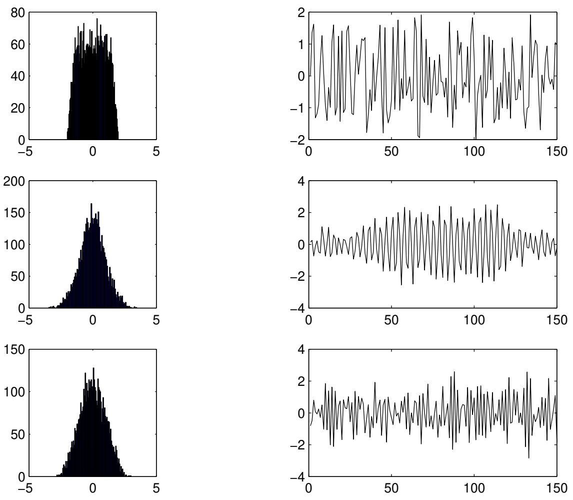


Figure 4: Examples of histograms and waveforms of source signals generated using the auto-regressive model.

sets. The flexible ICA algorithm shows better PI than the fast ICA algorithm, while the flexible ICA algorithm always requires larger GFLOPS than the fast ICA algorithm. The CII algorithm shows worse PI than the two ICA algorithms and its GFLOPS is between those of the two ICA algorithms or lower than that of the fast ICA for many sources. The NSS-TD-JD algorithm shows relatively stable characteristics of PI vs. number of sources and it required less GFLOPS when the number of sources is less than 40. However, the computation complexity of NSS-TD-JD increases very rapidly when the number of sources is greater than 40.

Furthermore, we investigate the PI and FLOPS of each algorithm when the mixing matrix is very ill-conditioned, the so-called "Hilbert matrix". The condition number of the Hilbert matrix is 1.55×10^4 for 4×4 matrix and 1×10^{19} for 64×64 matrix. Fig. 6 shows the simulation results using the Hilbert matrix. In this ill-conditioned situation, the fast ICA algorithm is better than the flexible ICA algorithm in viewpoints of PI (Fig. 6 (a)) and GFLOPS (Fig. 6 (b)). The CII algorithm shows worse PI than those of the two ICAs for No.(source)>20 and the GFLOPS is between those of the two ICAs. Still the NSS-TD-JD algorithm demonstrates the best PI with less FLOPS, although the FLOPS is much more than those of the fast ICA or CII algorithms for No.(sources)>40.

Finally, we propose one more benchmark problem in which sources are $s_i(k) = \sin(i\omega_o k)$ where $\omega_o = 200\pi$ and $i = 1, 2, \ldots, 10$. Such sources were successfully extracted using only the NSS-TD-JD approach.

4. CONCLUSIONS

In this paper, we compared several promising and efficient algorithms proposed for ICA/BSS problems, such as the fast ICA, flexible ICA, CII, and NSS-TD-JD algorithms, with respect to noise level, the number of sources, computational complexity, and ill condition of mixing matrices. Furthermore, in order to improve the robustness of algorithms in respect to noise, we proposed a robust prewhitening method using a modified covariance matrix with time delay.

In the speech signal experiment with additive Gaussian noise, the CII algorithm was better than the fast ICA and flexible ICA algorithms, since it used high-order cumulants which are insensitive to Gaussian noise. Also, the NSS-TD-JD algorithm used time-delayed correlation for noise robustness and this algorithm showed the best PI vs. SNR. Thus, the time-delayed correlation is a good strategy for reducing noise effects.

In particular, we improved robustness in respective of noise for the CII and NSS-TD-JD algorithms by the robust prewhitening with time delay by one sample. We can apply this method to improve noise robustness of other known algorithms.

For the artificially generated data, the NSS-TD-JD algorithm showed the best characteristics of PI with less FLOPS although FLOPS of NSS-TD-JD increased rapidly for the number of sources greater than 40. As a result, we can say that the NSS-TD-JD algorithm is adequate to separate sources with less computation complexity if the number of sources are less than 40. However, in general BSS problems with sources whose number is greater than 40, it is necessary to develop a separation algorithm with faster convergence and better performance.

The proposed benchmarks, such as the sinusoidal sources, source generated by a second order AR model, and the Hilbert mixing matrices, will be useful to compare and investigate characteristics of existing or in future proposed algorithms.

Acknowledgement: This work was supported by Brain Science Institute, RIKEN, Japan and Korea Ministry of Science & Technology under joint research program between Korea and Japan.

5. REFERENCES

[1] S. Amari, "Natural gradient works efficiently in learning," *Neural Computation*, pp. 251-276, Feb. 1998.

[2] A. Cichocki and R. Unbehauen, "Robust neural networks with on-line learning for blind identification and blind separation of sources," *IEEE Trans. Circuits and Systems-I: Fundamental Theory and Applications*, vol. 43, pp. 894-906, Feb. 1996.

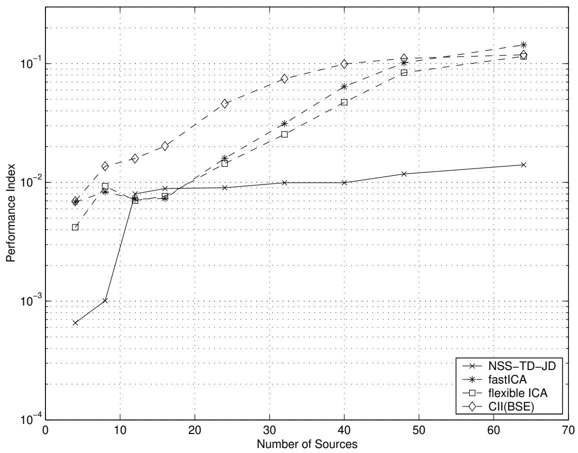
[3] S. Choi, A. Cichocki, and S. Amari, "Flexible independent component analysis," *J. VLSI Signal Processing-Systems for Signal, Image, and Video technology*, vol. 26, pp. 25-38, August 2000.

[4] S. Cruces, A. Cichocki, and L. Castedo, "Blind source extraction in Gaussian noise," *Proc. ICA'2000*, Helsinki, Finland, June 2000.

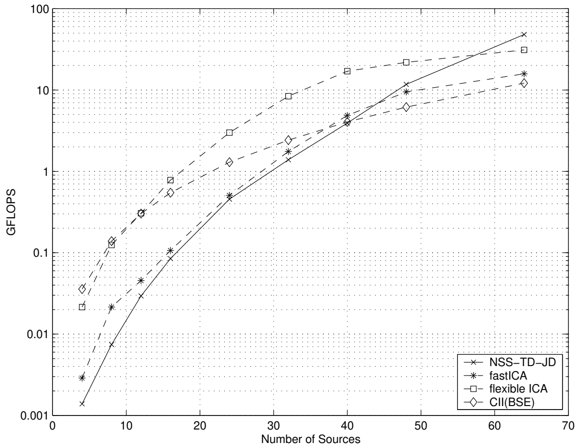
[5] S. Choi and A. Cichocki, "Blind separation of nonstationary sources in noisy mixtures," *IEE Electronics Letters*, vol. 36, no. 9, pp. 848-849, April 27, 2000.

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

[7] J. F. Cardoso and A. Souloumiac, "Jacobi angles for simultaneous diagonalization," *SIAM J. Mat. Anal. Appl.*, vol. 17, no. 1, pp. 161-164, Jan. 1996.

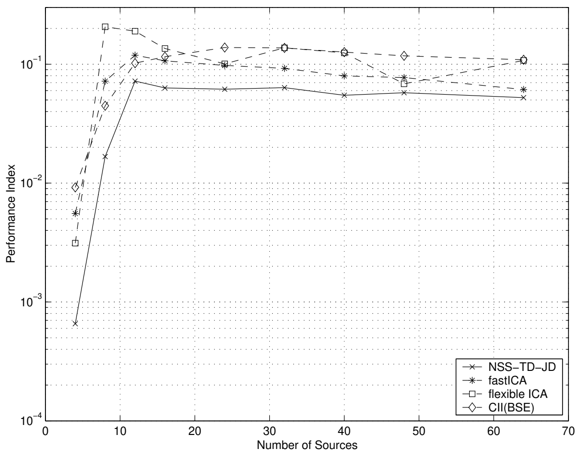


(a)

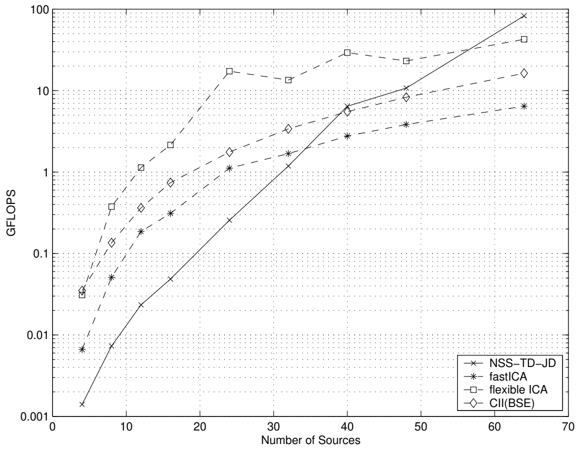


(b)

Figure 5: Simulation results with sources generated by the AR model (24): (a) PI vs. number of sources; (b) GFLOPS vs. number of sources.



(a)



(b)

Figure 6: Simulation results for Hilbert mixing matrix: (a) PI vs. number of sources; (b) GFLOPS vs. number of sources.