Sequential Source Separation for Mixed Kurtosis Sign Sources

K.Chinnasarn[1], *C.Lursinsap*[1] *and V.Palade*[2]

[1]*Advanced Virtual Intelligent Computing, Department of Mathematics, Chulalongkorn University, Bangkok, 10330, THAILAND.*
Email:krisana@buu.ac.th, chidchanok.l@chula.ac.th

[2]*Oxford University Computing Laboratory, Parks Road, Oxford, OX1 3QD, UK.*
Email:vasile.palade@comlab.ox.ac.uk

Abstract

This study concerns with applying a blind source extraction method and an online sub-block learning approach to the independent component analysis for mixed kurtosis sign sources. The study deals with the mutual information with natural gradient as a cost function. The prewhitenning step was used for decorrelating the existing correlation among the observed channels. It has been proved that the kurtosis sign of each source was recovered after the prewhitenning step. The blind source extraction algorithm was used within the separating equation, instead of the parallel blind source separation method. We separated the prewhitenned signals into the positive and negative kurtosis sign signals. Then, the online sub-block learning is applied for improving the learning performance. Experimental results presented in the paper prove the effectiveness of the method on different classical benchmark distributions.

1 Introduction

Recently, independent component analysis (ICA), or blind source separation (BSS), has become a high potential real world application problem. ICA problem concerns with the transformation and detransformation of the source signals **s** in the unknown environments. More precisely, the source distributions and the mixture environments are assumed to be totally unknown. The main objective of ICA problem is to recover the source signals from the observed signals which are collected from the receivers, such as microphones or sensors. Most practical approaches require some knowledge about the sources' distribution. In case more sources with different distributions are mixed together, then the evaluation method of the sources' distribution should be modified. In this paper, we present a simple and a powerful technique for classifying the class of the source distribution. We used the *Kurtosis* of the prewhitenned signals. Moreover, we will give a proof regarding the sub-block size, which we observed that it should be larger than the maximal frequency in time domain of the principal component. In addition, for learning efficiency, we selected the sequential blind source extraction method instead of the parallel blind source separation method.

The paper is organized as follows: Section 2 describes some background of a real-world ICA problem; Section 3 derives the mutual information learning with natural gradient and some modifications; Section 4 reviews the mixed kurtosis sign activation functions; Some experimental results are presented in section 5 and conclusions in section 6, respectively.

2 Problem Background

A very well-known practical example of the ICA application is the *cocktail party problem*. This problem assumes there are some people (sources s_i) talking simultaneously in the room, which is provided with some microphones for receiving what they are talking about. Herein, we assume that the number of the people is equal to the number of the microphones or sensors, $n = m$. Each microphone Mic_j gives you a signal, denoted as $x_j(t)$, where $1 \leq j \leq m$ and t is an index of time. Each of these recorded signals is a linear combination of the original signal s_i, $(1 \leq i \leq n)$ using the mixing matrix $\mathbf{H}$, as given below:

$$x_j(t) = \sum_{i=1}^{n} h_{ji} s_i(t) \tag{1}$$

where h_{ij}, $1 \leq i, j \leq n$ are the weighted sum parameters that depend on the distance in between the microphones and the speakers [7]. Commonly, the elements of the mixing matrix $\mathbf{H}$ and the probability density function of $s_i(t)$ are unknown in advance. The only basic assumption of *the cocktail party problem* is that all of the sources $s_i(t)$ are independent and identically distributed, (*iid*). The basic background of ICA were presented in [1] [4] [7]. The objective of an ICA problem is to transform the mixed signals, $\mathbf{x} = \mathbf{Hs}$, into the output signals, $\mathbf{y} = \mathbf{Wx}$, where the recovered signals $y_i(t)$ should be *iid* as well. In this paper, we used the mutual information with natural gradient (**MING** algorithms) which was proposed by Amari *et al.* [1] in 1996, for finding the demixing matrix $\mathbf{W}$.

3 Learning Algorithms

Let $p(\mathbf{y})$ be the joint probability density function of $\mathbf{y}$, and $q(\mathbf{y})$ be an estimated or the marginal probability density function of $\mathbf{y}$, in which all y_i are statistically independent, $q(\mathbf{y}) = \prod_{i=1}^{n} q_i(y_i)$. We will use the Kullback-Leibler divergence between the joint and the estimated probability density function of $\mathbf{y}$ as in [1] [6]

$$D_{pq} = \int_{-\infty}^{\infty} p(\mathbf{y}) \log \frac{p(\mathbf{y})}{q(\mathbf{y})} d\mathbf{y} \quad (2)$$

The Kullback-Leibler divergence always takes positive values, and becomes zero if $p(\mathbf{y})$ and $q(\mathbf{y})$ have the same distribution. It is invariant with respect to the invertible nonlinear transformations of variable y_i, including scaling and permutation [4] [6]. Amari [1] showed that Kullback-Liebler divergence $D(\mathbf{W})$ can be calculated from the average Mutual Information (MI) of y_i as follows:

$$D_{pq} = -h(\mathbf{y}) + \sum_{i=1}^{n} h_i(y_i) \quad (3)$$

where

$$\begin{array}{ll} h(\mathbf{y}) & = E[\log p(\mathbf{y})] \quad = -\int_{-\infty}^{\infty} p(\mathbf{y}) \log p(\mathbf{y}) d\mathbf{y} \\ h_i(y_i) & = E[\log q_i(y_i)] \quad = -\int_{-\infty}^{\infty} p(\mathbf{y}) \log q_i(y_i) d\mathbf{y} \end{array}$$

$h(\mathbf{y})$ is a differential entropy and $h(y_i)$ is a marginal entropy of the variable $\mathbf{y}$, respectively. From $\mathbf{y} = \mathbf{W}\mathbf{x}$, the differential entropy can be calculated as follows:

$$h(\mathbf{y}) = h(\mathbf{W}\mathbf{x}) = h(\mathbf{x}) + \log|\det(\mathbf{W})| \quad (4)$$

Applying (4) to (3), we get

$$D_{pq}(\mathbf{W}) \approx -h(\mathbf{x}) - \log|\det(\mathbf{W})| + \sum_{i=1}^{n} h_i(y_i) \quad (5)$$

To find $\mathbf{W}$ that minimizes $D_{pq}(\mathbf{W})$, we differentiate $D_{pq}(\mathbf{W})$ with respect to $\mathbf{W}$. In 1996 Amari *et al.* [1] reported the *natural gradient* learning for ICA problem, where $\mathbf{W}^T\mathbf{W}$ is an an optimal rescaling of the entropy gradient. Hence, the gradient direction of $\frac{\partial D_{pq}(\mathbf{W})}{\partial \mathbf{W}}$ should be:

$$\frac{\partial D_{pq}(\mathbf{W})}{\partial \mathbf{W}} \mathbf{W}^T\mathbf{W} = [-\mathbf{I} + \frac{\dot{q}_i(y_i)}{q_i(y_i)} \mathbf{y}^T]\mathbf{W} \quad (6)$$

The learning equation is given by an ordinary *steepest gradient descent* technique: $\Delta\mathbf{W}(t) = -\eta(t)\frac{\partial D(\mathbf{W})}{\partial \mathbf{W}}$, where η is the learning rate which depends on the learning time t. For improving the convergence speed, we added the momentum term $\Delta\mathbf{M}$ in the equation (6) [2]. Hence, $\mathbf{W}$ at time $t+1$ is adjusted by the following constructive step:

$$\mathbf{W}(t+1) = \mathbf{W}(t) + \Delta\mathbf{W}(t) + \Delta\mathbf{M}(t) \quad (7)$$

where

$$\begin{array}{ll} \Delta\mathbf{W}(t) & = \eta(t)[\mathbf{I} - \phi(\mathbf{y})\mathbf{y}(t)^T]\mathbf{W}(t) \\ \Delta\mathbf{M}(t) & = \beta\Delta\mathbf{W}(t) \\ \phi(\mathbf{y}) & = \frac{\dot{q}_i(y_i)}{q_i(y_i)} = [\frac{\dot{q}_1(y_1)}{q_1(y_1)}, \ldots, \frac{\dot{q}_n(y_n)}{q_n(y_n)}]^T \end{array}$$

The function $\phi(\mathbf{y})$ depends on the source probability density function, and β is the momentum rate.

3.1 Analysis on the Online Learning Sub-block Size

In the ICA problem, it is known that the sources s_i are multidimensional signals which are functionally expressed in terms of a vector as below (e.g. $1 \le i \le 3$):

$$s_i(t) = \left\{ \begin{array}{ll} s_1(t) & = A_1 e^{jk_1\Omega_1 t} \\ s_2(t) & = A_2 e^{jk_2\Omega_2 t} \\ s_3(t) & = A_3 e^{jk_3\Omega_3 t} \end{array} \right. \quad k = 0, \pm 1, \pm 2 \quad (8)$$

where A_i is an amplitude, $\Omega_i = 2\pi F_{0_i}$, and $k_i F_0$ is the fundamental frequency. The sources $s_i(t)$ are mixed by the unknown full-rank environment $\mathbf{H}$. The mixture equation of ICA problem can be derived again as follows:

$$x_j(t) = \sum_{i,j=1}^{3} h_{ij} A_j e^{jk_j\Omega_j t} \quad (9)$$

Without loss of generality, we assume that $[h_{11} = h_{12} = h_{13}]$, $[h_{21} = h_{22} = h_{23}]$, and $[h_{31} = h_{32} = h_{33}]$. We get:

$$x_j(t) = 3h_{1j} e^{jt} (\sum_{j=1}^{3} A_j e^{k_j\Omega_j}) \quad (10)$$

Equation (8) showed that the sources $s_i(t)$ have different fundamental frequency $k_i F_0$. Let's assume that s_1 is the principal component. The fundamental frequency of $s_1(t)$ was changed to $\sum_{j=1}^{3} k_j\Omega_j$ after the

transformation in equation (10). In order to recover the source signals from the observed signals, the data block size must be at least equal to the fundamental frequency in time domain. And the same in our simulations on various sources with different fundamental frequencies, we need some observed data points at least equal to the value of the maximal frequency F_{max}.

In this subsection, we describe an online sub-block ICA learning algorithm. We used an unsupervised multi-layer feed forward neural network for de-mixing the non-gaussian channels. Our network learning methods is a combination of online and batch learning techniques. Unknown signals $x_i(k_0 : k_0 + k)$ are fed into the input layer, where k_0 is the start index of the sub-block, $k \geq F_{max}$ is the length of the sub-block, and F_{max} is the value of the maximal frequency in the time domain of the principal component of the source s_i. Output signals $y_i(k_0 : k_0 + k)$ are produced by $y_i(k_0 : k_0 + k) = \mathbf{W}x_i(k_0 : k_0 + k)$, where $\mathbf{W}$ is called the de-mixing matrix. If the output channels $y_i(k_0 : k_0+k)$ depend on each other, the natural gradient descent in equation (7) still updates the de-mixing matrix $\mathbf{W}$. Then repeat to produce the output signals $y_i(k_0 : k_0 + k)$ until they become independent. The increase of the convergence speed of the online sub-block method is proved by the following theorem.

Theorem 1 *ICA online sub-block learning is converging faster than batch learning.*

Proof Considering K is the total time index of the signal and k is the time index number for each sub-block, where $k < K$. The learning equation (7) can be rewritten as follow:

$$\mathbf{W}_{t+1} = \mathbf{W}_t + \eta[\mathbf{I} - \phi(\mathbf{y})\mathbf{y}^T]\mathbf{W}_t + \beta\Delta\mathbf{W}_t \qquad (11)$$

The computational complexity of equation (11) depends on the correlation $\phi(\mathbf{y})\mathbf{y}^T$, where $\mathbf{y}^T$ is a transpose matrix of $\mathbf{y}$. For the batch learning with time index K, the complexity of (11) is of $O(K^3)$

On the other hand for the online sub-block learning, we have $\frac{K}{k}$ sub-blocks. The computational complexity of equation (11) is of $\frac{K}{k}O(k^3) = O(K.k^2)$. It is obvious that $O(K.k^2) < O(K^3)$, where $k < K$. Hence, the ICA online sub-block learning is faster than the batch learning.

3.2 Sequential Blind Separation

In the ICA problem, at most one gaussian channel is allowed because the transformation of two gaussianities are also gaussian in another variable [7]. The non-gaussian distribution can be classified into super-gaussian and sub-gaussian distributions. A super-gaussian signal has a sharp peak and a large tail probability density function (pdf). On the other hand, sub-gaussian signals have a flat pdf. As we described in the previous section, the nonlinear activation function $\phi_i(\mathbf{y})$ in equation (7) is determined by the sources' distribution. The graphical representation of the gaussian, super-gaussian, and sub-gaussian distributions are illustrated in figure 1. Typically, the source distribution is measured by the *Kurtosis* [4] [7] [9].

$$Kurtosis(\mathbf{s}) = \frac{E[\mathbf{s}^4]}{(E[\mathbf{s}^2])^2} - 3 \qquad (12)$$

where $Kurtosis(\mathbf{s})$ values are negative, zero, and positive for the sub-gaussianity, gaussianity, and super-gaussianity, respectively.

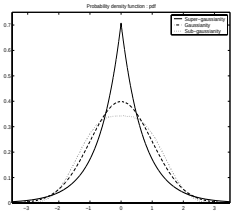


Figure 1: Probability density function of gaussian and non-gaussian distributions.

As described before in this contribution, we focus on the mixing and the de-mixing process of the unknown source signals $s_i, 1 \leq i \leq m$, which may have distinguished mathematical or physical model [4]. The source distribution $p(\mathbf{s})$ has been transformed to $\hat{p}(\mathbf{s})$ that depends on the mixing matrix $\mathbf{H}$. In other words, the kurtosis sign of the source might be changed. In order to recover the kurtosis sign of the source, we require the prewhitenning step on the observed signal $\mathbf{x}$. The prewhitenning step will decorrelate the existing correlation between the observed channels. In other words, $E[\tilde{\mathbf{x}}\tilde{\mathbf{x}}^T] = \mathbf{I}$ where $\tilde{\mathbf{x}} = diag(\frac{1}{\sqrt{d_1}}, \ldots, \frac{1}{\sqrt{d_m}})\mathbf{V}^T\mathbf{x}$ and the d_i and the $\mathbf{V}^T$ are the eigenvalue and the eigenvector of the covariance of the observed signal $\mathbf{x}$ [4]. After the projection of the observed signals $\mathbf{x}$, we obtain:

$$E[\mathbf{x}\tilde{\mathbf{x}}^T] \cong E[\mathbf{s}\mathbf{s}^T] = \mathbf{I} \qquad (13)$$

We substitute $\mathbf{s}$ in (12) with $\tilde{\mathbf{x}}$ in (13). Then we obtain:

$$Kurtosis(\tilde{x}_i) = \frac{E[s_i^4]}{(E[s_i^2])^2} - 3. \quad (14)$$

It means that, each component $\tilde{x}_i$ and $\tilde{x}_j$ are decorrelated or mutually independent after the prewhitenning transformation. We classify the prewhitenned signals into 2 sub-groups which are the positive and the negative kurtosis sign signals. We started the deseparating procedure with the positive kurtosis sign signals. Then, the negative kurtosis sign signals were consequently fed into the deseparating procedure.

4 Activation Functions for Mixed Kurtosis Sign Sources

As very well known, the source distribution functions have their own characteristics, depending on their nature illustrated in figure 1. Hence, the sources $\mathbf{s}$ might be recovered from the observed channels $\mathbf{x}$ without any prior information about the sources themselves, if we can determine an appropriate nonlinear activation function for each channel of the prewhitenned signals.

4.1 Existing Methods

In [5], Douglas *et al.* presented two nonlinear activation function for switching between sub-gassianity and super-gaussianity, as follow:

$$\phi_{sub}(\mathbf{y}) = \mathbf{y}^3 \quad \text{and} \quad \phi_{sup}(\mathbf{y}) = \tanh(10\mathbf{y}) \quad (15)$$

In [9], Lee *et al.* presented an extended Information Maximization algorithm, or *Infomax*. They suggested $\phi(\mathbf{y}) = \tanh(\mathbf{y}) - \mathbf{y}$ for sub-gaussian distribution and $\phi(\mathbf{y}) = \tanh(\mathbf{y}) + \mathbf{y}$ for super-gaussian distribution, respectively. A simple learning equation for separating the mixing of non-gaussian distribution is expressed as follow:

$$\mathbf{W}_{t+1} = \mathbf{W}_t + \eta(\mathbf{I} - \mathbf{K}\tanh(\mathbf{y})\mathbf{y}^T - \mathbf{y}\mathbf{y}^T)\mathbf{W}_t \quad (16)$$

where $K = diag[k_1, \ldots, k_m]^T$ is a diagonal matrix of signs. If we know the sources' distribution, then we can assign negative values for k_i if the sources are sub-gaussian distributed, and positive values if the sources are super-gaussian distributed, respectively. If the sources' distribution is unknown, the switching between the sub-gaussian and the super-gaussian learning rule is given by the following:

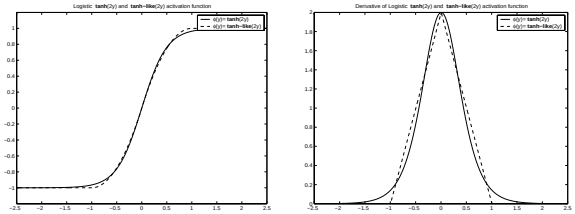
$$k_i = sign(E[\text{sech}^2]E[y_i^2] - E[\tanh(y_i)y_i]) \quad (17)$$

4.2 Low computational methods

In our previous paper [3], we presented the low computational activation function for demixing of super-gaussian and sub-gaussian distributions. Both functions are of 2^{nd} order. First, we presented an activation function for demixing a super-gaussian distribution. This function was derived from the **KTLF** (**K**wan **T**anh-**L**ike activation **F**unction) presented by Kwan in [8]. The function is an approximation of $\tanh(2\mathbf{y})$ function. He divided the approximation curve into 3 regions, which are the upper bound $\mathbf{y}(t) \geq L$, the nonlinear logistic tanh-like area $-L < \mathbf{y}(t) < L$, and the lower bound $\mathbf{y}(t) \leq -L$. The **KTLF** function is very similar to the $\tanh(2\mathbf{y})$ function (see figure 2(**a**)). The term $\frac{\alpha}{2}$ is needed for controlling $\mathbf{y}$ and we also suggest $L = 1$. The modified equation can be rewritten as follow:

$$\phi(\mathbf{y}) = \begin{cases} 1, & (\mathbf{y} \geq 1) \\ \acute{\mathbf{y}}(2 - \acute{\mathbf{y}}), & (0 \leq \mathbf{y} < 1) \\ \acute{\mathbf{y}}(2 + \acute{\mathbf{y}}), & (-1 < \mathbf{y} < 0) \\ -1, & (\mathbf{y} \leq -1) \end{cases} \quad (18)$$

where α is an upper-peak of the derivative of the activation function and $\acute{\mathbf{y}} = \frac{\alpha \mathbf{y}}{2}$. Figure 2 shows $\tanh(\alpha\mathbf{y})$, its approximation (the dash line) and their derivatives.

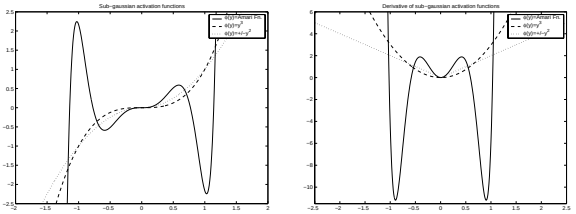


(**a**) (**b**)

Figure 2: (**a**) the $\phi(\mathbf{y}) = \tanh(2\mathbf{y})$ activation function, its approximation from equation (18) and (**b**) their derivatives

Given the graphical representation of the sub-gaussian activation functions illustrated in figure 3, it can be seen that the sub-gaussian activation functions can be separated into 2 regions: the positive and the negative regions. For demixing the sub-gaussian distribution, we proposed the bisection paraboloid function given in equation (19) which is a good approximation for the previously reported functions in the literature.

$$\phi(\mathbf{y}) = \begin{cases} +\mathbf{y}^2, & (\mathbf{y} \geq 0) \\ -\mathbf{y}^2, & (\mathbf{y} < 0) \end{cases} \quad (19)$$



(**a**) (**b**)

Figure 3: (**a**) Graphical representation of an activation function of 11^{th}, 3^{rd}, and 2^{nd} order activation function and (**b**) their derivatives.

5 Experimental Results

Some simulations have been made on both super-gaussian and sub-gaussian signals, which contained 46,500 data points for each channel. The super-guassian data sets consist of 3 sound sources taken from http://sound.media.mit.edu/ica-bench/ and 3 sound sources from http://bulsai.kaist.ac.kr/~jangbal/. For sub-gaussianity, we simulated our algorithm using the following three-synthesized signals:

1. $\mathbf{s}_1(t) = 0.1\sin(400t)\cos(30t)$
2. $\mathbf{s}_2(t) = 0.01\text{sign}[\sin(500t + 9\cos(40t))]$
3. $\mathbf{s}_3(t) =$ uniform noise in range [-0.05,0.05]

A mixing matrix $\mathbf{H}$ and an initial de-mixing matrix $\mathbf{W}$ are randomly generated. As presented in [2], the convergent criterion should be set so that the distance of Kullback-Leibler divergence is less than 0.00001 ($\Delta\mathbf{KL} \leq 0.00001$). We initialized the learning rate value to $\eta = 0.05$ and the momentum rate value the $\beta = 0.01\eta$. At each learning iteration, the learning was decreased by 1.005 ($\eta = \frac{\eta}{1.005}$). Some of the experimental results were presented in our previous papers [2] [3]. For improving the learning performance, we assigned the relationship between the online sub-blocks. The final de-mixing matrix $\mathbf{W}$ of the sub-block j is set to the initial de-mixing matrix of $j+1$ sub-block. The weight inheritance will maintain the output channel of unknown mixture environments. We consider that all the activation functions can recover the sources with different kurtosis value as it can be seen in table 1. The mean square errors are illustrated in table 2. In the simulation on super-gaussian distribution, we approximate our activation function from equation (18) by $\phi(y) = \tanh(10\mathbf{y})$, in order to compare the simulation results with the Douglas algorithm [5]. Figure 4 displays the original signals and the recovered signals for the sub-gaussian and the super-gaussian distributions. The original signals are displayed on the left column whereas the recovered signals are displayed on the right column. For examples, the super-gaussian source s_1 can be recovered at the output channel number y_5. As the same with the sub-gaussian source s_8 can be recovered at the output channel number y_8.

Table 1: The Kurtosis of the 9 source signals and the Kurtosis of the recovered signals via Douglas, Infomax and our low complexity functions **LFICA** from section [3].

Source types	Original kurtosis	Recovered Kurtosis		
		Douglas	Infomax	**LFICA**
$Super_{1st}$	$3.38e+0$	$1.08e+1$	$3.32e+0$	$2.69e+0$
$Super_{2nd}$	$5.84e-1$	$2.00e+0$	$6.05e-1$	$3.32e-1$
$Super_{3rd}$	$1.16e+0$	$1.48e+0$	$7.79e-1$	$1.06e+0$
$Super_{4th}$	$3.91e+1$	$1.11e+2$	$2.49e+1$	$6.56e+1$
$Super_{5th}$	$3.06e+1$	$2.24e+2$	$3.48e+1$	$1.67e+2$
$Super_{6th}$	$1.16e+0$	$2.46e+0$	$8.96e-1$	$8.28e-1$
Sub_{1st}	$-4.69e-6$	$-8.64e-1$	$-8.04e-1$	$-4.88e-1$
Sub_{2nd}	$-2.00e-8$	$-4.82e+0$	$-3.38e+0$	$-2.03e+0$
Sub_{3rd}	$-5.26e-8$	$-1.63e+0$	$-1.40e+0$	$-8.64e-1$

Table 2: The mean square error of the recovered signals via Douglas, Infomax and our low complexity functions **LFICA**.

Source types	Mean square error		
	Douglas	Infomax	**LFICA**
$Super_{1st}$	$6.17e+0$	$1.67e-2$	$5.24e+0$
$Super_{2nd}$	$6.88e+0$	$2.83e-1$	$5.95e+0$
$Super_{3rd}$	$3.73e+0$	$7.65e-2$	$3.83e+0$
$Super_{4th}$	$4.82e+0$	$6.49e-1$	$3.10e+0$
$Super_{5th}$	$1.05e-1$	$3.61e+0$	$5.26e-2$
$Super_{6th}$	$4.01e+0$	$5.53e-2$	$7.34e-2$
Sub_{1st}	$1.20e+0$	$1.11e+0$	$7.07e-1$
Sub_{2nd}	$1.58e+0$	$1.11e+0$	$1.03e+0$
Sub_{3rd}	$1.15e+0$	$1.28e+0$	$8.76e-1$

As an algorithmic performance measurement, we used the performance correlation index, derived from the performance index proposed by Amari *et al.* in [1]. In practice, the performance index can be replaced with the performance correlation index.

$$E = \sum_{i=1}^{N}(\sum_{j=1}^{N}\frac{|c_{ij}|}{max_k\,|c_{ik}|} - 1) + \sum_{j=1}^{N}(\sum_{i=1}^{N}\frac{|c_{ij}|}{max_k\,|c_{kj}|} - 1) \quad (20)$$

The matrix $C = \phi(\mathbf{y})\mathbf{y}^T$ is close to the identity matrix when the signal y_i and y_j are mutually uncorrelated or linearly independent. Figure 5 illustrates the performance correlation index on a logarithmic scale, during the learning process, using our proposed activation functions in subsection 4.2, and the typical activation functions for the non-gaussian mixtures from subsection 4.1. Figure 5(**a**) corresponds to the mixture of super-gaussian signals. The curve of tanh(.) was matched with its approximation function, and it can be seen that they converge over the

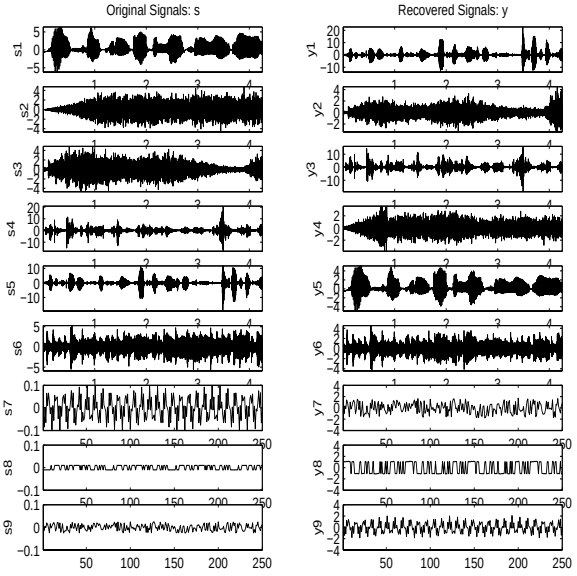
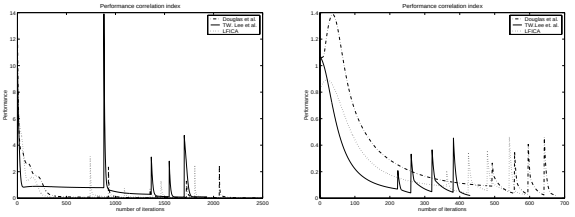


Figure 4: The mixing and demixing of sub-gaussian and super-gaussian distributions.

saturated region with the same speed. Figure 5(**b**) corresponds to the mixture of sub-gaussian distribution. The proposed activation function $\phi(\mathbf{y}) = \pm\mathbf{y}^2$ converges faster than $\phi(\mathbf{y}) = \mathbf{y}^3$ in the beginning, and slows down when the outputs y_i are close to saturated region. The derivative of the function $\phi(\mathbf{y}) = \pm\mathbf{y}^2$ is slower than the derivative of the function $\phi(\mathbf{y}) = \mathbf{y}^3$ when $\mathbf{y} \geq \pm 0.667$, see the slope of each function in figure 3.



(**a**) (**b**)

Figure 5: Performance correlation index for the sequential source separation of (**a**) super-gaussian mixtures, (**b**) sub-gaussian mixtures

6 Conclusions

A sequential blind source separation, or blind source extraction, and an online sub-block learning method for an independent component analysis for the non-gaussian mixtures using a two layered neural network have been presented in this paper. The sequential blind source separation converges faster to the local minima than the parallel blind source separation, when we use the Kullback-Liebler divergence based on Gram-Chalier expansion as presented in [1] and [6]. It has been proved that the proposed learning methods are efficient algorithms for non-gaussian mixtures.

Acknowledgement

This work is fully supported by a scholarship from the Ministry of University Affairs and the National Electronics and Computer Technology Center of Thailand.

References

[1] S.-I.Amari, A.Cichocki, and H.H.Yang. A New Learning Algorithm for Blind Signal Separation, *MIT Press*, pp.757-763, 1996.

[2] K.Chinnasarn and C.Lursinsap. Effects of Learning Parameters on Independent Component Analysis Learning Procedure, *Proceedings of the* 2^{nd} *International Conference on Intelligent Technologies*, pp. 312-316, 2001.

[3] K.Chinnasarn, C.Lursinsap, and V.Palade. Low Complexity function for Stationary Independent Component Mixtures, accepted to be published in *the Proceedings of KES2003*.

[4] A.Cichocki and S.-I.Amari. *Adaptive Blind Signal and Image Processing: Learning Algorithms and Applications*, John Wiley & Sons, Ltd., 2002.

[5] S.C.Douglas, T.-P.Chen, and A.Cichocki. Multichannel blind separation and deconvolution of sources with arbitary distributions, in, *Proc NNSP*, Amelia Island, FL, Sep. 1997, pp.436-445.

[6] S.Haykin. *Neural Network a Comprehensive foundation*. 2nd, Prentice Hall,1999.

[7] A.Hyvarinen and E.Oja. Independent Component analysis: algorithms and applications, *Neural Networks*. vol. 13 pp. 411-430, 2000.

[8] H.K.Kwan. Simple Sigmoid-like activation function suitable for digital hardware implementation, *Electronics Letter* Vol.28, no.15, pp.1379-1380, 1992.

[9] T.-W.Lee, M.Girolami and T.J.Sejnowski. Independent Component Analysis Using an Extended Informax Algorithm for Mixed Sub-Gaussian and Super-Gaussian Sources *Neural Computation*, Vol. 11, No. 2, pp. 409-433, 1999.