Self-Adaptive Blind Source Separation Based on Activation Functions Adaptation

Liqing Zhang, *Member, IEEE*, Andrzej Cichocki, and Shinichi Amari, *Fellow, IEEE*

***Abstract*—Independent component analysis is to extract independent signals from their linear mixtures without assuming prior knowledge of their mixing coefficients. As we know, a number of factors are likely to affect separation results in practical applications, such as the number of active sources, the distribution of source signals, and noise.The purpose of this paper to develop a general framework of blind separation from a practical point of view with special emphasis on the activation function adaptation. First, we propose the exponential generative model for probability density functions. A method of constructing an exponential generative model from the activation functions is discussed. Then, a learning algorithm is derived to update the parameters in the exponential generative model. The learning algorithm for the activation function adaptation is consistent with the one for training the demixing model. Stability analysis of the learning algorithm for the activation function is also discussed. Both theoretical analysis and simulations show that the proposed approach is universally convergent regardless of the distributions of sources. Finally, computer simulations are given to demonstrate the effectiveness and validity of the approach.**

***Index Terms*—Activation function, blind source separation, exponential family, independent component analysis.**

I. Introduction

BLIND source separation or independent component analysis has attracted considerable attention in the signal-processing and neural-network society, since it not only introduces a novel paradigm for signal processing, but also has rapidly growing applications in various fields, such as telecommunication systems, speech processing, image enhancement, and biomedical signal processing.

Several neural-networks and statistical signal-processing methods [2], [7], [11], [13], [16], [17], [21], [23], [26], [27] have been developed for blind signal separation. There are a number of factors that are likely to affect the separation performance in applications, such as the number of active sources, the distribution of source signals, time-variable mixtures, and noise.

The stability of learning algorithms [4], [13] is critical to successful separation of source signals from measurements. The stability conditions depend on the statistics of source signals. There are a number of ways to deal with the stability problem. Assuming that no prior information is available about the source distribution, one can estimate the statistics such as kurtosis online, so as to determine the characteristics of source signals and the activation functions. Amari *et al.* [4] presented a universal convergence approach that has equal convergence rate for different source signals. Another idea proposed by Pham [24] is to expand the activation functions in a linear combination of known functions and their coefficients are determined by the training data. The main problem of these known approaches is that it is inevitable to estimate some statistics of the source signals and online estimators may not be accurate enough to approximate the true statistics by using the output signals of the demixing model. In particular, when the source signals consist of both super- and sub-Gaussian signals, it is not easy to estimate the signs of kurtosis of the source signals using the sensor signals.

Some other statistical models, such as the generalized Gaussian model [12], [14], [20], the Gaussian mixture model [10], [22], and the Pearson system [19], are employed to estimate the distributions of source signals. The maximum likelihood method is applied to estimate the posterior distribution. Generally speaking, the estimation of distributions based on the maximum likelihood is computationally demanding and convergence is slow. Also, the above works did not cover convergence and stability analysis of the learning algorithm for the parameters in statistical generative models.

It is the purpose of this paper to develop a learning strategy to adapt the activation functions online so as to ensure the stability of the learning algorithm for the demixing model. Different from the previous works on the distribution estimation for the source signals, this paper attempts to avoid directly estimating the distributions of the sources, but to adapt the activation functions for the source signals online. The adaptation of activation functions has two purposes: to modify the activation functions such that the true solution becomes the stable equilibrium of learning system and to classify the source signals or to estimate the sparseness of source signals. The difference between the distribution estimation and activation function adaptation is that the activation function adaptation attempts to find an adequate activation function, which might not be the score function defined by the true distribution. Thus, it needs only a very few parameters in the activation function model. This simplification makes it easy to estimate the parameters in generative models and to reduce the computing cost. In order to accelerate the convergence rate of the learning algorithm for estimating activation functions, the natural gradient algorithm is also applied to update the parameters in the generative model. We will show

Manuscript received October 26, 2001; revised February 5, 2003. The work of L. Zhang was supported by the National Natural Science Foundation of China under Grant 60375015.

L. Zhang is with the Department of Computer Science and Engineering, Shanghai Jiaotong University, Shanghai 200030, China.

A. Cichocki and S. Amari are with the Brain-Style Information Systems Research Group, RIKEN Brain Science Institute, Saitama 351-0198, Japan.

Digital Object Identifier 10.1109/TNN.2004.824420

that the natural gradient algorithm does help to increase the convergence rate of the algorithm for estimating the activation functions. We further elaborate the generalized Gaussian distribution and study the convergence and stability of the learning process for updating the activation functions. Computer simulations are given to demonstrate the validity and efficiency of the adaptive algorithm.

There are some advantages to using the exponential generative model to estimate activation functions. It is easy to reveal the relation between the distribution and activation functions. Also, we can easily construct a linear connection with the activation functions for the exponential generative model if we want to separate signals with specific distributions. Another important property is that the method is consistent, i.e., both the updating rules for the demixing model and for the free parameters in the generative model make the cost function decrease to its minimum, if the learning rate is sufficiently small.

II. Formulation of the Problem

Assume that source signals are stationary zero-mean processes and are mutually statistically independent. Let $\mathbf{s}(k) = (s_1(k), \ldots, s_n(k))^T$ be the vector of unknown independent sources and $\mathbf{x}(k) = (x_1(k), \ldots, x_m(k))^T$ be a sensor vector, which is a linear instantaneous mixture of sources by

$$\mathbf{x}(k) = \mathbf{A}\mathbf{s}(k) + \boldsymbol{v}(k), \quad k = 1, 2, \ldots \tag{1}$$

where $\mathbf{A} \in \mathbf{R}^{m \times n}$ is an unknown mixing matrix of full rank and $\boldsymbol{v}(k)$ is the vector of Gaussian noises. The blind separation problem is to recover original source signals from observations $\mathbf{x}(k)$ without prior knowledge on the source signals and mixing matrix, unless the assumption of mutual independence of source signals. The demixing model used here is a linear transformation of the form

$$\mathbf{y}(k) = \mathbf{W}\mathbf{x}(k) \tag{2}$$

where $\mathbf{y}(k) = (y_1(k) \ldots, y_m(k))^T$, $\mathbf{W} \in \mathbf{R}^{m \times m}$ is a demixing matrix to be determined during training. We assume that $m \geq n$, i.e., the number of sensor signals is larger than the number of source signals. The general solution to the blind separation problem is to find a matrix $\mathbf{W}$ such that

$$\mathbf{W}\mathbf{A} = \mathbf{\Lambda}\mathbf{P} \tag{3}$$

where $\mathbf{\Lambda} = \begin{bmatrix} \mathbf{\Lambda}_0 \\ \mathbf{0} \end{bmatrix}$, $\mathbf{\Lambda}_0$ is a diagonal matrix and $\mathbf{P}$ is a permutation. In the case $m > n$, we train the demixing model $\mathbf{W}$ such that n components are designed to recover n source signals and the rest correspond to the zeros or noise.

The purpose of blind source separation is to adapt the demixing model such that its output signals are mutually independent. There exist a number of unknowns, such as the number of active sources and the probability density functions (pdfs) in the framework of blind source separation. The traditional approach is to estimate the number of active sources before training the demixing model, which may fail if the sensor signals are very noisy or the source signals are very weak. Different from the previous works on blind separation, we do not suggest to estimate the number of the active sources before training the demixing matrix.

We emphasize here that, in this framework, the sources of interest are distinguished from noise after training the demixing model. The discrimination between the sources and noise depends on the distribution and temporal structure of the separated signals, as well as some other knowledge of source signals. If a separated signal is sparsely distributed and has temporal structures, we consider it to be a source of interest.

Generally speaking, estimating the pdfs is computationally demanding and its convergence is usually very slow by using the ordinary gradient-descent method. We surmount the difficulty in two ways. First, we suggest the adaptation of the activation functions, instead of directly estimating the pdfs. As a result, we need only a very few parameters for the model of activation functions. Second, we use the natural gradient to train the parameters in the family of the activation functions to accelerate the convergence rate. Both theoretical analysis and computer simulations show that the proposed approach has a significant improvement in learning performance.

III. Learning Algorithm

Assume that $q_i(y_i, \boldsymbol{\theta}_i)$ is a model for the marginal pdf of y_i, $(i = 1, \ldots, m)$ parameterized by $\boldsymbol{\theta}_i$. Various approaches, such as entropy maximization and minimization of mutual information, lead to the cost function

$$l(\mathbf{y}, \boldsymbol{\theta}, \mathbf{W}) = -\log(|\det(\mathbf{W})|) - \sum_{i=1}^{n} \log q_i(y_i, \boldsymbol{\theta}_i) \tag{4}$$

where $\boldsymbol{\theta}_i$ is determined adaptively during training.

The estimation of the demixing model $\mathbf{W}$ can be formulated into the framework of the semiparametric statistical model [3]. In blind separation, the demixing matrix is considered as the parameter of interest and the pdfs of source signals are considered as the nuisance parameter, respectively. The semiparametric approach suggests the use of the estimating function to estimate the parameter $\mathbf{W}$. The estimating function for blind source separation [3] can be expressed by $\mathbf{F}(\mathbf{y}, \mathbf{W})$, with entries

$$f_{ij} = -\delta_{ij}\lambda_{ii} + \alpha_1 y_i y_j + \alpha_2 \varphi_i(y_i) y_j - \alpha_3 y_i \varphi_j(y_j) \tag{5}$$

where δ_{ij} is the Kronecker delta; λ_{ii}, α_1, α_2, and α_3 are parameters; and φ_i is a nonlinear activation function, depending on the distribution of the source signal $s_i(k)$. The best activation function is the score function defined by

$$\varphi_i(y_i) = -\frac{d \log p_i(y_i)}{dy_i} \tag{6}$$

where $p_i(.)$ is the true pdf of source s_i, which is considered to be the nuisance parameter in the ICA **AU: PLEASE SPELL OUT "ICA" —ED.** model. It is not necessary to precisely estimate the pdf in this semiparametric model. However, adequate activation functions will help to improve the learning performance for the demixing model. The online learning algorithm based on the estimating function can be described as

$$\Delta\mathbf{W}(k) = -\eta(k)\mathbf{F}(\mathbf{y}(k), \mathbf{W}(k))\mathbf{W}(k). \tag{7}$$

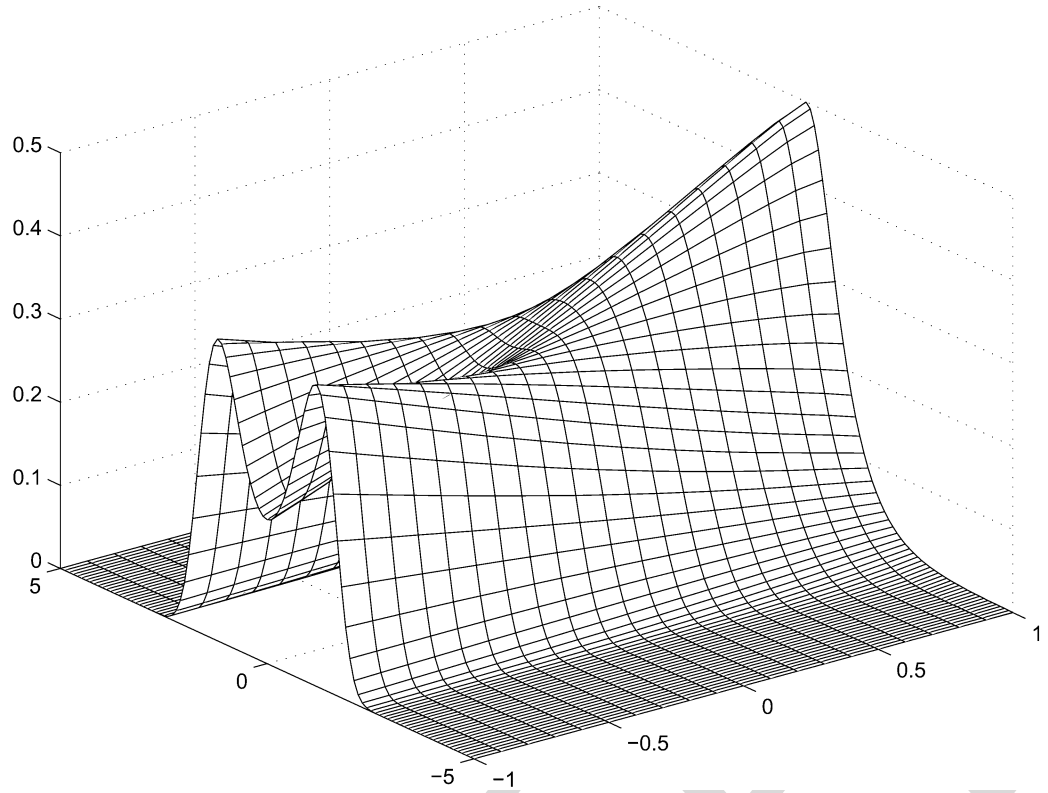

Fig. 1. The waveform of the homotopy family varying θ from -1 to 1.

Different parameters, α_1, α_2, and α_3, lead to different existing algorithms, such as the natural gradient algorithm [7] and the equivariant algorithm [13]. It should be noted that different algorithms have different stability regions. Therefore, the choice of the nonlinear activation function is vital to successful separation of source signals. There are a number of criteria to choose adequate activation functions [4]. If a source signal is super-Gaussian, the hyperbolic function $\varphi(y) = \tanh(y)$ is adequate for the activation function. On the other hand, if a source signal is sub-Gaussian, the cubic function is a good candidate for the activation function. However, in most real-world applications, such as biomedical data, we usually do not know the statistics of source signals and the number of active source signals in the measurements. In order to make learning algorithm (7) stable at the vicinity of the true solution, we suggest the online adaption of activation functions using the exponential generative model.

IV. Exponential Generative Model

The exponential generative model for the approximation of pdfs is described by

$$\boldsymbol{\mathcal{E}} = \{p(y, \boldsymbol{\theta}) | p(y, \boldsymbol{\theta}) = \exp\{-\psi(y, \boldsymbol{\theta}) + \mathcal{N}(\boldsymbol{\theta})\} \tag{8}$$

where $\boldsymbol{\theta}$ is the vector of free parameters and $\mathcal{N}(\theta)$ is the normalization term such that the integral of $p(y, \boldsymbol{\theta})$ over the whole interval $(-\infty, \infty)$ is equal to one. The exponential generative model covers a variety of pdfs, such as the generalized Gaussian distribution and the exponential family.

Example 1. Generalized Gaussian Distribution [6], [15], [24] : The generalized Gaussian model is described as

$$p_g(y, \theta, \sigma) = \left[2A(\theta, \sigma)\Gamma\left(1 + \frac{1}{\theta}\right)\right]^{-1} \exp\left(-\left|\frac{y}{A(\theta, \sigma)}\right|^{\theta}\right) \tag{9}$$

where $A(\theta, \sigma) = \sqrt{\sigma^2 \Gamma(1/\theta)/\Gamma(3/\theta)}$, $\Gamma(x) = \int_0^{\infty} \tau^{x-1} e^{-\tau} d\tau$ is the standard Gamma function, σ is the variance of random variable y, and θ is a free parameter that describes the sharpness of the distribution function. If $r = 2$, then $p_g(y, 2, 1)$ is the Gaussian distribution. If $r = 1$, then $p_g(y, 1, 1)$ is the Laplacian distribution.

Example 2. Exponential Family [8], [12] : The exponential family can be expressed in term of certain functions $\{C(y), F_1(y), \ldots, F_N(y)\}$ and a function $\mathcal{N}(\boldsymbol{\theta})$ as

$$p_e(y, \boldsymbol{\theta}) = \exp\left[-C(y) - \sum_{i=1}^{N} \theta_i F_i(y) + \mathcal{N}(\boldsymbol{\theta})\right]. \tag{10}$$

Here, $\psi(y, \boldsymbol{\theta}) = C(y) + \sum_{i=1}^{N} \theta_i F_i(y)$. There are some good properties, such as *flatness*, as a statistical model. Refer to [8] for a detailed discussion.

A. Construction of Exponential Generative Model

Here, we provide a feasible way to construct the exponential generative model for blind source separation. First, we define a activation function family with parameter $\boldsymbol{\theta}$

$$\{\varphi(y, \boldsymbol{\theta})\} \tag{11}$$

where $\boldsymbol{\theta}$ is the parameter to be determined. From the definition of the activation functions for blind separation

$$\varphi(y,\boldsymbol{\theta}) = -\frac{d\log p(y,\boldsymbol{\theta})}{dy} \tag{12}$$

or, equivalently, the pdf is given by

$$p(y,\boldsymbol{\theta}) = \exp\left\{-\int_0^y \varphi(\tau,\boldsymbol{\theta})d\tau + \mathcal{N}(\theta)\right\} \tag{13}$$

where $\mathcal{N}(\boldsymbol{\theta})$ is the normalization term.

Example 3. Homotopy Family: In blind separation, it is well known that the hyperbolic tangent function $\varphi(y) = \tanh(y)$ is a good activation function for super-Gaussian sources and the cubic function $\varphi(y) = y^3$ is a favorite choice for sub-Gaussian sources [5], [14], [18]. We can construct a homotopy family for the activation function space in the form

$$\varphi(y,\boldsymbol{\theta}) = \theta\tanh(y) + (1-\theta)y^3. \tag{14}$$

Therefore, we can construct the exponential generative model as

$$p_e(y,\theta) = \exp\left(-\theta\frac{|y|^4}{4} - (1-\theta)\log\operatorname{sech}(y) + \mathcal{N}(\theta)\right). \tag{15}$$

In this exponential generative model, $\psi(y,\theta) = \theta(|y|^4/4) + (1-\theta)\log\operatorname{sech}(y)$ and $\mathcal{N}(\theta)$ is the normalization term. When we vary parameter θ from 0 to 1, the pdf $p_e(y,\theta)$ changes from $p(y) = (2/\pi)\operatorname{sech}(y)$ to $p_g(y,4,1)$, defined in (9). Fig. 1 shows the waveform of the homotopy family varying θ from -1 to 1.

V. Adaptation of Activation Functions

In this section, we present a natural gradient approach to adapt the activation functions for blind source separation. The basic idea is to use an exponential generative family as a model for pdfs. The objective of blind source separation is to minimize the cost function

$$L(\boldsymbol{\theta},\mathbf{W}) = E[l(\mathbf{y},\boldsymbol{\theta},\mathbf{W})] \tag{16}$$

where $l(\mathbf{y},\boldsymbol{\theta},\mathbf{W})$ is defined by (4), $q_i(y_i,\boldsymbol{\theta}_i)$ is an approximate distribution of y_i in the exponential generative model (8), and $\boldsymbol{\theta} = (\boldsymbol{\theta}_1^T,\ldots,\boldsymbol{\theta}_m^T)^T$ is the vector of parameters to be determined adaptively. The existing algorithms for ICA usually adapt only demixing matrix $\mathbf{W}$, where $q_i(y_i,\boldsymbol{\theta}_i)$ or $\boldsymbol{\theta}_i$ are adequately chosen. The algorithm fails if the choice is inadequate. In this paper, we suggest not only to train the demixing matrix $\mathbf{W}$, but also to adapt the parameters in the exponential generative family simultaneously. Therefore, we attempt to find an adequate pdf in the exponential generative model, which minimizes the above cost function. The cost function $L(\boldsymbol{\theta},\mathbf{W})$ is minimized when $q_i(y_i,\boldsymbol{\theta}_i)$ is chosen to be the true pdf in the sense of KL **AU: PLEASE SPELL OUT "KL" —ED.** divergence. This justifies our approach.

For each component y_i of the output of the demixing model, we use the exponential generative model to approximate the distribution of y_i

$$q_i(y_i,\boldsymbol{\theta}_i) = \exp\big(-\psi(y_i,\boldsymbol{\theta}_i) + \mathcal{N}_i(\boldsymbol{\theta}_i)\big). \tag{17}$$

By minimizing the cost function (16) with respect to $\boldsymbol{\theta}$, using the gradient-descent approach, we derive learning algorithms for training parameters $\boldsymbol{\theta}$.

A. Gradient-Descent Learning

First, we apply the gradient-descent approach to train the parameters θ_i in the exponential generative model. Substituting (17) in the cost function (16), we obtain the derivative

$$\frac{\partial l(\mathbf{y},\boldsymbol{\theta},\mathbf{W})}{\partial\boldsymbol{\theta}_i} = \frac{\partial\psi(y_i,\boldsymbol{\theta}_i)}{\partial\boldsymbol{\theta}_i} - \mathcal{N}'(\boldsymbol{\theta}_i). \tag{18}$$

Therefore, the learning rule for updating $\boldsymbol{\theta}_i$ is described as

$$\Delta\boldsymbol{\theta}_i = -\eta\left(\frac{\partial\psi(y_i,\boldsymbol{\theta}_i)}{\partial\boldsymbol{\theta}_i} - \mathcal{N}'(\boldsymbol{\theta}_i)\right). \tag{19}$$

In particular, applying the learning rule (19) to the parameterized generative model (13), we obtain the adapting rule

$$\Delta\boldsymbol{\theta}_i = \eta\left(\int_0^{y_i}\frac{\partial\varphi(\tau,\boldsymbol{\theta})}{\partial\boldsymbol{\theta}_i}d\tau - \mathcal{N}'(\boldsymbol{\theta}_i)\right). \tag{20}$$

From the cost function (4), we see that the minimization of mutual information is equivalent to the maximum likelihood for parameters $\boldsymbol{\theta}_i$, because the first term in (4) does not depend on $\boldsymbol{\theta}_i$. Thus, it should be noted that the above learning rule is actually equivalent to the maximum log-likelihood algorithm for each component.

B. Natural Gradient Learning

When a parameter space has a certain underlying structure, the ordinary gradient of a function does not represent its steepest direction. Thus, the learning rule based on the ordinary gradient descent is sometimes very slow and suffers from the plateau phenomenon. The steepest descent direction in a Riemannian space is given by the natural gradient [2], which takes the form of

$$\tilde{\nabla}_{\boldsymbol{\theta}_i}L = \mathcal{G}_i^{-1}\nabla_{\boldsymbol{\theta}_i}L \tag{21}$$

where the matrix $\mathcal{G}_i$ is the Riemannian metric of the parameterized space. The Riemannian structure of the parameter space of statistical model $\{p(y,\boldsymbol{\theta})\}$ is defined by the Fisher information [1], [25]

$$\mathcal{G}_i(\boldsymbol{\theta}) = E\left[\frac{\partial\log p(y,\boldsymbol{\theta})}{\partial\boldsymbol{\theta}_i}\frac{\partial\log p(y,\boldsymbol{\theta})^T}{\partial\boldsymbol{\theta}_i}\right] \tag{22}$$

in the component form. Since Fisher information $\mathcal{G}_i(\boldsymbol{\theta}))$ is evaluated by the expectation of $(\partial\log p(y,\boldsymbol{\theta})/(\partial\boldsymbol{\theta}_i))(\partial\log p(y,\boldsymbol{\theta})^T/(\partial\boldsymbol{\theta}_i))$, we make use of an adaptive method to estimate the Fisher information, which is given by

$$\overline{\mathcal{G}}_i(k+1) = (1-\epsilon_k)\overline{\mathcal{G}}_i(k) + \epsilon_k\frac{\partial\log p(y,\boldsymbol{\theta})}{\partial\boldsymbol{\theta}_i}\frac{\partial\log p(y,\boldsymbol{\theta})^T}{\partial\boldsymbol{\theta}_i} \tag{23}$$

where ϵ_k is a time-dependent learning rate. When the dimension of parameter $\boldsymbol{\theta}_i$ is large, the computing cost will be expensive for the inversion of Fisher information $\mathcal{G}_i$ to realize the natural gradient learning. In order to overcome the problem, Amari, *et*

al. [9] proposed an adaptive approach to directly estimate the inverse of Fisher information $\mathcal{G}_i$, which is given by

$$\mathcal{H}_i(k+1) = (1+\epsilon_k)\mathcal{H}_i(k) - \epsilon_k \mathcal{H}_i(k) \frac{\partial \log p(y,\boldsymbol{\theta})}{\partial \boldsymbol{\theta}_i} \frac{\partial \log p(y,\boldsymbol{\theta})^T}{\partial \boldsymbol{\theta}_i} \mathcal{H}_i(k). \quad (24)$$

The estimated matrix $\mathcal{H}_i(k)$ is used to approximate the inverse of Fisher information $\mathcal{G}_i$ and the natural gradient learning algorithm is modified to the form

$$\Delta \boldsymbol{\theta}_i(k) = -\eta_k \mathcal{H}_i(k) \nabla_{\boldsymbol{\theta}_i} L. \quad (25)$$

Refer to [9] for a detailed discussion on online estimation for the Fisher information.

The dynamical behavior of natural gradient online learning has been analyzed and proved to be Fisher efficient, implying that it has asymptotically the same performance as the optimal batch estimation of parameters [2]. In Section VI, we will show that the natural gradient learning can overcame long plateaus that appear in ordinary gradient-descent learning. We will further discuss the convergence and stability of the natural gradient-learning algorithm based on the generalized Gaussian model.

Remark: Actually, from the semiparametric statistical theory for blind separation [3], minimization of cost function (16) may not lead to the true distribution of source signals. However, it suffices for us to choose adequate activation functions such that the true solution is a stable equilibrium of the learning algorithm.

C. Consistency

One important question is if it is consistent to update the demixing model $\mathbf{W}$ using the natural gradient algorithm and to estimate the $\boldsymbol{\theta}$ using the maximum likelihood at the same time. In fact, the learning rule for $\boldsymbol{\theta}$ by maximizing the log likelihood is equivalent to the one by minimizing the mutual information. This means that both learning rules for updating parameters $\boldsymbol{\theta}$ and demixing model $\mathbf{W}$ make the cost function $L(\boldsymbol{\theta}, \mathbf{W})$ in (16) decrease, provided that the learning rate is sufficiently small.

VI. Generalized Gaussian Model

In this section, we elaborate the generalized Gaussian family for blind source separation. Here, we emphasis that both the sharpness and the normalization term of the distribution play important roles in the adaptation of activation functions. We will see that the equilibrium of the estimator depends on the normalization term. The reason for studying the generalized Gaussian model is two-fold. From an analytic perspective, the generalized Gaussian family is quite flexible, covering a wide range of density functions. From the practical point of view, the generalized Gaussian distribution has been known to successfully model the characteristics of a variety of physical phenomena. The activation function family, commonly used for ICA algorithm

$$\varphi(y,\theta) = sign(y)|y|^{\theta-1} \quad (26)$$

is also derived from this generalized Gaussian family [14], [15]. We know that $\varphi(y,\theta)$, $1 \le \theta < 2$ is an adequate activation function for super-Gaussian signals and that $\varphi(y,\theta)$, $\theta > 2$ is good for sub-Gaussian signals. However, we usually do not know how many source signals are sub-Gaussian and how many super-Gaussian from the mixed sensor signals. In this paper, we suggest a way to use (25) to adapt the parameter $\boldsymbol{\theta}$.

In the generalized Gaussian distribution family (9), there are two free parameters: the variance σ and the sharpness θ. It is known that the solution to blind separation has certain ambiguities: scaling and permutation. The variance σ corresponds to the scaling of the recovered signal.

In order to reduce the complexity of estimation of the parameters in the exponential generative family, we employ the learning algorithm such that the outcome of the demixing model has unit variance. Therefore, it is not necessary to estimate the variance σ in the exponential generative family: we just set $\sigma = 1$ and use the notation $A(\theta)$ for $A(\theta,1)$ for simplicity. Now, the generalized Gaussian distribution is simplified as

$$p_g(y,\theta) = \exp\left(-\left|\frac{y}{A(\theta)}\right|^{\theta} + \mathcal{N}(\theta)\right) \quad (27)$$

where $\mathcal{N}(\theta) = -\log(2A(\theta)\Gamma(1+1/\theta))$.

A. Adaptation Rule

The ordinary gradient of the cost function (4) is give by

$$\frac{\partial l(\mathbf{y},\boldsymbol{\theta},\mathbf{W})}{\partial \theta_i} = -\frac{\partial \phi(y_i(k),\theta_i)}{\partial \theta_i} + \mathcal{N}'(\theta_i) \quad (28)$$

where

$$\frac{\partial \phi(y_i(k),\theta_i)}{\partial \theta_i} = \left|\frac{y_i}{A(\theta_i)}\right|^{\theta_i} \left[\log\left|\frac{y_i}{A(\theta_i)}\right| - \frac{\theta_i A'(\theta_i)}{A(\theta_i)}\right]. \quad (29)$$

The ordinary gradient-descent learning algorithm for estimating the activation function of the ith component of the demixing model is described by

$$\Delta\theta_i = -\eta \frac{\partial l(\mathbf{y},\boldsymbol{\theta},\mathbf{W})}{\partial \theta_i}. \quad (30)$$

Correspondingly, the natural gradient algorithm is given by

$$\Delta\theta_i = -\eta \mathcal{G}_i^{-1} \frac{\partial l(\mathbf{y},\boldsymbol{\theta},\mathbf{W})}{\partial \theta_i} \quad (31)$$

where $\mathcal{G}_i$ is the Fisher information defined by

$$\mathcal{G}_i = E\left[\frac{\partial l(\mathbf{y},\boldsymbol{\theta},\mathbf{W})}{\partial \theta_i} \frac{\partial l(\mathbf{y},\boldsymbol{\theta},\mathbf{W})}{\partial \theta_i}^T\right]. \quad (32)$$

The ordinary gradient algorithm (30) and the natural gradient algorithm (31) have the same set of equilibria, but have different learning dynamics.

B. Equilibria of Learning Dynamics

In this section, we analyze the equilibria of the learning dynamics of Laplacian, Gaussian, and Sub-Gaussian signals. For simplicity, we neglect the subscript i in the following discussion

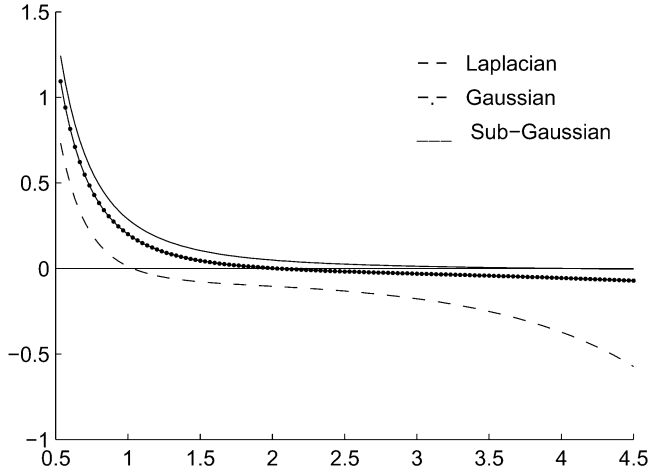

Fig. 2. The equilibria of ordinary gradient adaptation rule for the Laplacian, Gaussian and Sub-Gaussian distributions using the generalized Gaussian family $p_g(y,\theta)$.

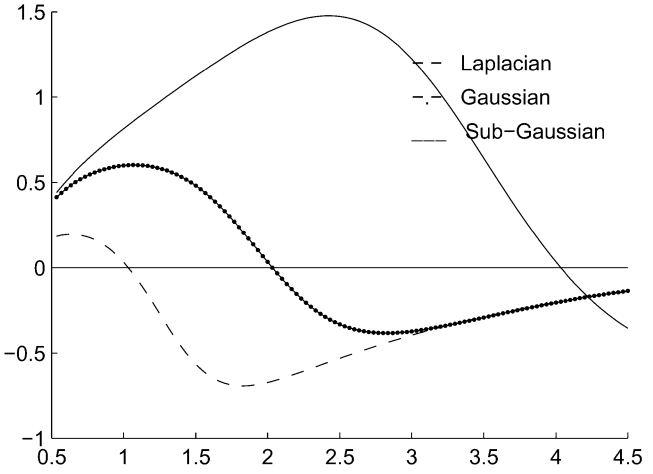

Fig. 3. The equilibria of natural gradient adaptation rule for Laplacian, Gaussian and Sub-Gaussian distributions using the generalized Gaussian family $p_g(y,\theta)$.

if it does not raise any ambiguity. From the statistical learning theory, we know that the equilibria of the updating rule satisfy

$$E\left[\frac{\partial\psi(y_i(k),\theta_i)}{\partial\theta_i}\right]=\mathcal{N}'(\theta_i). \tag{33}$$

Assuming that signal y_i is a random variable with the pdf $p_i(y_i)$, by the law of large numbers in statistics, we have

$$E\left[\frac{\partial\psi(y_i(k),\theta_i)}{\partial\theta_i}\right]=\int\frac{\partial\psi(\xi,\theta_i)}{\partial\theta_i}p_i(\xi)d\xi. \tag{34}$$

In order to estimate the equilibria of learning dynamics, we define

$$f(\theta_i)=\int\frac{\partial\psi(\xi,\theta_i)}{\partial\theta_i}p_i(\xi)d\xi-\mathcal{N}'(\theta_i). \tag{35}$$

With the help of numerical calculation, we plot the curves of the above function for the following three different distributions: $p_g(y,1)$, $p_g(y,2)$, and $p_g(y,4)$. The zeros of the function $f(\theta)$ depend on the distribution of the random variable. If the random variable has Laplacian distribution $p_g(y,1)$, the function $f(\theta)$ has a unique zero $\theta=1$ in the interval $[0.5,4.5]$. If the random variable has Gaussian distribution $p_g(y,2)$, the function $f(\theta)$ has a unique zero $\theta=2$ in the interval $[0.5,4.5]$. If the random variable has sub-Gaussian distribution $p_g(y,4)$, the function $f(\theta)$ has a unique zero $\theta=4$ in the interval $[0.5,4.5]$.

Fig. 2 illustrates the equilibria of function $f(\theta)$ for the three different distributions. It should be noted that the properties of these zeros are different. They have different slopes, which affect the convergence rate of the learning algorithm. If the natural gradient method is used for training parameters θ_i, the performance of learning will improve dramatically. Fig. 3 shows the curves of the following function for the three different distributions:

$$f_N(\theta_i)=\mathcal{G}_i(\theta_i)^{-1}\left[\int\frac{\partial\psi(\xi,\theta_i)}{\partial\theta_i}p_i(\xi)d\xi-\mathcal{N}'(\theta_i)\right]. \tag{36}$$

The slopes in Fig. 3 in the vicinity of the equilibria are much steeper than the slopes in Fig. 2. This indicates that the natural gradient algorithm will give a better learning performance than the ordinary gradient algorithm.

VII. Stability Analysis

In this section, we study the stability of the learning algorithms both for the activation function and the demixing model with the help of numerical calculation.

A. Stability of Algorithm for Activation Functions

For each component of the output of the demixing model, we employ learning algorithm (19) to estimate the parameters in the activation functions. It is easily seen that the equilibrium of the learning algorithm satisfies

$$E\left[\frac{\partial\phi(y,\theta)}{\partial\theta}\right]-\mathcal{N}'(\theta)=0. \tag{37}$$

The statistical learning dynamics can be described as

$$\frac{d\theta}{dt}=f(\theta)=E\left[\frac{\partial\phi(y,\theta)}{\partial\theta}\right]-\mathcal{N}'(\theta). \tag{38}$$

The stability of the above dynamical system depends on Hessian matrix $E[\partial^2L(\boldsymbol{\theta},\mathbf{W})/\partial\boldsymbol{\theta}^2]$ at the equilibrium. For the Laplacian signal, the Hessian $E[\partial^2L(\boldsymbol{\theta},\mathbf{W})/\partial\boldsymbol{\theta}^2]=0.82$, which means that the equilibrium point is stable. Similarly, we can calculate the Hessian for the Gaussian and sub-Gaussian signals. They are also positive. Therefore, the other equilibria are also stable for Gaussian and sub-Gaussian distribution, respectively.

B. Stability of Algorithm for Demixing Model

In order to make the outputs of the demixing model have unit variance, we choose Cardoso's equivariant algorithm

$$\Delta\mathbf{W}=\eta\left(\mathbf{I}-\mathbf{y}\mathbf{y}^T-\boldsymbol{\varphi}(\mathbf{y})\mathbf{y}^T+\mathbf{y}\boldsymbol{\varphi}^T(\mathbf{y})\right)\mathbf{W} \tag{39}$$

where $\boldsymbol{\varphi}(\mathbf{y})=(\varphi_1(y_1),\ldots,\varphi_m(y_m))^T$ and

$$\varphi_i(y_i)=\frac{\theta_i}{A(\theta_i)}\left|\frac{y_i}{A(\theta_i)}\right|^{\theta_i-1}\operatorname{sign}(y_i),\quad\theta_i\geq1. \tag{40}$$

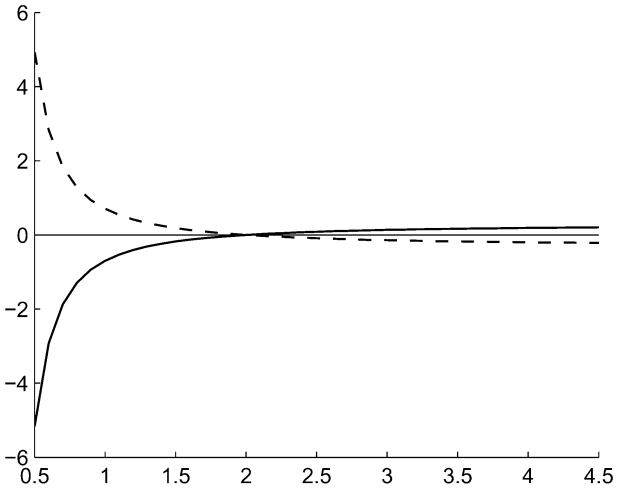

Fig. 4. Statistics of κ_i with different activation functions $\varphi_L(y)$ (dotted line) and $\varphi_S(y)$ (solid line), respectively.

The stability condition for the algorithm is

$$\kappa_i + \kappa_j > 0, \text{ for } 1 \le i \le j \le m \tag{41}$$

where $\kappa_i = E[\varphi_i'(y_i) - y_i\varphi_i(y_i)]$ for $i = 1, \ldots, m$. We will prove, that for the Laplacian and sub-Gaussian signals with distribution $p_g(y, 4)$, statistics $\kappa_i > 0$. For the Gaussian signal, $\kappa_i = 0$.

For the Laplacian distribution, the equilibrium of learning algorithm (31) is $\theta_i = 1$ and the corresponding activation function is $\varphi_L(y_i) = sign(y_i)$. Furthermore, we will prove that, for a random variable y_i with distribution $p_g(y_i, \theta)$, $0.5 < \theta < 2$, the condition $\kappa_i > 0$ is satisfied. To this end, we define a function with variable θ

$$g_L(\theta) = \int (\varphi_L'(\xi) - \xi\varphi_L(\xi))p_g(\xi, \theta)\xi. \tag{42}$$

Substituting $\varphi_L(y_i) = \text{sign}(y_i)$ and (9) into (42) and integrating (42) over $(-\infty, \infty)$ with respect to ξ, we obtain the explicit expression of $g_L(\theta)$

$$g_L(\theta) = 2C(\theta)\frac{\theta - A^2\Gamma(\frac{2}{\theta})}{\theta} \tag{43}$$

where $C(\theta) = (2A(\theta)\Gamma(1/\theta + 1))^{-1}$. Fig. 4 plots the function $g_L(\theta)$ over interval $[0.5, 4.5]$. It is seen that in interval (0.5, 2), function $g_L(\theta)$ is positive. This indicates that $\kappa_i > 0$ in the interval with the activation function $\varphi_L(y_i) = \text{sign}(y_i)$.

Similarly, we can also analyze the statistics κ_i for sub-Gaussian signals. In this case, we choose the activation function $\varphi_S(y_i) = 4/A(4)[y_i/A(4)]^3$. Correspondingly, we define the following function:

$$g_S(\theta) = \int (\varphi_S'(\xi) - \xi\varphi_S(\xi))p_g(\xi, \theta)\xi. \tag{44}$$

Substituting the expressions $\varphi_S(y_i)$ and (9) into (44) and integrating (44) over $(-\infty, \infty)$ with respect to ξ, we obtain easily the explicit expression of $g_S(\theta)$

$$g_S(\theta) = \frac{8C(\theta)A^3(\theta)}{A(4)\theta}\left(3\Gamma\left(\frac{3}{\theta}\right) - A^2(\theta)\Gamma\left(\frac{5}{\theta}\right)\right). \tag{45}$$

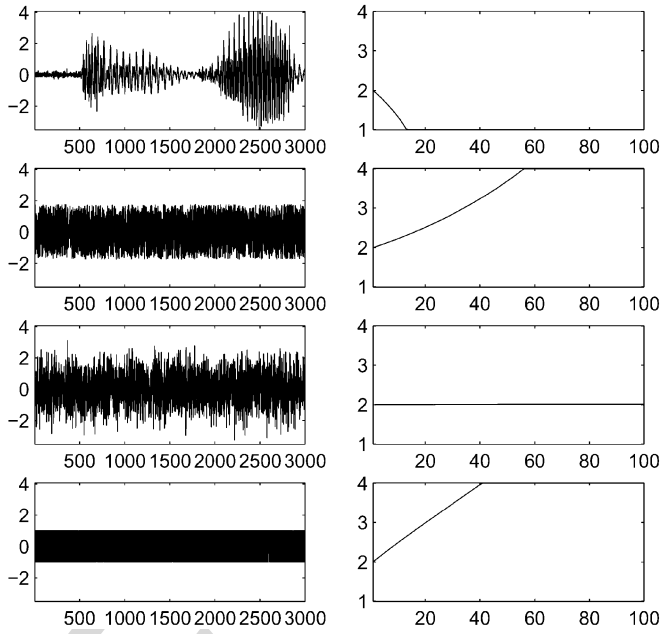

Fig. 5. Adaptation dynamics of parameters for speech signal, i.i.d., Gaussian signal and binary signal.

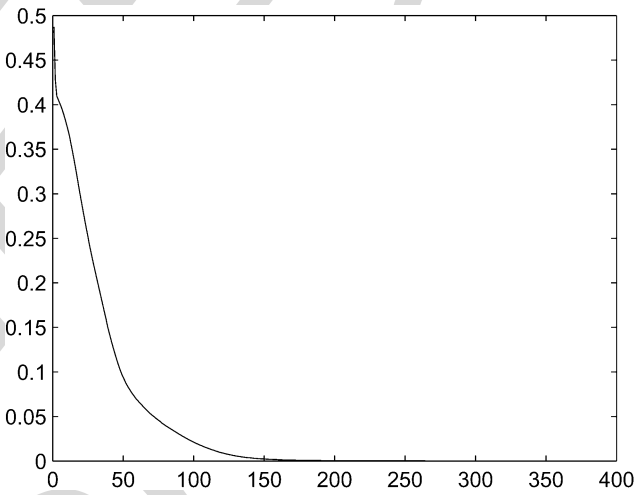

Fig. 6. Averaged convergence performance of cross-talk intersymbol interference (ISI) with well-conditioned mixtures.

Fig. 4 plots the function $g_S(\theta)$ over interval $[0.5, 4.5]$. It is seen that, in interval (2,4], function $g_S(\theta)$ is positive, which means that $\kappa_i > 0$ in the interval with the activation function $\varphi_S(y_i)$. In the same way, it is easy to verify that $\kappa_i = 0$ for Gaussian signals with activation function $\varphi_G(y) = y^2$.

Therefore, from the above analysis, we infer that learning algorithm (19) always makes the true solution a stable equilibrium of learning dynamics if the number of Gaussian source signals is less than one. Furthermore, the adaptation rule is able to identify the statistical properties of source signals. For example, we can consider the separated signal as super-Gaussian if its corresponding parameter θ_i is less than 2.

In this framework, the true solution is always the locally stable equilibrium of the learning process regardless of source distributions, if we adapt both the demixing model and the activation functions. This property is called universal convergence.

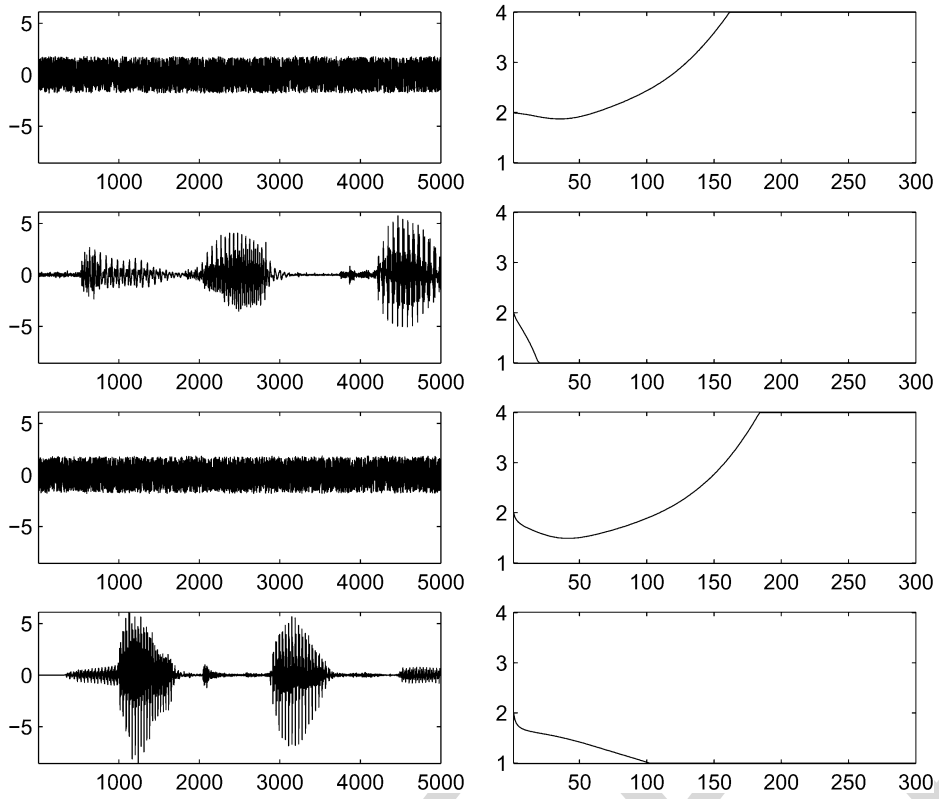

Fig. 7. Adaptation dynamics of parameters in the generalized Gaussian family for a well-conditioned mixture.

VIII. Simulations

In this section, we give a number of computer simulations to demonstrate the effectiveness and performance of the proposed adaptation rule for the activation function.

Example 1: In this example, we intend to show the performance of (25) for four different types of signals, including speech signal, independent identically distributed (i.i.d.) signals uniformly distributed in $[-1, 1]$, Gaussian signals, and binary signals. The first three signals are considered to be the super-Gaussian, sub-Gaussian, and Gaussian signals, respectively. The generalized Gaussian model is used to model the distribution of sources and (25) is employed to train the parameters. The initial guess is set to $\theta = 2$ for all four signals. We restrict parameter θ to the interval [1], [4] to avoid singularity during training. The first column of Fig. 5 plots these four signals and second column shows their histograms of leaning dynamics for the parameter θ, correspondingly.

We see from this simulation that parameter θ for the binary signal converges to 4 although the distribution of binary signals is bimodal, which does not belong in the generalized Gaussian model. As we know, $\varphi(y) = y^3$ is a good activation function for binary signals, which can ensure the stability of (7). This indicates that it is not necessary to precisely estimate the distribution of source signals; instead, we need only to estimate the class of source signals, such as super-Gaussian and sub-Gaussian. This simplification will dramatically reduce the computing cost.

Another observation is that the natural gradient learning for θ can improve learning performance as compared with the ordinary gradient learning. For super-Gaussian signals, it usually takes less than 20 iterations to reach its equilibrium, while for sub-Gaussian signals, it takes less than 100 iterations to reach the satisfactory solution.

Example 2: In this simulation, we would like to illustrate the learning performance of the proposed algorithm when the mixed sensor signals are used as training data. We choose four signals as the source signals. The first two, $s_1(k)$ and $s_2(k)$, are speech signals, which are considered to be super-Gaussian, and the last two, $s_3(k)$ and $s_4(k)$, are i.i.d. signals uniformly distributed in $[-1, 1]$, which are regarded as sub-Gaussian signals. If the same activation function is used for all components, (7) will fail to converge to the true solution, because the stability conditions are not satisfied. Here, we employ the generalized Gaussian family to approximate the distribution functions of the output signals. Learning algorithm (25) is used to adapt the activation function of each component of the outputs and (7) is employed to train the demixing matrix $\mathbf{W}$.

A large number of simulations are performed to demonstrate the performance of the learning strategy. Mixing matrix $\mathbf{A}$ is randomly generated by computer. Sensor signals $\mathbf{x} = \mathbf{A}\mathbf{s}$ are used as training data. In order to evaluate the general performance of the algorithm, we use the average of the cross-talk index.

If mixing matrix $\mathbf{A}$ is well conditioned (say, the condition number $cond(\mathbf{A}) \leq 20$), parameter θ will converge to the true value within 100 iterations for super-Gaussian and within 200 iterations for sub-Gaussian signals, respectively. Fig. 6 illustrates the averaged histogram of the cross-talk index of 100 trials.

Fig. 7 illustrates the histogram of $\boldsymbol{\theta}(k)$, where the first column is the output signals of the demixing model. From this example,

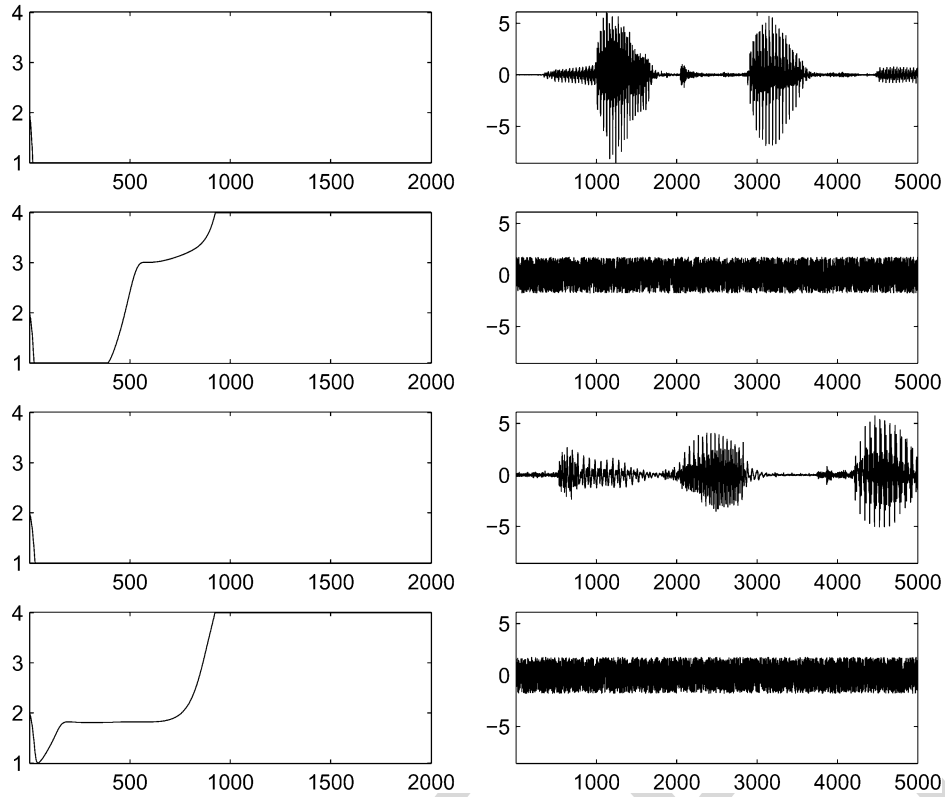

Fig. 8. Adaptation dynamics of parameters in the generalized Gaussian family for an ill-conditioned mixture.

we observed that the convergence of θ for super-Gaussian signals is much faster than that for sub-Gaussian ones. The learning processes of θ and $\mathbf{W}$ are closely correlated. Only when θ approaches to an adequate value, i.e., $\theta < 2$ for super-Gaussian and $\theta > 2$ for sub-Gaussian, the demixing matrix will converge to the true solution.

If mixing matrix $\mathbf{A}$ is ill conditioned (say, the condition number $cond(\mathbf{A}) \geq 1000$), the algorithm is still convergent, but has different learning dynamics. Here, we give an example. The mixing matrix is the Hilbert matrix

$$\mathbf{A} = \left[\frac{1}{i+j}\right]_{4\times 4}. \tag{46}$$

The Hilbert matrix is ill conditioned with condition number $cond(A) = 1.5514 \times 10^4$. Fig. 8 illustrates the histogram of parameters $\boldsymbol{\theta}$ during learning process by using algorithm (25). The demixing matrix $\mathbf{W}(k)$ converges to

$$\mathbf{W} = 1.0e+03 \begin{bmatrix} 0.0122 & -0.0374 & 0.0387 & -0.0127 \\ -0.3838 & 2.1627 & -3.4845 & 1.7041 \\ 0.3830 & -2.4500 & 4.2385 & -2.1699 \\ -0.1702 & 1.1761 & -2.1348 & 1.1294 \end{bmatrix}. \tag{47}$$

Example 3. Noisy Case: This simulation is performed to demonstrate the noise tolerance of the parameter estimator. The signal-to-noise ratio (SNR) is defined as

$$\mathrm{SNR} = 10\log_{10}\left(\frac{\sigma_s^2}{\sigma_e^2}\right) \tag{48}$$

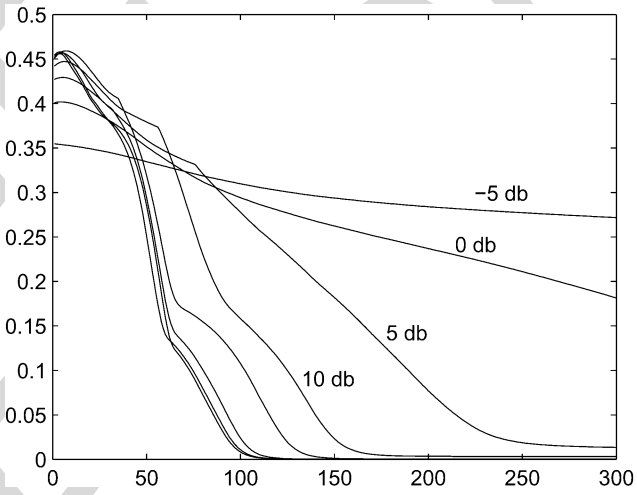

Fig. 9. Adaptation dynamics of cross-talk index for different SNR, varying from 30 db to -5 db.

where σ_s^2 and σ_e^2 are the variances of signals and noises, respectively. The four sources are the same as in example 2. Mixing matrix $\mathbf{A}$ is chosen as a 6×4 matrix, which is randomly generated by computer. This means that we have six sensor signals and four source signals. White Gaussian noises are added with different energy levels, varying from $\mathrm{SNR} = 30$ to $\mathrm{SNR} = -5$ db. The observed sensor signals

$$\mathbf{x}(k) = \mathbf{A}\mathbf{s}(k) + \mathbf{v}(k) \tag{49}$$

are used to train both the demixing matrix $\mathbf{W} \in \mathbf{R}^{6\times 6}$ by algorithm (7) and the parameters $\boldsymbol{\theta}$ by algorithm (31). Fig. 9 illus-

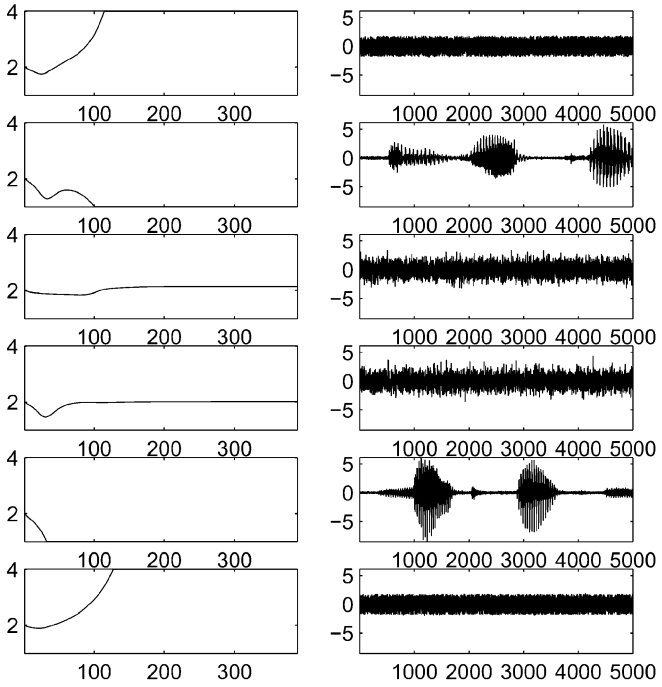

Fig. 10. Learning dynamics of parameters and outputs of the demixing model.

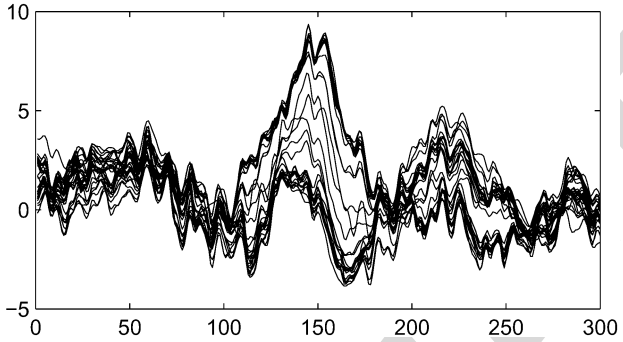

Fig. 11. 62-channel EEG measurements.

trates the histogram of the cross-talk index for different noise levels. From this simulation, we see that the algorithm can tolerate 5-db noise. When the SNR reduces further, the separation performance suddenly decays.

Another observation is that, in this noisy data case when we use a 6×6 matrix as a demixing model, the output of the demixing model are the four source signals and two Gaussian signals. Fig. 10 illustrates the output signals of the demixing model, the first column being the histogram of parameters $\boldsymbol{\theta}$ during learning and the second column being the output signals $\mathbf{y}$.

Example 4. Electroencephalographic (EEG) Data Analysis: In this experiment, we apply the proposed method to analyze the event-related potentials of EEG data. The EEG experiment is designed to study the binocular coordination in the visual system. The purpose of this experiment is to investigate how the visual system integrates the visual neural signals from two eyes. It is well known that binocular rivalry occurs when two different images are presented simultaneously to both the left and right eyes of the subject. Here, we attempt to reveal how the visual system integrates the binocular visual

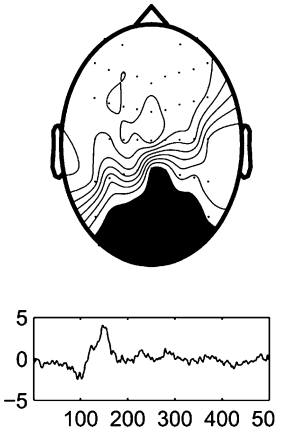

Fig. 12. First component scalp map, separated by our ABS algorithm, which corresponds to the evoked potential in the visual cortex.

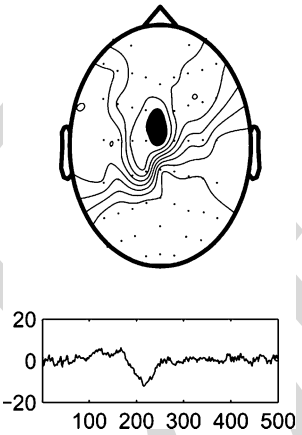

Fig. 13. Second component scalp map, separated by our ABS algorithm, which corresponds to the evoked potential in the prefrontal cortex.

neural information when the images are spatially correlated. To this end, we generate the two images from a picture of human face. We split the picture into two images, which are complementary. Thus, the two images are completely different if we do not concern their context. However, because these two images are complementary, we can recover the original face by merging two images. During the experiment, these two complementary images are presented to the left and right eyes, respectively, of the subject. The EEG data is recorded with a 64-channel NeuroScan with sampling frequency 1000 Hz. In order to increase the SNR, we take 20 trial averaging data as the sensor signals.

Fig. 11 plots 62 channels of EEG measurements. The proposed adaptive blind separation (ABS) method is applied to separate the visual evoked potentials from the EEG measurements. Learning algorithms (7) and (31) are used to train the demixing matrix and parameters in activation functions. The homotopy family is used as the model for the activation functions. We discriminate the sources of interest from noise by using two criteria, the sparseness and temporal structures. Fig. 12 plots the first component of interest, which corresponds to the evoked potential at the visual cortex. Fig. 13 plots the second component of interest, which corresponds to the evoked potential at the prefrontal cortex.

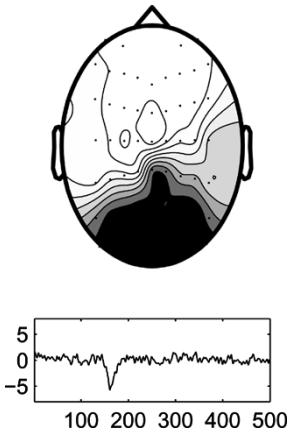

Fig. 14. First component scalp map, separated by the extended infomax algorithm, which corresponds to the evoked potential in the visual cortex.

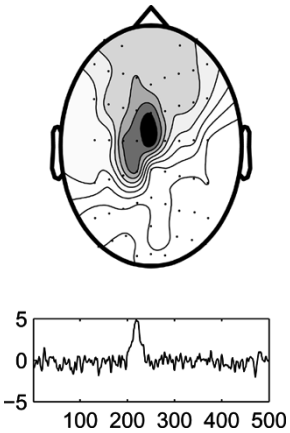

Fig. 15. Second component scalp map, separated by the extended infomax algorithm, which corresponds to the evoked potential in the prefrontal cortex.

In order to compare separation performance of the ABS method with the others, the extended infomax algorithm [26] is also applied to the EEG data to separate the visual evoked potentials. The extended infomax method adapts the activation function by switching between fixed sub- and super-Gaussian nonlinear functions. We also use the same criteria (the sparseness and temporal structures) to select the components of interest from the separated signals. Figs. 14 and 15 plot the scalp maps of two components of interest. It is not difficult to see that both methods can separate the first component of interest, which corresponds to the visual evoked potential in the visual cortex. However, the experiment shows that the proposed ABS method has much better separating performance than the extended infomax method to separate the second component of interest, which corresponds to the neural activity in face recognition.

IX. Conclusion

In this paper, we present an exponential generative model for approximation of the distributions of source signals. A natural gradient algorithm for activation function adaptation is developed based on minimization of mutual information. Convergence and stability analysis of the algorithm are also provided. Both theoretical analysis and computer simulation show that the proposed method has a faster convergence rate than the ordinary gradient method for the activation function adaptation. In this framework, the true solution is always the locally stable equilibrium of the learning process, regardless of source distributions, if we adapt both the demixing model and the activation functions. This property is called universal convergence. This method can also be used to estimate the class of source signals, such as super-Gaussian and sub-Gaussian signals.

Adaptation of activation functions is different from the estimation of the distribution. The main objective of activation function adaptation is to make the true solution a stable equilibrium of learning system. Thus, the number of parameters for each component usually is very small. As a result, such strategy can reduce the computing cost dramatically, as compared with estimation of the distribution functions.

References

[1] S. Amari, *Differential—Geometrical Methods in Statistics*. Berlin, Germany: Springer-Verlag, 1985, vol. 28, Lecture Notes in Statistics.

[2] ——, "Natural gradient works efficiently in learning," *Neural Comput.*, vol. 10, pp. 251–276, 1998.

[3] S. Amari and J.-F. Cardoso, "Blind source separation—Semiparametric statistical approach," *IEEE Trans. Signal Processing*, vol. 45, pp. 2692–2700, Nov. 1997.

[4] S. Amari, T. Chen, and A. Cichocki, "Stability analysis of adaptive blind source separation," *Neural Networks*, vol. 10, pp. 1345–1351, 1997.

[5] S. Amari and A. Cichocki, "Adaptive blind signal processing—Neural network approaches," *Proc. IEEE*, vol. 86, pp. 2026–2048, Oct. 1998.

[6] S. Amari, A. Cichocki, and H. Yang, "Blind signal separation and extraction: Neural and information theoretic approaches," in *Unsupervised Adaptive Filtering*, S. Haykin, Ed. New York: Wiley, 2000, vol. I, pp. 63–138.

[7] S. Amari, A. Cichocki, and H. H. Yang, "A new learning algorithm for blind signal separation," in *Advances in Neural Information Processing Systems 8 (NIPS * 95)*, G. Tesauro, D. S. Touretzky, and T. K. Leen, Eds: <<AU: PLEASE PROVIDE NAME AND LOCATION OF PUBLISHER —ED.>>, 1996, pp. 757–763.

[8] S. Amari and H. Nagaoka, *Methods of Information Geometry*. London, U.K.: Amer. Math. Soc. and Oxford Univ. Press, 2000.

[9] S. Amari, H. Park, and K. Fukumizu, "Adaptive method of realizing natural gradient learning for multilayer perceptrons," *Neural Comp.*, vol. 12, pp. 1399–1409, 2000.

[10] H. Attias, "Independent factor analysis," *Neural Comput.*, vol. 11, no. 4, pp. 803–851, 1999.

[11] A. J. Bell and T. J. Sejnowski, "An information maximization approach to blind separation and blind deconvolution," *Neural Comput.*, vol. 7, pp. 1129–1159, 1995.

[12] G. Box and G. Tiao, *Bayesian Inference in Statistical Analysis*. Reading, MA: Addison-Wesley, 1973.

[13] J.-F. Cardoso and B. Laheld, "Equivariant adaptive source separation," *IEEE Trans. Signal Processing*, vol. 43, pp. 3017–3029, Dec. 1996.

[14] S. Choi, A. Cichocki, and S. Amari, "Flexible independent component analysis," *J. VLSI Signal Process.*, vol. 20, pp. 25–38, 2000.

[15] A. Cichocki, I. Sabala, S. Choi, B. Orsier, and R. Szupiluk, "Self adaptive independent component analysis for sub-Gaussian and super-Gaussian mixtures with unknown number of sources and additive noise," in *Proc. 1997 Int. Symp. Nonlinear Theory and its Application (NOLTA'97)*, 1997, pp. 731–734.

[16] A. Cichocki and R. Unbehauen, "Robust neural networks with on-line learning for blind identification and blind separation of sources," *IEEE Trans. Circuits Syst. I*, vol. 43, pp. 894–906, Nov. 1996.

[17] P. Comon, "Independent component analysis: a new concept?," *Signal Process.*, vol. 36, pp. 287–314, 1994.

[18] S. Douglas, A. Cichocki, and S. Amari, "Multichannel blind separation and deconvolution of sources with arbitrary distributions," in *Proc. IEEE Workshop Neural Networks for Signal Processing (NNSP'97)*, Sept. 1997, pp. 436–445.

[19] J. Eriksson, J. Karvanen, and V. Koivunen, "Source distribution adaptive maximum likelihood estimation of ica model," in *Proc. ICA'00*, P. Pajunen and J. Karhunen, Eds., Helsinki, Finland, June 2000, pp. 227–232.

[20] R. Everson and S. Roberts, "Independent component analysis: A flexible nonlinearity and decorrelating manifold approach," *Neural Comput.*, vol. 11, pp. 1957–1983, 1999.
[21] C. Jutten and J. Herault, "Blind separation of sources, Part I: An adaptive algorithm based on neuromimetic architecture," *Signal Process.*, vol. 24, pp. 1–10, 1991.
[22] T. Lee and M. Lewicki, "The generalized gaussian mixture model using ica," in *Proc. ICA'00*, P. Pajunen and J. Karhunen, Eds., Helsinki, Finland, June 2000, pp. 239–244.
[23] E. Oja and J. Karhunen, "Signal separation by nonlinear hebbian learning," in *Computational Intelligence—A Dynamic System Perspective*, M. Palaniswami, Y. Attikiouzel, R. Marks II, D. Fogel, and T. Fukuda, Eds. Piscataway, NJ: IEEE Press, 1995, pp. 83–97.
[24] D. T. Pham and P. Garat, "Blind separation of mixtures of independent sources through a quasi maximum likelihood approach," *IEEE Trans. Signal Processing*, vol. 45, pp. 1712–1725, July 1997.
[25] C. R. Rao, "Information and accuracy attainable in the estimation of statistical parameters," *Bull. Calcutta Math. Soc.*, vol. 37, pp. 81–91, 1945.
[26] W. T. Lee, M. Girolami, and T. Sejnowski, "Independent component analysis using an extended infomax algorithm for mixed sub-gaussian and super-gaussian sources," *Neural Comput.*, vol. 11, no. 2, pp. 606–633, 1999.
[27] L. Zhang, A. Cichocki, and S. Amari, "Natural gradient algorithm for blind separaiton of overdetermined mixture with additive noise," *IEEE Signal Processing Lett.*, vol. 6, pp. 293–295, Nov. 1999.

Liqing Zhang AU: PLEASE PROVIDE AUTHOR PHOTO IN EITHER TIFF, EPS, OR PS FORMAT, AT 220 DPI —ED. received the B.S. degree in mathematics from Hangzhou University **AU: PLEASE PROVIDE CITY AND COUNTRY FOR UNIV. —ED.** in 1983 and the Ph.D. degree in computer sciences from Zhongshan University, **AU: PLEASE PROVIDE CITY —ED.** China in 1988.

He was with the Department of Automation, South China University of Technology, **AU: PLEASE PROVIDE CITY FOR UNIV. —ED.**,where he became an Associate Professor in 1990 and then a Full Professor in 1995. He joined the Laboratory for Advanced Brain Signal Processing, RIKEN Brain Science Institute, Saitama, Japan, in 1997 as a Research Scientist. Since 2002, he has been with the Department of Computer Sciences, Shanghai Jiaotong University, Shanghai, China. He has published more than 80 papers. His research interests include neuroinformatics, visual computing, adaptive systems, and statistical learning.

Andrzej Cichocki (M'96) **AU: PLEASE PROVIDE AUTHOR PHOTO IN EITHER TIFF, EPS, OR PS FORMAT, AT 220 DPI —ED.** received the M. Sc. (with honors), Ph.D., and Habilitate Doctorate (Dr. Sc.) degrees, all in electrical engineering, from the Warsaw University of Technology, Poland, in 1972, 1975, and 1982, respectively.

Since 1972, he has been with the Institute of Theory of Electrical Engineering and Electrical Measurements, Warsaw University of Technology, where he became a Full Professor in 1991. He spent was with the University Erlangen-Nuernberg, Germany, for a few years as the Alexander Humboldt Research Fellow and a Guest Professor. Since 1995, he has been working in the Brain Science Institute RIKEN, Saitama, Japan, as a Team Leader of the laboratory for Open Information Systems. He currently is Head of Laboratory for Advanced Brain Signal Processing. He is the coauthor of three books: *Adaptive Blind Signal and Image Processing—Learning Algorithms and Applications* (Wiley: New York, 2002), *MOS Switched-Capacitor and Continuous-Time Integrated Circuits and Systems* (Springer-Verlag: Berlin, 1989), and *Neural Networks for Optimization and Signal Processing* (Teubner-Wiley: **AU: PLEASE PROVIDE CITY FOR PUBLISHER —ED.**, 1993) and more than 150 research papers. His current research interests include optimization, bio-informatics, neurocomputing, and signal and image processing, especially analysis and processing of multisensory biomedical data.

Dr. Cichocki is currently Associate Editor of IEEE TRANSACTIONS ON NEURAL NETWORKS and recently became a Member of the core group who established a new IEEE Circuits and Systems Technical Committee for Blind Signal Processing and a Member of Steering Committee of ICA workshops.

Shinichi Amari (M'71–M'88–F'94) **AU: PLEASE PROVIDE AUTHOR PHOTO IN EITHER TIFF, EPS, OR PS FORMAT, AT 220 DPI —ED.** graduated from the University of Tokyo, Japan, in 1958, where he majored in mathematical engineering, and received the Dr. Eng. degree from the University of Tokyo in 1963.

He was an Associate Professor at Kyushu University **AU: PLEASE PROVIDE CITY AND COUNTRY FOR UNIV. —ED.**. He was an Associate and then Full Professor in the Department of Mathematical Engineering and Information Physics, University of Tokyo, where he is currently Professor-Emeritus. He is the Director of RIKEN Brain Science Institute, Saitama, Japan. He has been engaged in research in wide areas of mathematical engineering and applied mathematics, such as topological network theory, differential geometry of continuum mechanics, pattern recognition, mathematical foundations of neural networks, and information geometry.

Dr. Amari was Founding Coeditor-in Chief of *Neural Networks*. He served as President of the International Neural Network Society, Council Member of the Bernoulli Society for Mathematical Statistics and Probability Theory, and is President-Elect of the Institute of Electrical, Information and Communication Engineers. He has been awarded the Japan Academy Award, IEEE Neural Networks Pioneer Award, IEEE Emanuel R. Piore Award, Neurocomputing Best Paper Award, and IEEE Signal Processing Society Best Paper Award, among many others.