

Nonlinear Blind Source Separation Using Higher Order Statistics and a Genetic Algorithm

Ying Tan, *Member, IEEE*, and Jun Wang, *Senior Member, IEEE*

Abstract—Demixing independent source signals from their nonlinear mixtures is a very important issue in many scenarios. This paper presents a novel method for blindly separating unobservable independent source signals from their nonlinear mixtures. The demixing system is modeled using a parameterized neural network whose parameters can be determined under the criterion of independence of its outputs. Two cost functions based on higher order statistics are established to measure the statistical dependence of the outputs of the demixing system. The proposed method utilizes a genetic algorithm (GA) to minimize the highly nonlinear and nonconvex cost functions. The GA-based global optimization technique is able to obtain superior separation solutions to the nonlinear blind separation problem from any random initial values. Compared to conventional gradient-based approaches, the GA-based approach for blind source separation is characterized by high accuracy, robustness, and convergence rate. In particular, it is very suitable for the case of limited available data. Simulation results are discussed to demonstrate that the proposed GA-based approach is capable of separating independent sources from their nonlinear mixtures generated by a parametric separation model.

Index Terms—Blind source separation, feedforward neural networks, genetic algorithms, higher order statistics, nonlinear mixture, statistical independence.

I. INTRODUCTION

IN MANY applications such as biomedical, speech, and sonar/radar signal processing, signals come from multichannels and are received by several sensors. The signals from the sensors are generally mixtures of many independent sources [1], [2]. For example, electrocardiogram signals in medical diagnosis are collected from multiple electrodes placed at different locations of a human body and are used to diagnose the cause of diseases and locate sources of illness. Similarly, electroencephalograph signals collected from eight to 16 sensors at different locations on the scalp of a patient's head are processed to seize spike pulses. The same kind of problem can be also found in aerial processing for sonar and radar signals or speech processing for enhancement of speech signals in a strong background-noise environment. Therefore, signal separation is a basic problem in numerous practical

applications and fields as listed above [3]–[5]. The study of this problem is of great importance in both theory and applications.

The problem of source separation concerns extracting independent signals from their linear or nonlinear mixtures. Source separation may be achieved in different ways according to the amount of available prior information. So-called blind source separation (BSS) seeks to recover original source signals from their mixtures without any prior information on the sources themselves or the mixing parameters of the mixtures. BSS techniques have received extensive attention beyond the context of signal processing due to their very weak requirements or conditions about signal sources and mixing channels. In view of this, we make the following moderate assumptions for our later discussions in this paper: 1) the parametric form of mixtures is known; 2) signal sources are statistically independent; and 3) the number of sensors is equal to that of sources. Because of the weak conditions, from another point of view, the separation system may be seen as a black box that receives mixtures at its inputs and provides the estimation of original sources at its outputs. However, the outputs of the separation system are not known *a priori* due to the nature of blind separation. As a result, we only expect the outputs of the system to be statistically independent. Thus, a BSS algorithm adjusts the internal parameters of the separation system so as to obtain the independence of its outputs via unsupervised learning. When the parameters of the separation system are tuned correctly, an estimation of sources can be obtained at the outputs regardless of the indeterminacies of permutation and scaling [2].

Neural-network models with learning capabilities for online BSS from linear mixing signals were first developed by Herault and Jutten [6]–[8]. Since then, most BSS studies address the case of linear and instantaneous mixtures, which was extended recently to linear convolved mixtures, e.g., for convolutive mixing of sources with causal FIR filters [9], [10], and even nonlinear mixtures [11], [12].

Generally, the mixing process of multiple sensors contains some nonlinear transformation such as the saturation distortion of sensors. The study of nonlinear BSS is more realistic and important than linear BSS in practice. However, blind separation of the original signal in nonlinear mixtures has rarely been addressed because if x and y are two independent random variables, then any nonlinear functions $f(x)$ and $g(y)$ are also statistically independent regardless of the specific form of f and g . That is to say, it is impossible to separate original sources using only the source independence assumption of some unknown nonlinear transformations of the sources [13].

In spite of many difficulties in separating independent sources from nonlinear mixtures, several effective models and

Manuscript received August 12, 1999; revised July 26, 2000, December 14, 2000, and February 21, 2001. This work was supported by the Hong Kong Research Grants Council under Grant CUHK4150/97E.

Y. Tan was with The Chinese University of Hong Kong, Shatin, N.T., Hong Kong, China. He is now with the Department of Electronic Engineering and Information Science, University of Science and Technology of China, Hefei 230027, China (email: ytan@ustc.edu.cn).

J. Wang is with the Department of Automation and Computer-Aided Engineering, The Chinese University of Hong Kong Shatin, N.T., Hong Kong, China (email: jwang@acae.cuhk.edu.hk).

Publisher Item Identifier S 1089-778X(01)04277-1.

methods were proposed recently for nonlinear BSS. Deco [14] studied a volume-conserving nonlinear transform for nonlinear BSS. Pajunen *et al.* [15] used Kohonen's self-organizing map (SOM) to extract sources from nonlinear mixtures. It is a model-free method but suffers from the exponential growth of network complexity and interpolation error in recovering continuous sources. Taleb *et al.* [16] proposed an entropy-based BSS algorithm in post nonlinear mixtures. Yang *et al.* [17] proposed an information back-propagation algorithm for interchannel nonlinear mixtures in the sense of entropy maximization and mutual information minimization, and adopted a sigmoidal nonlinear transformation of the nonlinear model based on the work by Burel [11]. Very recently, Tan *et al.* applied radial basis function networks for nonlinear BSS [12]. These newly developed models are established on the basis of parametric models because it is very important for nonlinear BSS to obtain unique separating results when only the independence of sources are known *a priori*. All of these methods are developed based on gradient-based optimization to avoid computing some unknown quantities in an unsupervised manner. Therefore, these methods are susceptible to the local minima problem during the learning process and are thus limited in many practical applications.

On the other hand, different from the likelihood estimation of probability, which can be performed with local minima, the BSS problem requires obtaining a global optimum. Furthermore, the learning objective functions of the BSS problem are multimodal and highly nonlinear. However, all of the existing learning algorithms of BSS systems are based on stochastic gradient methods such as back propagation, bigradient, and natural gradient [1], [2]. These conventional gradient optimization techniques may converge to "bad" solutions unless good initial values are provided, which is impossible in view of the blind hypothesis. Therefore, it is important to develop new BSS algorithms on the basis of global optimization techniques.

In order to overcome the local minimum problem in many existing methods, here we propose a BSS approach based on a genetic algorithm (GA). In this approach, we first define some cost functions to measure the independence of the outputs, which consist of higher order statistics (HOS) of the outputs. Then, by using a GA to minimize the cost functions, we can obtain highly accurate estimation of original sources at the outputs of the separation system.

Several researchers have applied GAs to blind signal processing. Chen *et al.* [24], [25] studied the application of a micro-GA method proposed by Krishnakumar [26] to blind channel identification based on higher order cumulant fitting. Alkanhal *et al.* [27] made use of GAs for blind identification of nonminimum phase finite-impulse response systems by evaluating a cost function defined in terms of output cumulants.

Yet the applications of GAs to BSS is rare. Most recently, Yoshioka *et al.* [28] reported a preliminary study. This method can separate original images from noise-corrupted images by directly minimizing the Kullback-Leibler (KL) divergence using GAs. However, this method needs to estimate the prior probability of the sources. It is well known that BSS requires the global optimization solution, but serious local minima of current existing approaches obstruct their wide applications in many practical fields [1]–[16].

This paper presents a novel procedure for separating original independent sources from their nonlinear mixtures using a GA and HOS of the outputs of a separation system. The proposed approach differs from previous ones in several aspects. The first aspect is to utilize a global optimization method to learn the unknown parameters of the separation system. The second aspect is to minimize a predetermined cost function that measures the independence of the outputs of the separation system and can be expressed using HOS. The third aspect is that this method is able to handle both linear and nonlinear mixtures. Our simulation studies demonstrate that the GA-based BSS scheme is robust to estimation errors in HOS and can achieve global optimal solutions from any initial values of the separation system. Furthermore, since the number of parameters to be optimized in this problem is usually small, GAs are particularly effective and efficient for this kind of optimization problem.

The paper is arranged as follows. In Section II, we present the principle of the BSS problem. In Section III, we propose two cost functions based on HOS. In Section IV, a GA with some modifications suitable for our problem at hand are described. In Section V, simulation results are given to show the effectiveness and validity of the proposed GA-based method. Finally, conclusions are provided in Section VI.

II. BLIND SOURCE SEPARATION

Based on the specific application and available prior information about source signals, the source mixing process can be described by various mathematical models. Here, we discuss a general mixing model that can describe most of actual mixing processes. Let unobservable source signals be $\mathbf{s}(t) = [s_1(t), s_2(t), \dots, s_n(t)]^T$ with mutually independent and stationary components. We further assume that each source has a moment of any order with a zero mean. The mixture is generally expressed as

$$\mathbf{x}(t) = \mathbf{A}\mathbf{f}(\mathbf{H}\mathbf{s}(t)) \quad (1)$$

where $\mathbf{x}(t) = [x_1(t), x_2(t), \dots, x_n(t)]^T$, the superscript T denotes the transpose operator, $\mathbf{f}(\cdot) = [f_1(\cdot), f_2(\cdot), \dots, f_n(\cdot)]^T$ is an unknown componentwise nonlinear transform function called the nonlinear mixing function, and $\mathbf{H}, \mathbf{A} \in \mathbf{R}^{n \times n}$ are unknown nonsingular mixing matrices that mimic the instantaneous mixtures of signals.

The unknown mixing and proposed separating systems for BSS are illustrated in Fig. 1. As shown in the figure, the unknown mixing system at the left part of Fig. 1 can be modeled as a cascaded instantaneous linear mixing, componentwise nonlinear transformation, and linear mixing. Actually, this model is realistic since most of the practical mixing systems can be well described by Fig. 1, which contains both channel and interchannel nonlinearities. At the same time, we also give the separating system at the right part of Fig. 1, which is an inverse procedure of the mixing process among which the nonlinear separating function $\mathbf{g} = [g_1(\cdot), g_2(\cdot), \dots, g_n(\cdot)]^T$ of the separating system is a parametric inverse function of the nonlinear mixing transform function \mathbf{f} .

The task of the separating system is to recover the unobservable original signals $\mathbf{s}(t)$ from the mixtures $\mathbf{x}(t)$ without any

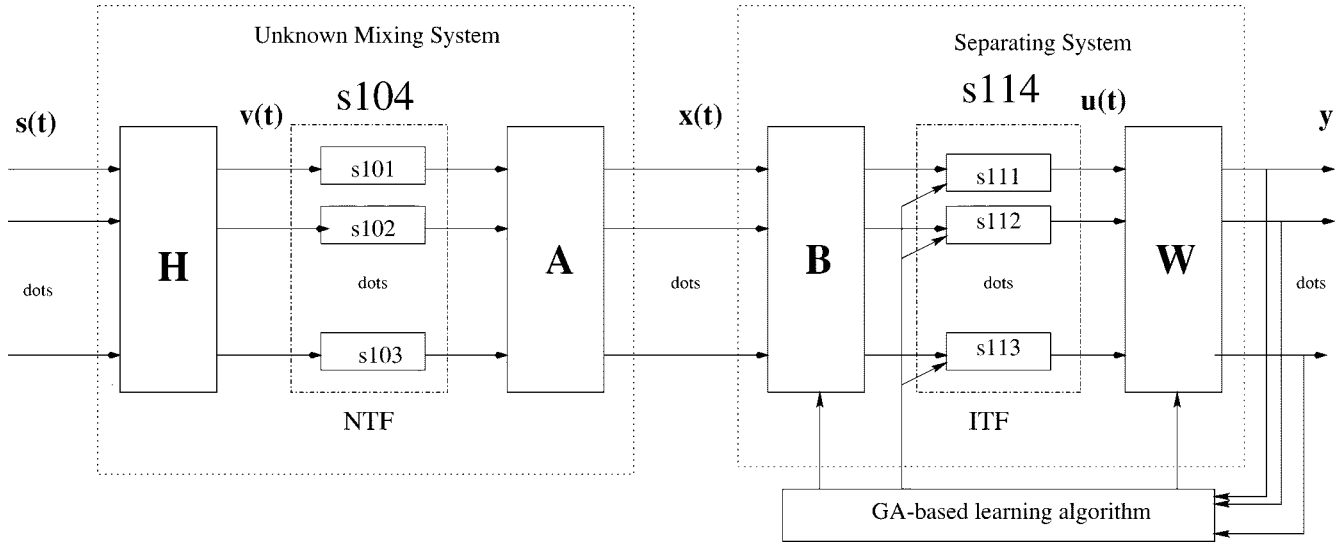


Fig. 1. Mixing and separating systems in BSS.

knowledge of the source signals $s(t)$, the mixing matrices \mathbf{H} , \mathbf{A} , or the parameters of the nonlinear transform function $\mathbf{f}(\cdot)$. Since we assume the nonlinear transform function $\mathbf{f}(\cdot)$ and its inverse $\mathbf{g}(\cdot)$ are of parametric network formulations, our problem is to estimate the parameters of demixing matrices \mathbf{B} and \mathbf{W} and the separating function for the outputs $\mathbf{y}(t)$ to approximate the original sources apart from the indeterminations of a permutation and scaling due to the weakness of the hypotheses. That is to say, the learning algorithm of this BSS problem is to tune the parameters of the separation system so as to achieve independence of the outputs.

As shown in Fig. 1, the outputs of the separating system are

$$\mathbf{y}(t) = \mathbf{W}_g[\mathbf{B}\mathbf{x}(t)]. \quad (2)$$

Substituting (1) into (2), we can obtain

$$\mathbf{y}(t) = \mathbf{P}\mathbf{D}\mathbf{s}(t) \quad (3)$$

where \mathbf{P} is a permutation matrix and \mathbf{D} is a nonsingular and diagonal matrix. Equation (3) holds when the following condition is satisfied:

$$\mathbf{B}\mathbf{A} = \mathbf{I}, \quad \mathbf{g}(\cdot) = \mathbf{f}^{-1}(\cdot), \quad \mathbf{W}\mathbf{H} = \mathbf{P}\mathbf{D}. \quad (4)$$

This means that the components of the outputs \mathbf{y} are independent. Therefore, we have the following proposition.

Proposition: For the mixing and separation systems given in Fig. 1, suppose that we know the parametric formulation of the mixing and separation systems and assume that the number of original sources is equal to that of mixtures, if (4) holds through the adjustment of parameters of the separation system, then the outputs of the separating system $\mathbf{y}(t) = [y_1(t), y_2(t), \dots, y_n(t)]^T$ are mutually independent.

For the nonlinear mixing transform function \mathbf{f} , we assume it has the inverse function \mathbf{f}^{-1} , which also has a parametric form. Specifically, it is seen from Fig. 1 that the separating system can be implemented by a two-layer feed-forward neural network, where the input weight is matrix \mathbf{B} , the nonlinear transform

function \mathbf{g} is the activation function of the neurons in hidden layer, and the output connection weight is given by matrix \mathbf{W} .

Usually, the activation function g can be of the following forms.

- 1) Threshold function

$$g(v) = \begin{cases} 1, & \text{if } v > 0 \\ 0, & \text{if } v = 0 \\ -1, & \text{if } v < 0 \end{cases}. \quad (5)$$

- 2) Piecewise-linear function

$$g(v) = \frac{|1 + kv| - |1 - kv|}{2} = \begin{cases} 1, & \text{if } v \geq \frac{1}{k} \\ kv, & \text{if } \frac{1}{k} > v > -\frac{1}{k} \\ -1, & \text{if } v \leq -\frac{1}{k} \end{cases} \quad (6)$$

where $k > 0$ is the amplification factor inside the linear region of operation and v_s is the saturation level. This form of an activation function may be viewed as an approximation to a nonlinear amplifier. There are two special situations of piecewise-linear functions. One is a linear function if the linear region of operation is maintained without running into saturation. The other is that the piecewise-linear function reduces to a threshold function if the amplification factor of the linear region is made infinitely large.

- 3) Sigmoid function

$$g(v) = \tanh\left(\frac{av}{2}\right) = \frac{1 - \exp(-av)}{1 + \exp(-av)} \quad (7)$$

where $a > 0$ is the gain of the neuron. The sigmoid function is by far the most common form of activation function used in the construction of neural networks. It is a strictly increasing and differentiable function, whereas the threshold function is not.

In particular, if \mathbf{g} is a linear function, Fig. 1 becomes the conventional BSS problem in linear simultaneous mixture, which is well documented. Here we stress the nonlinear BSS problem when \mathbf{g} is chosen to be nonlinear. From (5)–(7), we know that the activation function for the hidden layer in the mixing system

is an unbounded function even on a bounded input region if we employ a two-layer neural network with bounded activation function given in (5)–(7) as the separation system. For example, if \mathbf{g} is the sigmoid function in (7), then $\mathbf{f} = \text{arctanh}(d\mathbf{s})$, where d denotes the parameter of the mixing nonlinear function. In practice, in our simulation studies, as long as the inverse function \mathbf{f}^{-1} can be approximated by a parametric neural network, we can recover the inputs from the outputs of a parametric two-layer neural network with a monotonic activation function for each neuron in the hidden layer. In order to measure the independence of the outputs effectively, we will define two cost functions in terms of easily computable HOS in the next section.

III. COST FUNCTIONS

A. Independence Conditions and Constraints

In order to measure the independence of the outputs, it is very natural to use the probability criterion of random variables. Stochastic independence of random variables is defined from its probability distribution and the concept of the statistical independence is more general than decorrelation, which takes only second-order statistics into account. Let $p(y)$ be the probability density function (pdf) of random variable y , $p_{y_i}(y_i)$ be marginal pdf of y_i , and $p(\mathbf{y})$ be the joint pdf of random vector \mathbf{y} , we have the following conditions for the independence of random variables.

Independence Conditions: For the independent components of random vector \mathbf{y} , the following statements are mutually equivalent:

- 1) $p(\mathbf{y}) = \prod_{i=1}^n p_{y_i}(y_i)$;
- 2) $KL[p_y(\mathbf{y}), p(\mathbf{y})] = \int p_y(\mathbf{y}) \log(p_y(\mathbf{y})/p(\mathbf{y})) d\mathbf{y} = 0$;
- 3) $\int [p(\mathbf{y}) - \prod_{i=1}^n p_{y_i}(y_i)]^2 d\mathbf{y} = 0$.

Obviously, Condition 1 is the definition of independence of random variables. Condition 2 is just the famous KL divergence between two probability distribution $p_y(\mathbf{y})$ and $p(\mathbf{y}) = \prod_{i=1}^n p_{y_i}(y_i)$. According to the KL divergence, one can obtain the mutual information criterion of \mathbf{y} , which has been studied widely in the context of blind signal processing. It is easily shown that Condition 3 is a direct result of Conditions 1 and 2. Therefore, the three statements are sufficient and necessary conditions of the mutual independence of random variables; i.e., minimizing the KL divergence of \mathbf{y} or the left part of Statement 3 to zero can make the outputs mutually independent and vice versa.

Although the above conditions are necessary and sufficient for the independence of the outputs of the separation system, they need the estimation of the probability or entropy of the output variables and mixing input variables, which are unknown under the “blind” assumption. So, these conditions seem difficult to use. In order to avoid the difficulties in estimating the probability of unknown random variables, we wish to find an expression equivalent to the conditions, but is easier to deal with in terms of higher order statistical moments without estimating the probability and the entropy of the outputs.

For the convenience of our later discussions, recall the definition of the characteristic function of random variables. It is well

known that the characteristic function and the pdf are related as a Fourier transform pair [29], i.e.,

$$\Phi_{\mathbf{y}}(\mathbf{v}) = \int p_y(\mathbf{y}) \exp(-j\mathbf{v}^T \mathbf{y}) d\mathbf{y} = E[\exp(-j\mathbf{v}^T \mathbf{y})] \quad (8)$$

where $\mathbf{v} = [v_1, \dots, v_n]^T$ is a vector of variables in Fourier transform domain and $j = \sqrt{-1}$ stands for an imaginary unit. In particular, for the single random variable, we have

$$\Phi_{y_i}(v_i) = E[\exp(-jv_i y_i)]. \quad (9)$$

In what follows, we will use two methods to reexpress the independence conditions of the theorem. One uses the moment statistics of the nonlinear function transform of the outputs. The other uses the HOS of the outputs. In order to further facilitate learning, constrain the outputs, and achieve good separation results, we impose two conventional constraints on the outputs of the separation system

$$C_1 := \sum_{i=1}^n E(y_i)^2 = 0 \quad (10)$$

$$C_2 := \sum_{i=1}^n [E(y_i^2) - 1]^2 = 0. \quad (11)$$

Constraint (10) is very natural and can always be achieved by adjusting the bias of neurons in the output layer when the system is implemented in a neural network. Constraint (11) is used to limit the variance of the outputs to be unity, which is very useful in the context of blind signal processing according to many simulation studies. So, we will add these two constraints in the cost functions to be defined and call them basic constraints.

B. Dependence Measure by Nonlinear Function Moment

In view of (8) and (9), according to the probability theory on the independence of random variables, Condition 1 can be expressed equivalently by using the characteristic function as

$$\Phi_{\mathbf{y}}(\mathbf{v}) = \prod_{i=1}^n \Phi_{y_i}(v_i). \quad (12)$$

Since the characteristic function of a random vector is equal to the product of that of each component of the outputs when they are mutually independent, if we take the Taylor expansion at both sides of (12), cross moments must be zero. As a result, we get an idea that if all joint cross moments of the outputs are forced to zero, the independence of the outputs can be achieved. This means that we can get an expression of the dependence measure by using moments that are easier to cope with. Unfortunately, it is impossible to take the moments of the all orders into account since the exhaustive computation is infeasible. To amend it, one can use a nonlinear transform function of signals before computing its moments. In this way, we can indirectly take all the moments into account to separate nonlinear mixtures. Suppose we have n infinitely differentiable functions $h_1(\cdot), \dots, h_n(\cdot)$ and have the following Taylor expansion:

$$h_i(y_i) = \sum_{j=0}^{\infty} \frac{h_i^{(j)}(0)}{j!} y_i^j = \sum_{j=0}^{\infty} c_{j,i} y_i^j, \quad i = 1, \dots, n \quad (13)$$

where we define $c_{j,i} = h_i^{(j)}(0)/j!$ to simplify notations. If we take the central moment of quantity $\prod_{i=1}^n h_i(y_i)$, we can get

$$M(h_1(y_1), \dots, h_n(y_n)) = E \left(\prod_{i=1}^n h_i(y_i) \right) - \prod_{i=1}^n E(h_i(y_i)). \quad (14)$$

Substituting (13) into (14), we have

$$M(h_1(y_1), \dots, h_n(y_n)) = \sum_{j_1=0}^{\infty} \dots \sum_{j_n=0}^{\infty} c_{j_1,1} \dots c_{j_n,n} M(y_1^{j_1}, \dots, y_n^{j_n}). \quad (15)$$

If the components in \mathbf{y} are mutually independent, then we have $M(y_1^{j_1}, \dots, y_n^{j_n}) = 0$ according to (14). From (15), we know that $M(h_1(y_1), \dots, h_n(y_n))$ is null if all $M(y_1^{j_1}, \dots, y_n^{j_n}) = 0$ for $j_k = 1, \dots, \infty$, where $k = 1, \dots, n$. Hence, $M(h_1(y_1), \dots, h_n(y_n)) = 0$ is a necessary condition for y_1, \dots, y_n to be independent. Based on the above discussions, we can define the following cost function for BSS in case of two independent sources:

$$C(\theta) = \sum_{i=1}^n \sum_{j \neq i}^n M(h_i(y_1), h_j(y_2))^2 + \alpha C_1 + \beta C_2 \quad (16)$$

where θ is the parameter vector of the separation system and α and β are two positive constants weighting the dependence measure and the basic constraints. Even though this is only a necessary condition for the dependence of outputs, it is very easy to deal with and good separation results are often obtained by minimizing this cost function, as will be further illustrated by our simulation results.

C. Dependence Measure by Higher Order Moments

By using Statement 3 to measure the dependence of random variables, we can get another expression of dependence by the higher order moments of the outputs \mathbf{y} . Before we proceed, a smoothing window should be enforced on the difference of the joint pdf and product of marginal pdf of \mathbf{y} for reducing the fluctuations on the coefficients of the expression and decreasing the errors of real data. Hence, the measure of the dependence of random variables becomes

$$D(\mathbf{y}) = \left\| p_{\mathbf{y}}(\mathbf{y}) - \prod_{i=1}^n p_{y_i}(y_i) \right\|_w^2 = \int \left\{ \left[p(\mathbf{y}) - \prod_{i=1}^n p_{y_i}(y_i) \right] * w(\mathbf{y}) \right\}^2 d\mathbf{y} \quad (17)$$

with

$$w(\mathbf{y}) = \prod_{i=1}^n w(y_i) \quad (18)$$

where $*$ in (17) denotes the convolution operator and $w(y_i)$ ($i = 1, \dots, n$) are window functions that may take different forms such as rectangular window function, up-cosine function, or Gaussian window function and can be selected according to specific applications. By considering (8), taking the Fourier trans-

form of (17), and considering the properties of the Fourier transforms, we can obtain

$$\int \left\| b f \Phi_{\mathbf{y}}(\mathbf{v}) - \prod_{i=1}^n \Phi_{y_i}(v_i) \right\|^2 W(\mathbf{v}) d\mathbf{v} \quad (19)$$

where $W(\mathbf{v})$ is the Fourier transform of the window function $w(\mathbf{y})$ defined in (18).

Since most signals considered are bounded and the moments of bounded random variables always exist, we can take the Taylor expansion around the origin of the characteristic function

$$\Phi_{\mathbf{y}}(\mathbf{v}) = \sum_{\alpha_1 \dots \alpha_n} \frac{1}{\alpha_1! \dots \alpha_n!} \frac{\partial^{\alpha_1 + \dots + \alpha_n} \Phi_{\mathbf{y}}(\mathbf{0})}{\partial v_1^{\alpha_1} \dots \partial v_n^{\alpha_n}} v_1^{\alpha_1} \dots v_n^{\alpha_n} \quad (20)$$

$$\Phi_{y_i}(v_i) = \sum_{\alpha_i=0}^{\infty} \frac{1}{\alpha_i!} \frac{\partial^{\alpha_i} \Phi_{y_i}(0)}{\partial v_i^{\alpha_i}} v_i^{\alpha_i}. \quad (21)$$

According to the convention that the partial derivative of order zero of a function is the function itself, we obtain

$$\Phi_{\mathbf{y}}(\mathbf{v}) - \prod_{i=1}^n \Phi_{y_i}(v_i) = \sum_{\alpha_1 \dots \alpha_n} Q_{\alpha_1 \dots \alpha_n} v_1^{\alpha_1} \dots v_n^{\alpha_n} \quad (22)$$

where

$$Q_{\alpha_1 \dots \alpha_n} = \frac{1}{\alpha_1! \dots \alpha_n!} \cdot \left\{ \frac{\partial^{\alpha_1 + \dots + \alpha_n} \Phi_{\mathbf{y}}(\mathbf{0})}{\partial v_1^{\alpha_1} \dots \partial v_n^{\alpha_n}} - \prod_{i=0}^n \frac{\partial^{\alpha_i} \Phi_{y_i}(0)}{\partial v_i^{\alpha_i}} \right\}. \quad (23)$$

According to the definitions of a characteristic function and the moment of random variables, we have the following relationship between the origin moment and the derivatives of the characteristic function at zero:

$$\frac{\partial^{\alpha_1 + \dots + \alpha_n} \Phi_{\mathbf{y}}(\mathbf{0})}{\partial v_1^{\alpha_1} \dots \partial v_n^{\alpha_n}} = (-j)^{\alpha_1 + \dots + \alpha_n} E[y_1^{\alpha_1} \dots y_n^{\alpha_n}] \quad (24)$$

$$\frac{\partial^{\alpha_i} \Phi_{y_i}(0)}{\partial v_i^{\alpha_i}} = (-j)^{\alpha_i} E[y_i^{\alpha_i}]. \quad (25)$$

Thus

$$Q_{\alpha_1 \dots \alpha_n} = \frac{1}{\alpha_1! \dots \alpha_n!} (-j)^{\alpha_1 + \dots + \alpha_n} M_{\alpha_1 \dots \alpha_n} \quad (26)$$

where $M_{\alpha_1 \dots \alpha_n}$ stands for the $(\alpha_1 + \dots + \alpha_n)$ th central moment of \mathbf{y} .

In view of (19)–(26), the expression of the measure of dependence can be rewritten as

$$\int \left| \sum_{\alpha_1 \dots \alpha_n} Q_{\alpha_1 \dots \alpha_n} v_1^{\alpha_1} \dots v_n^{\alpha_n} \right|^2 W(\mathbf{v}) d\mathbf{v} \quad (27)$$

which can also be transformed as

$$\int \sum_{\alpha_1 \dots \alpha_n} \sum_{\beta_1 \dots \beta_n} Q_{\alpha_1 \dots \alpha_n} Q_{\beta_1 \dots \beta_n}^* v_1^{\alpha_1 + \beta_1} \dots v_n^{\alpha_n + \beta_n} W(\mathbf{v}) d\mathbf{v}.$$

By interchanging the sequence of summation and integral, we obtain

$$\sum_{\alpha_1 \dots \alpha_n} \sum_{\beta_1 \dots \beta_n} Q_{\alpha_1 \dots \alpha_n} Q_{\beta_1 \dots \beta_n}^* \cdot \int v_1^{\alpha_1 + \beta_1} \dots v_n^{\alpha_n + \beta_n} W(\mathbf{v}) d\mathbf{v}. \quad (28)$$

Finally, we reach

$$\sum_{\alpha_1 \dots \alpha_n} \sum_{\beta_1 \dots \beta_n} d_W(\alpha_1, \dots, \alpha_n, \beta_1, \dots, \beta_n) \cdot M_{\alpha_1 \dots \alpha_n} M_{\beta_1 \dots \beta_n} \quad (29)$$

with

$$d_W(\alpha_1, \dots, \alpha_n, \beta_1, \dots, \beta_n) = (-j)^{\alpha_1 + \dots + \alpha_n - \beta_1 - \dots - \beta_n} \cdot R_W(\alpha_1, \dots, \alpha_n, \beta_1, \dots, \beta_n) \quad (30)$$

where

$$R_W(\alpha_1, \dots, \alpha_n, \beta_1, \dots, \beta_n) = \int v_1^{\alpha_1 + \beta_1} \dots v_n^{\alpha_n + \beta_n} W(\mathbf{v}) d\mathbf{v} \quad (31)$$

is the $(\alpha_1 + \beta_1 + \dots + \alpha_n + \beta_n)$ th moment of the transformed window function $W(\mathbf{v})$. Note that the positive constant in (28) is disregarded without any effect on our conclusion.

In this way, we have obtained the exact expression of the dependence measure in (17) in terms of the HOS of outputs, which is much easier to cope with. Because (29) is always non-negative, it becomes zero if and only if all the HOS $M_{\alpha_1 \dots \alpha_n}$ are equal to zero. It also means that the independence of the outputs has been achieved by dealing with the HOS of outputs only. In addition, according to the property and smoothing effect of the window functions, it is easily shown that the coefficient $d_W(\alpha_1, \dots, \alpha_n, \beta_1, \dots, \beta_n)$ decreases gradually to zero as the index of sum $\alpha_1 + \beta_1 + \dots + \alpha_n + \beta_n$ increases. This property allows us to use the finite sum of lower orders to approximate (29) with a satisfactory accuracy. For this purpose, we can define the following cost function:

$$C(\theta) = \sum_{\alpha_1 \dots \alpha_n} \sum_{\beta_1 \dots \beta_n} d_W(\alpha_1, \dots, \alpha_n, \beta_1, \dots, \beta_n) \cdot M_{\alpha_1 \dots \alpha_n} M_{\beta_1 \dots \beta_n} + \alpha C_1 + \beta C_2 \quad (32)$$

where the basic constraints are also included in the cost function to help obtain the correct separation in nonlinear mixtures. This criterion is exactly the same as the dependence criterion of outputs. So, it is also a sufficient and necessary condition for the independence of outputs. As a result, minimizing (32) will result in the independence of outputs and reaching the goal of BSS. However, the cost function in (32) includes the moments of the outputs in any order, which are impossible to realize in specific calculation. Therefore, in our specific implementation, we only take finite moments (up to K th moment) into account, then (32) becomes

$$C(\theta) = \sum_{\sum_i \alpha_i < K} \sum_{\sum_i \beta_i < K} d_W(\alpha_1, \dots, \alpha_n, \beta_1, \dots, \beta_n) \cdot M_{\alpha_1 \dots \alpha_n} M_{\beta_1 \dots \beta_n} + \alpha C_1 + \beta C_2. \quad (33)$$

By using the derivation results of (10), (11), (16), and (33), we approach our aforementioned goal of defining cost functions by HOS for BSS. The minimization of the cost function in (33) gives the correct separation results for linear or nonlinear mixtures in our parametric model. In the next section, we present the GA-based method to fulfill the search of the optimal parameter set of the separation model based on the cost functions specified in (16) and (33).

IV. GA-BASED SEPARATION APPROACH

A canonical GA is constituted by the operations of parameter encoding, population initialization, mate selection, crossover, mutation, and population replacement. Thus, the operations of a GA can be concisely described individually as follows.

Encoding: The parameters of the parametric separation system to be optimized are generally encoded into genes and chromosomes (also call individuals) as a string of binary digits using one's complement representation. Assume that the parameters lie in some bounded region

$$|\theta_k| \leq b_k \quad \text{for } k = 1, \dots, h \quad (34)$$

where h represents the number of the parameters. The length of the gene and individual can be calculated as the length of the binary string B_k needed to encode θ_k based on b_k and the desired accuracy. Other encoding methods are also possible. In [20]–[30], the value of each parameter is taken as a gene and, thus, the individual is encoded by a string of real numbers instead of binary numbers.

Initial Population Generation: The initial population is generated randomly in the range of each parameter. Therefore, at the beginning of the separating procedure, N individuals are generated as random binary strings.

Evaluation of Fitness: After the initial population generation, the fitness of each individual is determined. Fitness is a numeric index to measure the effectiveness of each individual of the population as a solution, which is usually utilized to select members of the population for reproduction. For the BSS problem, we can define a fitness function based on the defined cost functions in (16) or (33). As a result, the value of the fitness function F_i for i th individual I_i can be defined as

$$F_i(I_i) = \frac{1}{C(\hat{\theta}_i)} \quad (35)$$

where $C(\hat{\theta}_i)$ is the cost function with parameter set $\hat{\theta}_i$ to be determined, which is encoded as the i th individual. For this fitness function, though it is very sensitive to the change of individual $\hat{\theta}_i$, the value of the fitness near the global optimal solution approaches infinity. To amend this problem, we can add a minimum positive constant ϵ expressible by an actual digital computer to the denominator of (35), i.e., $F_i(I_i) = [C(\hat{\theta}_i) + \epsilon]^{-1}$.

Selection Operator: A pair of individuals is selected from the current population for mating using roulette wheel selection.

Crossover Operator: A multipoint crossover is applied to the newly selected (parents) individuals to generate two offspring. Specifically, the number of crossover points in our application is equal to the number of the parameters to be optimized.

TABLE I
COMPUTATIONAL COMPLEXITY OF THE GA-BASED BSS METHOD BASED ON THE COST FUNCTIONS DEFINED IN (16) AND (33)

| | |
|-------------------------------------|--|
| Complexity of cost function in (16) | multiplications: $(N_s + 1)n^2 + 1$ additions: $n^2 - n + 2N_s - 2$ |
| Complexity of cost function in (33) | multiplications: $(N_s - 1)[(K - n)(K - n + 1) + 2]/2 + 2^{(K-n)(K-n+1)/2} - 1$ additions: $(N_s - 1)(K - n)(K - n + 1)/2 + N_s/2 + 2^{(K-n)(K-n+1)/2}$ |
| Complexity of basic constraints | multiplications: $N_s + 2n + 2$ additions: $2N_s + 3n - 4$ |
| Complexity per function evaluation | multiplications: $N_p(N_s n^2 + n^2 + N_s + 2n + 4)$ additions: $N_p(n^2 + 2n + 4N_s - 5)$ |
| Number of function evaluation | $N_p N_g$ |

Mutation Operator: Random mutation operator is applied to the newly generated offspring to prevent from premature convergence. It randomly alters the gene from zero to one or from one to zero with a probability expressed by P_m , where P_m is called mutation probability.

For BSS, the convergence speed of the canonical GA was found to be very slow. An elitist strategy was used in which the best individual in the current generation always survived into the next generation to accelerate the convergence rate of the GA. We also adopted a reinitialization mechanism with the elitist strategy to explore a wider space besides the mutation operator.

In addition, in order to terminate the iteration of the GA optimization process, a stopping criterion was defined as no further improvement of the best fitness value in several consecutive generations (usually, three or four generations were chosen) so as to increase the chance of converging to a true global optimum solution.

In summary, the GA-based BSS algorithm can be implemented as the following iterative procedure.

- 1) An initial population $\{\hat{\theta}_i\}_{i=1}^N$ of size N is created from a random initial set of parameters. The encoding length of each parameter is 16 bits. By decoding the individual to get the parameter of the system, the fitness for each individual is evaluated using (35).
- 2) Two mates are selected for reproduction with probabilities proportional to their fitness using roulette wheel selection.
- 3) The multipoint crossover operator with crossover probability P_c is applied to the two mates and a pair of offspring are generated.
- 4) The mutation operator with probability P_m is applied to the newly generated offspring.
- 5) The fitness values for the offspring are computed after they are decoded as the parameter sets of the parametric system.
- 6) Steps 2–5 are repeated until an entirely new population of individuals is generated.
- 7) The previous population is replaced with the new population with the addition of an elitist selection.
- 8) If the stopping criterion is satisfied, go to step 11.
- 9) If generation number is greater than a predetermined value go to step 10, otherwise go to step 2.

10) Reinitialize the population with the best individual in the current population survival, go to step 2.

11) Output the individual with the best fitness value and terminate the iterative procedure.

The solution found by this GA-based BSS algorithm is used to construct the separation system, which is employed to separate the mixtures to get the original unknown sources. The convergence properties of the proposed GA-based approach are difficult to analyze because of the highly complex nature of the underlying optimization problem.

In order to evaluate the computational time, we coarsely summarized the computational complexity of the proposed GA-based BSS algorithm in Table I, where N_p denotes the population size, N_g is the number of generations to obtain convergence, n is the number of channels, and N_s is the data length of mixtures. When we derive the computational complexity, we take into account the computational requirements of the cost function and fitness evaluation only since the computational complexity of various genetic operators are generally negligible, compared with the computational complexity of cost function and fitness evaluation). Moreover, we count each nonlinear function transform as one multiplication.

Extensive experiments in the next section demonstrated the validity and effectiveness of the GA-based BSS approach. Because of the probabilistic nature of the GA-based method, the proposed method almost converges to a global optimal solution on average. In our simulations, however, we have not encountered any nonconvergent case.

V. SIMULATION RESULTS

To verify the validity and performance of the proposed algorithm, several computer simulations were conducted to test the GA-based approach to blind separation of independent sources from their nonlinear mixture.

Example 1: Consider the mixing case of two independent random signals—a random binary signal and a random signal with uniform distribution on interval $(-1, 1)$. The mixing matrices \mathbf{H} and \mathbf{A} are chosen randomly as

$$\mathbf{H} = \begin{pmatrix} 0.6744 & 0.3248 \\ 0.2461 & 0.9217 \end{pmatrix}, \quad \mathbf{A} = \begin{pmatrix} 0.9121 & 0.2292 \\ 0.4763 & 0.7348 \end{pmatrix}.$$

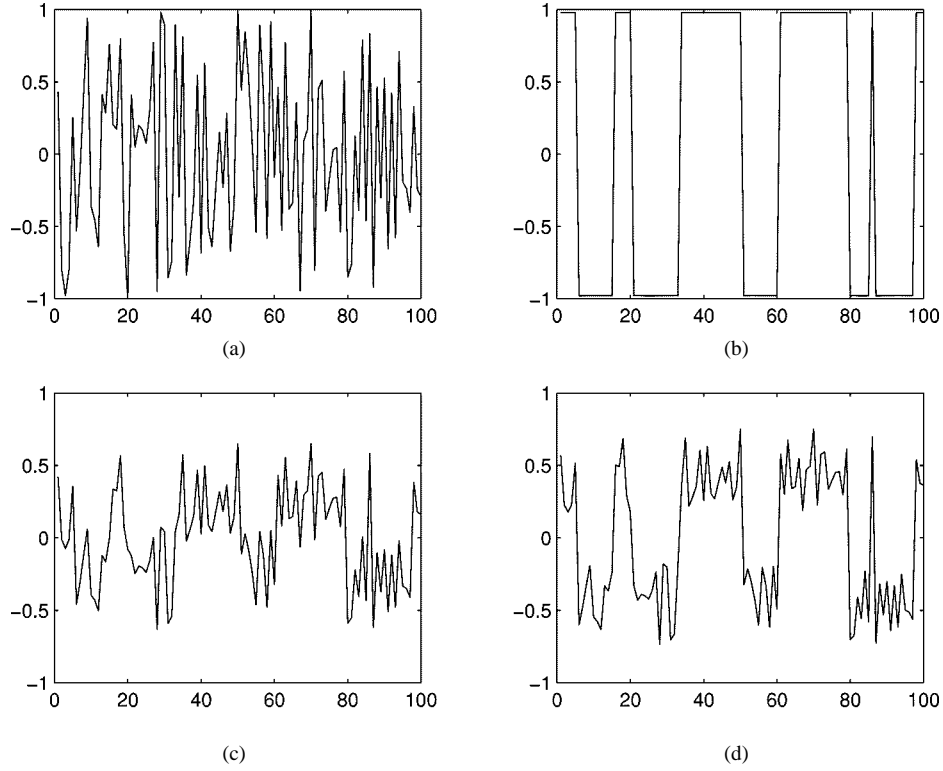


Fig. 2. Two source signals and their mixtures with inverse hyperbolic tangent nonlinearity. (a) One source signal $s_1(t)$ and (b) another source signal $s_2(t)$. (c) One mixture $x_1(t)$ of the two source signals with hyperbolic tangent transform function and (d) another mixture $x_2(t)$.

The nonlinear transform function in each channel is the hyperbolic tangent function with unity amplitude and half-unity gain in (36)

$$f_i(u_i) = -2\gamma_i \arctanh(-\delta_i u_i). \quad (36)$$

Fig. 2 depicts the two original sources and their mixtures.

The separation system adopts a two-layer neural network with piecewise-linear neurons in input and output layers in (6) and sigmoid neurons in hidden units as in (7). Specifically, the sigmoid activation functions are in the form of

$$g_i(v_i) = \alpha_i \frac{1 - \exp(-\beta_i v_i)}{1 + \exp(-\beta_i v_i)} \quad (37)$$

where $\alpha_i > 0$ and $\beta_i > 0$ are respectively the amplitude and gain of the i th neuron in the hidden layer. Thus, we can derive the corresponding mixing nonlinear transform in the left part of Fig. 1.

For this parametric separation system, we have 12 parameters (i.e., matrices \mathbf{B} and \mathbf{W} and amplitude α and gain β for two neurons in the hidden layer) to be tuned during the learning process. Because the amplitude parameter of the activation function can be absorbed into the demixing matrices, we can let $\alpha_i = 1$ without loss of generality. Thus, we have ten remaining parameters to determine.

We encoded each of the ten parameters as 16-bit binary string; each individual was represented by a 160-bit binary string. The population size was $N = 30$. The crossover probability $P_c = 0.99$ while mutation probability $P_m = 0.15$ by trial and error.

We also assume the maximum generation for GA process to be 30. If the stopping criterion is not satisfied, the GA process is restarted with the elitist strategy. To accelerate convergence, we adopt a restart strategy in this experiment.

The data length of the two sources are only 150 points in this experiment. When tuning the parameters, these data are fed into the separation system repeatedly until the GA-based learning process converges. All the results given here were obtained by averaging over 50 different runs. Each run used same mixture samples with a different randomly initialized population.

For the cost function of (16), we choose two sets of nonlinear functions. Case I is $f_1(x_1) = \tanh(x_1)$ and $f_2(x_2) = x_2^3$. Case II is $f_1(x_1) = \tanh(x_1)$ and $f_2(x_2) = \tanh(0.5x_2)$. The two cases give similar results. Figs. 3 and 4 plot the separation results and evolution curves based on the cost functions in the two cases.

For the cost function of (33), the order of the Taylor expansion is chosen as up to four, i.e., $K = 4$. We also employ a rectangular windowing function for the calculation of the cost function. Certainly, other type window functions such as Gaussian window and up-cosine window can also be adopted even though the selection of window functions are closely related to the application. Figs. 5 and 6 depict the evolution of the cost function in (33) and the separated signals by using our GA-based BSS approach with this cost function, respectively.

It can be seen from the simulation results and Figs. 3 and 6 that the GA-based BSS approach is promising and can give approximately global optimal solutions from any random initialization. The successful signal separation in case of nonlinear

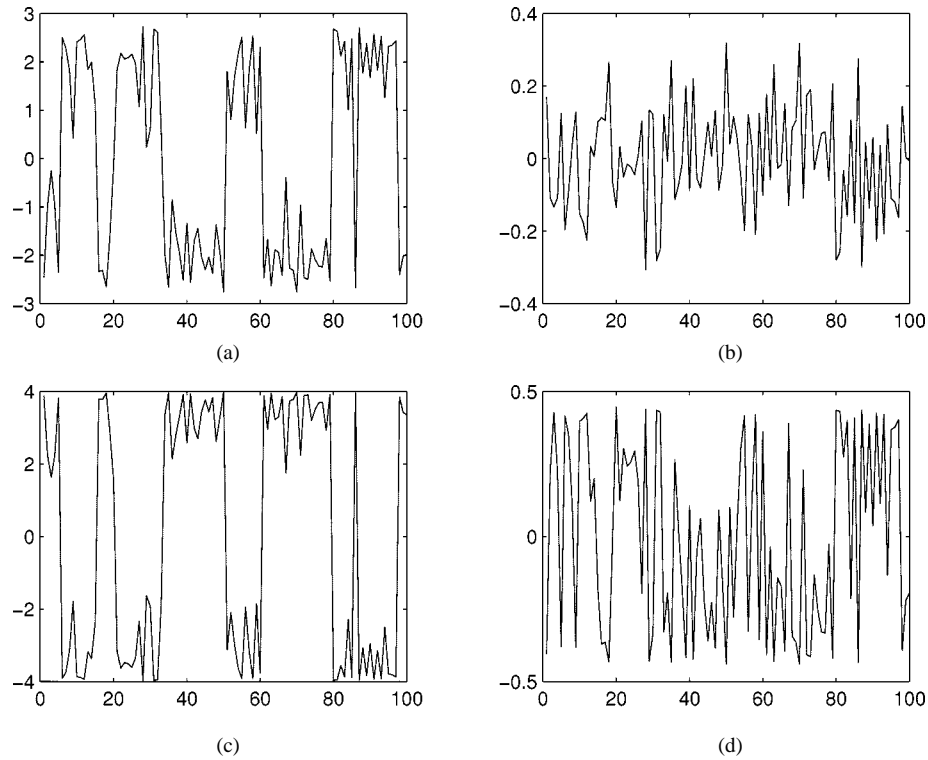


Fig. 3. Two separated signals of our method with two sets of nonlinear functions. (a) and (b) Case I. (c) and (d) Case II. For case I, nonlinear functions are chosen as $f_1(x_1) = \tanh(x_1)$ and $f_2(x_2) = x_2^3$. The obtained separated signals are shown in subplot (a) and subplot (b). In a similar way, for case II, nonlinear functions are chosen as $f_1(x_1) = \tanh(x_1)$ and $f_2(x_2) = \tanh(0.5x_2)$. The resulting separated signals are shown in subplot (c) and subplot (d).

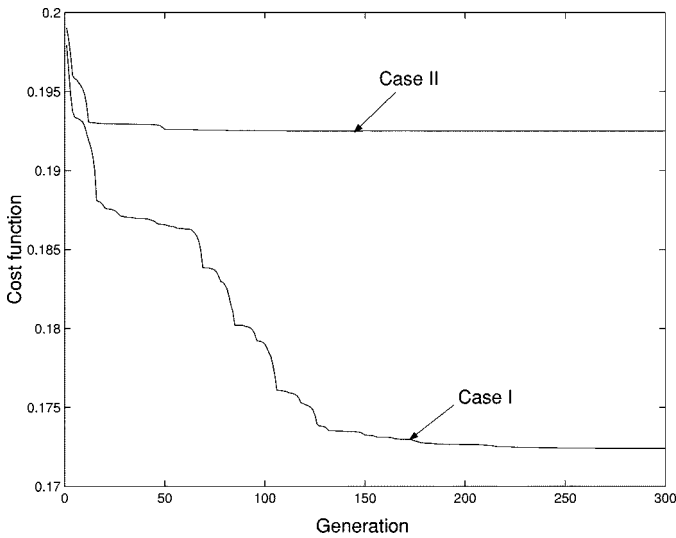


Fig. 4. Averaged evolution curves of the cost functions in (16) over 50 runs with different initializations for two sets of nonlinear functions: case I and case II.

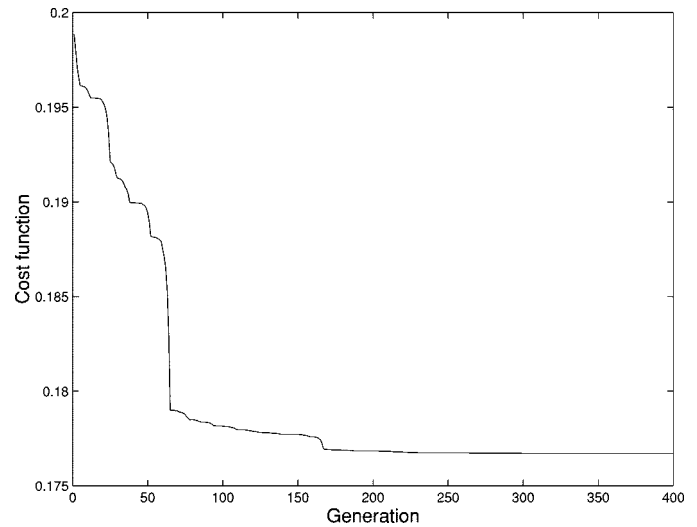


Fig. 5. Averaged evolution curve over 50 runs for the cost functions in (33) when $K = 4$.

mixtures are indeed achieved by our proposed GA-based approach.

At last, the simulation results in this example are summarized in Tables II and III. They clearly list the estimated mean and deviation of the parameters of the separation system over 50 runs with different random initializations for two cost functions in (16) and (33), respectively. Apart from the GA-based method, we also give two experimental computation results of a gra-

dient-based method for the two cost functions for comparison. It turns out that the gradient-based method fails to obtain the correct parameters of the separation system from different initializations due to its limited ability of local search and complex nonlinear characteristics of the problem. From Tables II and III, the GA-based method has the capability of blindly separating independent source signals from their nonlinear mixtures and has much better performance than the gradient-based method.

TABLE II
MEANS AND DEVIATIONS OF THE PARAMETERS IN THE SEPARATION SYSTEM OVER 50 RUNS FOR THE COST FUNCTION IN (16) WITH TWO CHOSEN NONLINEAR FUNCTIONS (CASE I AND CASE II) BY THE GA-BASED METHOD AND A GRADIENT-BASED METHOD IN EXAMPLE 1

| method | function | parameter | b_{11} | b_{12} | b_{21} | b_{22} | w_{11} | w_{12} | w_{21} | w_{22} | β_1 | β_2 |
|----------------|----------|-----------|----------|----------|----------|----------|----------|----------|----------|----------|-----------|-----------|
| GA | case I | mean | -1.3125 | -1.7500 | -0.8438 | -1.1250 | -0.7286 | 1.7499 | -2.0043 | -2.0106 | 1.0245 | 0.3221 |
| | | deviation | 0.0039 | 0.1363 | 0.0759 | 0.1664 | 0.1006 | 0.1419 | 0.0858 | 0.0609 | 0.0379 | 0.0387 |
| method | case II | mean | -1.2989 | -1.7392 | -0.8377 | -1.1220 | -0.7125 | 1.7697 | -1.9913 | -1.9804 | 0.7110 | 0.6612 |
| | | deviation | 0.1009 | 0.0844 | 0.1530 | 0.1397 | 0.1096 | 0.1391 | 0.1122 | 0.0581 | 0.0493 | 0.0580 |
| gradient based | case I | 1st run | 1.0989 | -1.6263 | -0.5529 | 0.7896 | 0.5956 | 1.9144 | 1.9133 | -1.8721 | 0.4449 | 0.6946 |
| | | 2nd run | -0.8155 | -0.9845 | -0.2079 | -0.7069 | -0.5999 | 2.1260 | -0.5534 | -2.9304 | 0.0118 | 0.8939 |
| method | case II | 1st run | -0.5543 | 1.5566 | -0.0492 | 0.5901 | -0.6366 | 1.2424 | 1.7821 | -0.7896 | 0.4156 | 0.0504 |
| | | 2nd run | 0.8407 | -0.6647 | 0.1764 | -0.8812 | 0.0252 | 0.6630 | -0.4411 | -0.8612 | 0.2726 | 0.2651 |

TABLE III
MEANS AND DEVIATIONS OF THE PARAMETERS IN THE SEPARATION SYSTEM OVER 50 RUNS FOR THE COST FUNCTION IN (33) WHEN $K = 4$ BY USING THE GA-BASED METHOD AND A GRADIENT-BASED METHOD IN EXAMPLE 1

| method | parameter | b_{11} | b_{12} | b_{21} | b_{22} | w_{11} | w_{12} | w_{21} | w_{22} | β_1 | β_2 |
|----------------|-----------|----------|----------|----------|----------|----------|----------|----------|----------|-----------|-----------|
| GA | mean | -1.2807 | -1.6892 | -0.8299 | -1.0621 | -0.6880 | 1.8303 | -1.9356 | -1.9754 | 0.9548 | 0.2856 |
| | deviation | 0.0736 | 0.1242 | 0.1463 | 0.0388 | 0.1810 | 0.1138 | 0.1264 | 0.0469 | 0.1098 | 0.1863 |
| gradient-based | 1st run | 0.3212 | -0.5007 | -0.4038 | 0.5152 | -0.4455 | 0.5432 | 0.9425 | -1.4682 | 0.4893 | 0.1859 |
| | 2nd run | 0.6698 | 0.5284 | -0.2446 | -0.5381 | -0.3833 | -0.8321 | -1.3328 | 0.6441 | 0.7006 | 0.9827 |

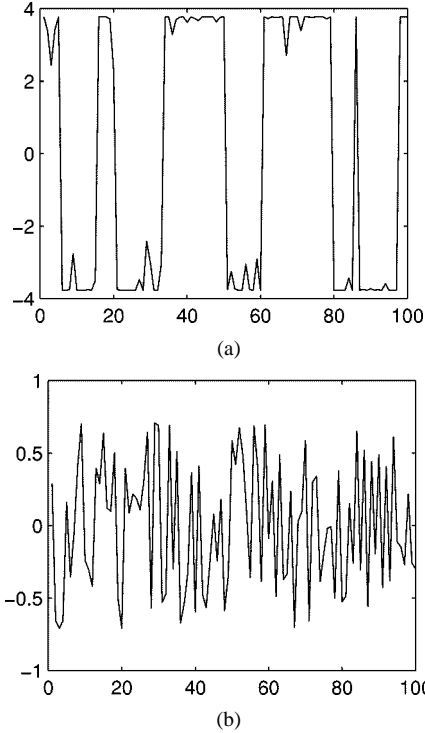


Fig. 6. Two separated signals by using the proposed GA-based method with the cost functions in (33) when $K = 4$. (a) $Y_1(t)$. (b) $Y_2(t)$.

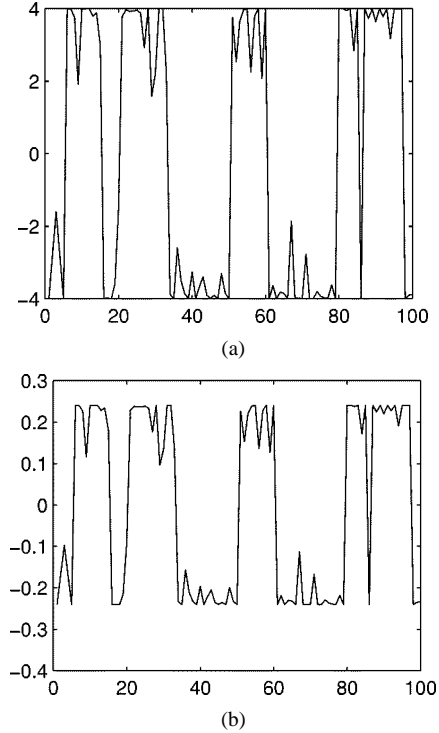


Fig. 7. (a) and (b) Two outputs give the same source signal, but the other one (b) does not appear at the outputs. A ghost solution appears when using the cost function by nonlinear function moments.

TABLE IV
COMPUTATIONAL COMPLEXITY OF THE PROPOSED GA-BASED BBS METHOD IN EXAMPLE 1

| | |
|-------------------------------------|--|
| Complexity of cost function in (16) | 605 multiplications 300 additions |
| Complexity of cost function in (33) | 659 multiplications 586 additions |
| Complexity of basic constraints | 156 multiplications 302 additions |
| Complexity per function evaluation | 22860 multiplications 17910 additions |
| Number of function evaluation | $30N_g$ |

From the simulation results, we know that the required generation for the GA-based BSS system to converge and get a good solution is typically a few hundred generations. Therefore, the computational advantage over conventional gradient methods is also obvious. Specifically, the computational complexity of this example is summarized in Table IV. It turns out from our simulation results that the performance of the cost function by HOS is slightly better than that of the cost function by nonlinear function but the computational complexity of the former is substantially higher than that of the latter. So, a compromise should be made between the two cases. If the accuracy and performance of a given problem is more important, the former is preferred,

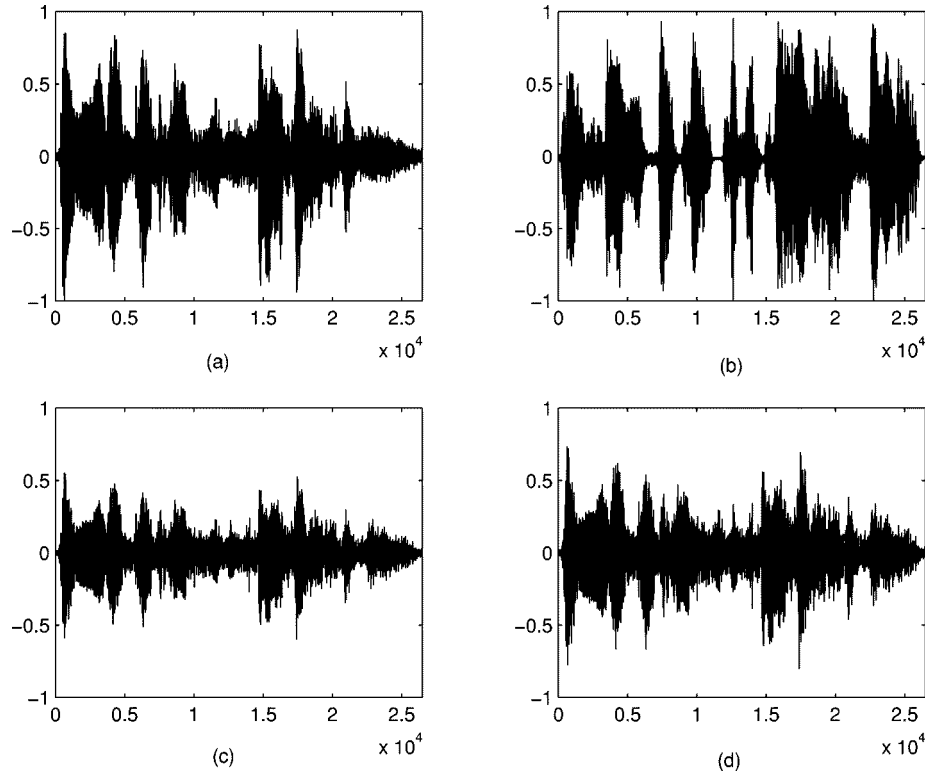


Fig. 8. Original speech signals of a woman and a man. (a) Woman's speech. (b) Man's speech. (c) and (d) Their mixtures.

otherwise the latter is desirable to save limited computational resources. In addition, in our simulation we also encounter a few ghost solutions when using the cost function (16) because it is only a necessary condition. A ghost solution is shown in Fig. 7(a) and (b), where two outputs give the same source signal, but the other one (b) does not appear at the outputs. No divergence occurs in our simulations using the cost function by HOS in (33).

Example 2: In order to further test the practical applicability of the proposed method, we consider a "cocktail party" problem. Two speakers, a man and a woman, are considered in this test. The sampling rate is 11.02 kHz and each sample is quantized as 8 bits. Their speeches are mixed by randomly mixing matrices and monotonically nonlinear transformation. The mixtures, also digitalized as 8 bits per point, are used as the inputs of the separation system. All data are normalized in the range of $[-1, +1]$ for the sake of computation convenience. Fig. 8 shows the original speeches to be recovered and their mixtures.

We use a two-layer neural network with sigmoid activation function for the neurons in a hidden layer but other neurons are linear. Also, there are ten decision parameters to be determined in this case. For the GA, we choose the population size $N = 30$, the crossover probability $P_c = 1.0$ while mutation probability $P_m = 0.25$. Similar to Example 1, a multiple-point crossover and elitist strategy are employed. There is no restart mechanism in this experiment. The evolution process of the cost functions are averaged on five runs. Fig. 9 shows the separated results by the proposed GA-based BSS approach based on the cost function defined in (33). To further illustrate the validity of our separation system, the cross correlations between the separated sig-

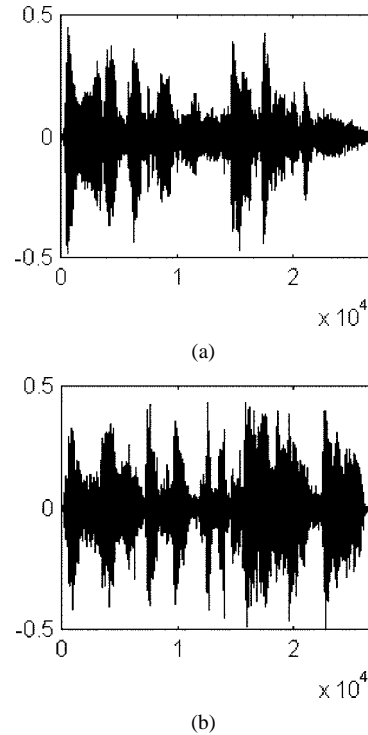


Fig. 9. Separated speech signals by using the proposed GA-based BSS approach based on the cost function in (33). (a) Woman's speech. (b) Man's speech.

nals with original speech signals and their mixtures are listed in Table V. The separating experiment based on the cost function in (16) is also carried out by using the proposed approach. To

TABLE V

CROSS CORRELATIONS BETWEEN THE SEPARATED SIGNALS AND CLEAR SPEECH SIGNALS AND THEIR MIXTURES TO ILLUSTRATE THE VALIDITY OF THE PROPOSED SEPARATION SYSTEM, WHERE s_1 AND s_2 ARE THE CLEAR SPEECH SIGNALS AND THEIR MIXTURES ARE DENOTED BY x_1 AND x_2 , y_1 AND y_2 ARE SEPARATED SIGNALS

| | s_1 | s_2 | x_1 | x_2 |
|-------|--------|-------------------------|--------|--------|
| y_1 | 0.0135 | 7.0832×10^{-4} | 0.0102 | 0.0079 |
| y_2 | 0.0012 | 0.0220 | 0.0043 | 0.0166 |

save space, the separated results are not shown due to its somewhat similarity to that in Fig. 9. It is shown from the graphs of Figs. 8 and 9 that our proposed method achieved the successful separation of the two speech signals from their nonlinear mixtures.

VI. CONCLUDING REMARKS

A GA-based BSS approach has been developed for BSS from the nonlinear mixtures of independent sources. The proposed method overcomes the local minima problem occurred in the conventional gradient-based methods and can obtain global optimal solutions to nonlinear BSS problem from any initial conditions. Extensive simulation results demonstrate the validity and performance of this GA-based BSS method. Apart from obtaining globally optimal separations, the proposed GA-based method is also characterized by high accuracy, high robustness, and fast convergence. Since the number of parameters to be optimized in BSS is usually small, the GA-based method is very suitable for this kind of problem. Another feature of this method is that it also suits the case of short available data. This is particularly useful in some medical applications where source signals may appear in a very short time period.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers and editors for their constructive and invaluable suggestions to help improving the presentation of this paper.

REFERENCES

- [1] J. Cardoso, "Blind signal separation: statistical principles," *Proc. IEEE*, vol. 86, pp. 2009–2025, Oct. 1998.
- [2] S. Amari and A. Cichocki, "Adaptive blind signal processing—Neural network approach," *Proc. IEEE*, vol. 86, pp. 2026–2048, Oct. 1998.
- [3] S. Makeig, A. Bell, T.-P. Jung, and T. J. Sejnowski *et al.*, "Independent component analysis in electro-encephalographic data," in *Advances in Neural Information Processing Systems*, M. Mozer *et al.*, Eds. Cambridge, MA: MIT Press, 1996, vol. 8, pp. 145–151.
- [4] C. B. Papadimas and A. Paulraj, "A constant modulus algorithm for multi-user signal separation in presence of delay spread using antenna arrays," *IEEE Signal Processing Lett.*, vol. 4, pp. 178–181, June 1997.
- [5] A.-J. van der Veen, S. Talvar, and A. Paulraj, "A subspace approach to blind space-time signal processing for wireless communication systems," *IEEE Trans. Signal Processing*, vol. 45, pp. 173–190, Jan. 1997.
- [6] J. Herault and C. Jutten, "Space or time adaptive signal processing by neural network models," in *Proc. AIP Conf.*, Snowbird, UT, 1986, pp. 206–211.
- [7] C. Jutten and J. Herault, "Blind separation of sources, Part I: An adaptive algorithm based on neuromimetic architecture," *Signal Processing*, vol. 24, pp. 1–20, 1991.
- [8] P. Comon, "Independent component analysis—a new concept?," *Signal Processing*, vol. 36, pp. 287–314, 1994.
- [9] S. Li and T. J. Sejnowski, "Adaptive separation of mixed sound sources with delays by a beamforming Herault-Jutten network," *IEEE J. Oceanic Eng.*, vol. 20, pp. 73–79, Jan. 1995.
- [10] U. A. Lindgren and H. Broman, "Source separation using a criterion based on second-order statistics," *IEEE Trans. Signal Processing*, vol. 46, pp. 1837–1850, July 1998.
- [11] G. Burel, "Blind separation of sources: a nonlinear neural algorithm," *Neural Netw.*, vol. 5, pp. 937–947, 1992.
- [12] Y. Tan, J. Wang, and J. M. Zurada, "Nonlinear blind source separation using a radial basis function network," *IEEE Trans. Neural Networks*, vol. 12, pp. 134–144, Jan. 2001.
- [13] A. Hyvarinen and P. Pajunen, "Nonlinear independent component analysis: Existence and uniqueness results," *Neural Netw.*, vol. 12, pp. 429–439, 1999.
- [14] G. Deco and W. Brauer, "Nonlinear higher order statistical decorrelation by volume-conserving neural architectures," *Neural Netw.*, vol. 8, pp. 525–535, 1995.
- [15] P. Pajunen, A. Hyvarinen, and J. Karhunen, "Nonlinear blind source separation by self-organizing maps," in *Progress in Neural Information Processing: Proceedings of the International Conference on Neural Information Processing*. New York: Springer-Verlag, 1996, vol. 2, pp. 1207–1210.
- [16] A. Taleb, C. Jutten, and S. Olympeff, "Source separation in post nonlinear mixtures: an entropy-based algorithm," in *Proc. Eur. Symp. Artificial Neural Networks*, 1998, pp. 2089–2092.
- [17] H. H. Yang, S. Amari, and A. Cichocki, "Information back-propagation for blind separation of sources from nonlinear mixture," in *Proc. IEEE Int. Conf. Neural Networks*, 1997, pp. 2141–2146.
- [18] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI: Univ. Michigan Press, 1975.
- [19] D. E. Goldberg, *Genetic Algorithms in Search Optimization, and Machine Learning*. Reading, MA: Addison-Wesley, 1989.
- [20] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*. Berlin, Germany: Springer-Verlag, 1992.
- [21] A. J. Chipperfield and P. J. Fleming, "Parallel genetic algorithms: A survey," Univ. Sheffield, Sheffield, U.K., ACSE Res. Rep. 518, 1994.
- [22] L. Yao and W. Sethares, "Nonlinear parameter estimation via the genetic algorithm," *IEEE Trans. Signal Processing*, vol. 42, pp. 927–935, Apr. 1994.
- [23] J. F. Miller *et al.*, "Digital filter design at gate-level using evolutionary algorithms," in *Proceedings of the 1st Genetic and Evolutionary Computation Conference*, vol. 2, W. Banzhaf *et al.*, Eds., San Mateo, CA, 1999, pp. 1127–1134.
- [24] S. Chen, Y. Wu, and S. McLaughlin, "Genetic algorithm optimization for blind channel identification with higher order cumulant fitting," *IEEE Trans. Evol. Comput.*, vol. 1, pp. 259–265, Nov. 1997.
- [25] S. Chen and Y. Wu, "Genetic algorithm optimization for maximum likelihood joint channel and data estimation," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, vol. 3, May 1998, pp. 2190–2195.
- [26] K. Krishnakumar, "Microgenetic algorithm optimization for stationary and nonstationary function optimization," in *Proc. SPIE Intelligent Control and Adaptive Systems*, vol. 1196, 1989, pp. 289–296.
- [27] M. A. Alkanhal and S. A. Alshebeili, "Blind identification of nonminimum phase FIR system: Cumulants matching via genetic algorithms," *Signal Processing*, vol. 67, pp. 25–34, 1998.
- [28] M. Yoshioka and S. Omatu, "Signal separation method using genetic algorithms," in *Proc. IEEE Int. Joint Conf. Neural Networks*, vol. 2, 1998, pp. 909–912.
- [29] A. Papoulis, *Probability, Random Variables, and Stochastic Process*, 2nd ed. New York: McGraw-Hill, 1984.
- [30] S. Matwin, T. Szapiro, and K. Haigh, "Genetic algorithms approach to a negotiation support system," *IEEE Trans. Syst., Man, Cybern.*, vol. 21, pp. 102–114, Jan.–Feb. 1991.



Ying Tan (M'98) was born in Yingshan County, Sichuan Province, China, on September 5, 1964. He received the B.S. degree in electronic engineering from The University of Science and Technology of China, Hefei, China, in 1985, the M.S. degree in electronic engineering from Xidian University, Xijian, China, in 1988, and the Ph.D. degree in electronic engineering from Southeast University, Nanjing, China, in 1997.

He was a Teaching Assistant Professor from 1988 to 1989 and then became a Lecturer in 1990. In 1997, he became a Postdoctoral Research Fellow and then an Associate Professor with the Department of Electronic Engineering and Information Science, University of Science and Technology of China. He visited the Chinese University of Hong Kong in 1999. He has authored or coauthored more than 60 academic papers in refereed journals and conferences and several books and book chapters. His current research interests include neural network theory and its applications, computational intelligence, signal and information processing, pattern recognition, bioinformatics, statistical learning theory and analysis, and artificial intelligence.

Dr. Tan is a Member of the Institution of Electrical Engineers and a Senior Member of the China Institute of Electronics. He has received a number of academic and research achievement awards from universities and his country due to his outstanding contributions and distinguished works, including the 1998 Wong Kuan-cheng Postdoctoral Fellowship of the Chinese Academy of Science for his outstanding research contributions.



Jun Wang (S'89–M'90–SM'93) received the B.S. degree in electrical engineering and the M.S. degree in systems engineering from Dalian University of Technology, Dalian, China, in 1982 and 1985, respectively, and the Ph.D. degree in systems engineering from Case Western Reserve University, Cleveland, OH, in 1991.

He is currently a Professor and Director of the Computational Intelligence Laboratory of the Department of Automation and Computer-Aided Engineering, Chinese University of Hong Kong, Hong Kong, China. Previously, he was an Associate Professor with the University of North Dakota, Grand Forks. His current research interests include neural networks and their engineering applications.

Dr. Wang is an Associate Editor of the IEEE TRANSACTIONS ON NEURAL NETWORKS.