American Journal of Information Science and Computer Engineering
Vol. 3, No. 5, 2017, pp. 64-70
http://www.aiscience.org/journal/ajisce
ISSN: 2381-7488 (Print); ISSN: 2381-7496 (Online)

An Estimation of Distribution Algorithm Utilizing Opposition-Based Learning for Nonlinear Blind Sources Separation

Ying Gao[1, *], Waixi Li[2]

[1]Department of Computer Science and Technology, Guangzhou University, Guangzhou, P.R. of China
[2]School of Mechanical and Electric Engineering, Guangzhou University, Guangzhou, P.R. of China

Abstract

An estimation of distribution algorithm utilizing opposition-based learning is firstly proposed in this paper. In the proposed algorithm, opposite population is generated from the current population by calculating opposite numbers, and the best individuals in the population with the current population and the opposite population are selected to form the next population based on fitness values. Then, demixing system of the blind sources separation with post-nonlinear mixture is modeled using a multi-input multi-output wavelet neural network whose parameters can be determined under the criterion of independence of its outputs. A criterion of independence based on higher order moments is used to measure the statistical dependence of the outputs of the demixing system, and the proposed algorithm is utilized to minimize the criterion. Finally, the proposed algorithm is compared with the version of the original estimation of distribution algorithm on some well-known benchmarks, and used to a post-nonlinear blind sources separation problem with two independent random signals. The relative experimental results demonstrate that the algorithm outperforms the original estimation of distribution algorithm, and is effective for post-nonlinear blind source separation.

Keywords

Estimation of Distribution Algorithm, Oposition Based Learning, Nonlinear Blind Sources Separation

Received: March 29, 2017 / Accepted: July 20, 2017 / Published online: September 18, 2017

@ 2017 The Authors. Published by American Institute of Science. This Open Access article is under the CC BY license.
http://creativecommons.org/licenses/by/4.0/

1. Introduction

The problem of blind source separation (BSS) has been intensively studied during the last twenty years, mainly in the case of linear instantaneous mixtures, and more recently for linear convolutive mixtures. 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. Thought it is difficulty in separating independent sources from nonlinear mixtures, several effective models and methods were proposed recently for nonlinear BSS such as Jordi *et al.* [1], Leonardo *et al.* [2], Amit *et al.* [3], David *et al.* [4-5], Sprekeler *et al.* [6], Ehsandoust *et al.* [7-8]. These methods are mostly developed based on gradient-based optimization to avoid computing some unknown quantities in an unsupervised manner. Therefore, they are susceptible to the local minima problem during the learning process and are thus limited in many practical applications.

Evolutionary algorithms have been introduced to solve nonlinear optimization problems. Estimation of distribution algorithms (EDA) [9] are evolutionary algorithms that use probabilistic models to represent relevant information about the search space. The idea is to capture, in the form of probabilistic

* Corresponding author
E-mail address: falcongao@sina.com.cn (Ying Gao), liuwaixi@sina.com.cn (Waixi Li)

dependencies between the variables, information about promising areas of the search space that can be used to improve the search for better solutions. Machine learning techniques are used to learn the probabilistic models and sample new solutions from them. EDA have been widely applied in combinatorial optimization and continuous optimization domains. It has been proven that EDA has some special characteristics of concise concept, good global searching ability. The performance of an EDA highly depends on how well it estimates and samples the probability distribution. Much research in EDA focuses on different approaches to probabilistic modeling and sampling, such as directed graphical models (Bayesian networks) [10-12], undirected graphical model (Markov Random Field) [13-14], etc.

In this paper, an estimation of distribution algorithm utilizing opposition-based learning [15] is proposed, and is applied to post-nonlinear blind source separation. The proposed algorithm employs opposition-based learning for population initialization and next population generation in the classical estimation of distribution algorithm. Opposite population is generated from the current population by calculating opposite numbers. Population with the current population and the opposite population is sorted based on fitness values, and the best individuals in the population are selected to form the next population. Then, demixing system of the post-nonlinear blind sources separation is modeled using a multi-input multi-output wavelet neural network whose parameters can be determined under the criterion of independence of its outputs. A criterion of independence based on higher order moments is used to measure the statistical dependence of the outputs of the demixing system, and the estimation of distribution algorithm utilizing opposition-based learning is utilized to minimize the criterion. The proposed algorithm is compared with the version of the original estimation of distribution algorithm on some well-known benchmarks, and used to a post-nonlinear blind sources mixture with two independent random signals. The relative experimental results demonstrate that the algorithm outperforms the original estimation of distribution algorithm, and is effective for post-nonlinear blind source separation.

2. EDA Utilizing Opposition-Based Learning

2.1. Estimation of Distribution Algorithm

Estimation of distribution algorithms differ from traditional evolutionary algorithms. Instead of applying genetic operators like mutation and crossover to the parents, it estimates a probability distribution over the search space, and then samples the offspring individuals from this distribution. Let P(t) be the population of solutions at generation t. EDA work in the following iterative way:

Step 1: Selection. Select M promising solutions from P(t) to form the parent set Q(t) by a selection method (e.g., truncation selection);

Step 2: Modeling. Build a probabilistic model p(x) based on the statistical information extracted from the solutions in Q(t);

Step 3: Sampling. Sample new solutions according to the constructed probabilistic model p(x);

Step 4: Replacement. Fully or partly replace solutions in P(t) by the sampled new solutions to form a new population P(t+1);

One of the major issues in estimation of distribution algorithms is how to select parents. A widely-used selection method in estimation of distribution algorithms is the truncation selection. In the truncation selection, individuals are sorted according to their objective function values. Only the best individuals are selected as parents.

Another major issue in estimation of distribution algorithms is how to build a probability distribution model p(x). In estimation of distribution algorithms for the global continuous optimization problem, the probabilistic model p(x) can be a Gaussian distribution model [16], a Gaussian mixture [9, 17], a histogram [18].

In Gaussian distribution model with diagonal covariance matrix, the joint density function of the k-th generation is written as follows:

$$p_t(\mathbf{x}) = \prod_{d=1}^{n} N(x_d; \mu_d(t), \sigma_d(t)) \tag{1}$$

$$N(x_d;\mu_d(t),\sigma_d(t)) = \frac{1}{\sqrt{2\pi\sigma_d(t)}} exp\left(\frac{1}{2}(\frac{x_d-\mu_d(t)}{\sigma_d(t)})^2\right) \tag{2}$$

In (1), the n-dimensional joint probability distribution is factorized as a product of n univariate and independent normal distributions. There are two parameters for each variable required to be estimated in the t-th generation: the mean, $\mu_d(t)$, and the standard deviation, $\sigma_d(t)$. They can be estimated as follows:

$$\hat{\mu}_d(t) = \frac{1}{M}\sum_{i=1}^{M} x_{i,d}(t) \tag{3}$$

$$\hat{\sigma}_d(t) = \sqrt{\frac{1}{M}\sum_{i=1}^{M}(x_{i,d}(t) - \hat{\mu}_d(t))} \tag{4}$$

where $d = 1,2,..., n$, $(x_{i,1}(t), x_{i,2}(t),..., x_{i,n}(t))$ are values of the i-th variable of the selected M parent solutions in the t-th generation.

EDA perform better than some other evolutionary algorithms due to its ability to utilize the global statistical information collected from the previous search.

2.2. Opposition-Based Learning

The scheme of opposition-based learning was first introduced by H. R. Tizhoosh [15]. The opposition-based learning is general enough and can be utilized in a wide range of learning and optimization fields to make algorithms faster. Opposite numbers are defined as follows:

Let $x \in [a, b]$, then the opposite number x' is defined as

$$x' = a + b - x \tag{5}$$

The above definition can be extended to higher dimensions as follows:

Let $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ be an n-dimensional vector, where $x_i \in [a_i, b_i], i = 1, \ldots, n$. Opposite vector of $\mathbf{x} = (x_1, \ldots, x_n)$ is defined by $\mathbf{x}' = (x_1', \ldots, x_n')$ where $x_i' = a_i + b_i - x_i$,

Assume $f(\mathrm{x})$ is a fitness function which is used to measure candidate's optimality. According to the opposite point definition, $\mathbf{x}' = (x_1', \ldots, x_n')$ is the opposite of $\mathbf{x} = (x_1, \ldots, x_n)$. Now, if $f(\mathbf{x}') \geq f(\mathbf{x})$, then point x can be replaced with $\mathbf{x}'$; otherwise we continue with x. Hence, the point and its opposite point are evaluated simultaneously to continue with the fitter one [15].

2.3. EDA utilizing Opposition-Based Learning

There are two important steps in estimation of distribution algorithms which are explicitly modeling the probability distribution of the good solutions found so far and sampling new solutions by use of the constructed model. We incorporate the opposition-based learning mechanism into the estimation of distribution algorithms in order to improve the convergence performances of estimation of distribution algorithms. The opposite numbers are used in population initialization and also for generating new populations during the evolutionary process. We consider the following unconstrained global optimization problem: $\min f(\mathbf{x})$, $\mathbf{x} = (x_1, \ldots, x_n)$, and $f(\mathbf{x})$ is a continuous real value function defined on the hypercube $D = [a_1, b_1] \times \ldots \times [a_n, b_n]$. n is dimensional number of search space. Estimation of distribution algorithms using opposition-based learning as follows:

Step 1 Initialize a population Q(0) randomly and calculate opposite population $\mathrm{Q}'(0)$ by the opposite vector;

Step 2 Select the fittest individuals from the initial population Q(0) and the opposite population $\mathrm{Q}'(0)$ as the initial population P(0);

Step 3 Build a probabilistic model $p(\mathrm{x})$ based on the statistical information extracted from the individuals in P(t);

Step 4 Sample individuals according to the constructed probabilistic model $p(\mathrm{x})$ to form population Q(t+1);

Step 5 Calculate opposite population $\mathrm{Q}'(t+1)$ by the opposite vector;

Step 6 Select the fittest individuals from the set $\{\mathrm{Q}(t+1) \cup \mathrm{Q}'(t+1)\}$ as the next generation population P(t+1);

Step 7 If the given stopping condition is not met, goto Step 3;

The pseudocode for EDA using opposition-based learning is presented as follows:

Initialize a population Q(0) randomly

FOR i=1 to N, d=1 to n

Initialize individual $x_{i,d}$ randomly within $[a_d, b_d]$;

END FOR

Calculate opposite population $\mathrm{Q}'(0)$ by the opposite vector

FOR i=1 to N, d=1 to n

$$x_{i,d}' = a_d + b_d - x_{i,d}$$

END FOR

Select N fittest individuals from $\{\mathrm{Q}(0) \cup \mathrm{Q}'(0)\}$ as initial population P(0)

WHILE stop criteria are not attained DO

FOR d=1 to n

$$\mu_d(t) = \frac{1}{N}\sum_{i=1}^{N} x_{i,d}(t), \quad \sigma_d(t) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_{i,d}(t) - \mu_d(t))}$$

END FOR

$$N(x_d; \mu_d(t), \sigma_d(t)) = \frac{1}{\sqrt{2\pi\sigma_d(t)}} exp\left(\frac{1}{2}(\frac{x_d - \mu_d(t)}{\sigma_d(t)})^2\right)$$

$$p_t(\mathbf{x}) = \prod_{d=1}^{n} N(x_d; \mu_d(t), \sigma_d(t))$$

Generate population Q(t+1)

FOR i=1 to N, d=1 to n

$x_{i,d}(t+1)$ is sampled according to $N(x_d; \mu_d(t), \sigma_d(t))$

END FOR

Generate opposite population $Q'(t+1)$

FOR *i*=1 to N, *d*=1 to n

$$x'_{i,d}(t+1) = a_d + b_d - x_{i,d}(t+1)$$

END FOR

Select N fittest individuals from $\{Q(t+1) \cup Q'(t+1)\}$ as P(*t*+1)

END WHILE

3. EDA utilizing Opposition-Based Learning for Nonlinear BSS

A generic nonlinear mixture model for blind source separation can be described as

$$\mathbf{x}(t) = \mathbf{F}(\mathbf{s}(t)) \tag{6}$$

Where

$$x_i(t) = F_i(s_1(t), s_2(t), \cdots, s_n(t)), \quad i = 1,2,\cdots,n \tag{7}$$

$$\mathbf{F} = [F_1, F_2, \cdots, F_n]^{\mathrm{T}} \tag{8}$$

$$\mathbf{s}(t) = [s_1(t) \quad s_2(t) \quad \cdots \quad s_n(t)]^{\mathrm{T}} \tag{9}$$

$$\mathbf{x}(t) = [x_1(t) \quad x_2(t) \quad \cdots \quad x_n(t)]^{\mathrm{T}} \tag{10}$$

(9) called the independent source vector, and (10) called vector of observed random variables.

Especially, the post-nonlinear mixture model for blind source separation can be described as:

$$\mathbf{x}(t) = \mathbf{f}(\mathbf{A}\mathbf{s}(t)) \tag{11}$$

Where A is a mixing matrix.

$$\mathbf{f} = [f_1(\cdot), f_2(\cdot), \cdots, f_n(\cdot)]^{\mathrm{T}} \tag{12}$$

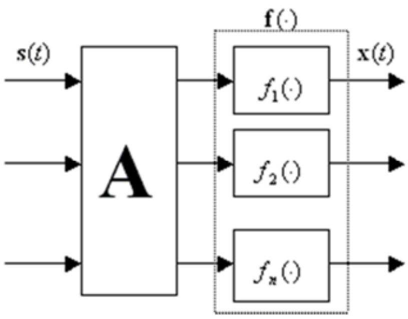

Figure 1. Post-nonlinear mixture model.

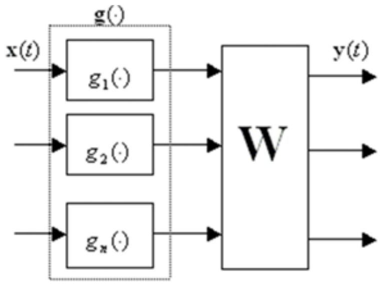

Figure 2. Separating system of post-nonlinear mixture model.

Figure 1 shows the post-nonlinear mixture model described in (11), and Figure 2 shows blind source separation system of post-nonlinear mixture model. According to Figure 1 and Figure 2, the output of the post-nonlinear separating system can been written as

$$\mathbf{y}(t) = \mathbf{W}\mathbf{g}(\mathbf{x}(t)) \tag{13}$$

Substituting (11) into (13), we can obtain $\mathbf{y}(t) = \mathbf{W}\mathbf{g}(\mathbf{f}(\mathbf{A}\mathbf{s}(t)))$, If $\mathbf{g}(\cdot) = \mathbf{f}^{-1}(\cdot)$ and $\mathbf{WA} = \mathbf{PD}$, then this means that the components of the outputs $\mathbf{y}$ are independent. Where $\mathbf{P}$ is a permutation matrix and $\mathbf{D}$ is a nonsingular and diagonal matrix.

For the nonlinear mixing transform function $\mathbf{f}$, we assume it has the inverse function $\mathbf{f}^{-1}$. $\mathbf{y}(t) = \mathbf{W}\mathbf{g}(\mathbf{x}(t))$ is a multi-input multi-output system, a wavelet neural network [19] of Figure 3 is used to approximate the unknown multi-input multi-output system.

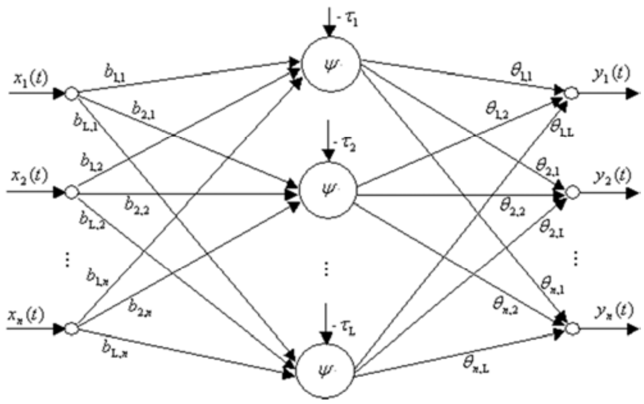

Figure 3. Wavelet neural network model.

Where $\mathbf{x}(t) \in \mathrm{R}^n$, $\mathbf{y}(t) \in \mathrm{R}^n$, $\psi(\mathbf{x})$ is an orthonormal wavelet basic function. The outputs of the wavelet neural network showed in Figure 3 can been written as

$$y_i(t) = \sum_{j=1}^{L} \theta_{i,j} \psi(b_{i1}x_1(t) + b_{i2}x_2(t) + \cdots + b_{in}x_n(t) - \tau_j)$$
$$i = 1,2,\cdots,n \tag{14}$$

(14) can be the following matrix form

$$\mathbf{y}(t) = \mathbf{\Theta\Psi}(\mathbf{Bx}(t) - \boldsymbol{\tau}) \quad (15)$$

Where

$$\mathbf{\Theta} = \begin{bmatrix} \theta_{1,1} & \theta_{1,2} & \cdots & \theta_{1,\mathrm{L}} \\ \theta_{2,1} & \theta_{2,2} & \cdots & \theta_{2,\mathrm{L}} \\ \vdots & \vdots & \ddots & \vdots \\ \theta_{n,1} & \theta_{n,2} & \cdots & \theta_{n,\mathrm{L}} \end{bmatrix} \quad (16)$$

$$\mathbf{B} = \begin{bmatrix} b_{1,1} & b_{1,2} & \cdots & b_{1,n} \\ b_{2,1} & b_{2,2} & \cdots & b_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ b_{\mathrm{L},1} & b_{\mathrm{L},2} & \cdots & b_{\mathrm{L},n} \end{bmatrix} \quad (17)$$

$$\boldsymbol{\tau} = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \vdots \\ \tau_{\mathrm{L}} \end{bmatrix} \quad (18)$$

$$\mathbf{\Psi} = \begin{bmatrix} \psi(b_{1,1}x_1(t) + b_{1,2}x_2(t) + \cdots + b_{1,n}x_n(t) - \tau_1) \\ \psi(b_{2,1}x_1(t) + b_{2,2}x_2(t) + \cdots + b_{2,n}x_n(t) - \tau_2) \\ \\ \psi(b_{\mathrm{L},1}x_1(t) + b_{\mathrm{L},2}x_2(t) + \cdots + b_{\mathrm{L},n}x_n(t) - \tau_{\mathrm{L}}) \end{bmatrix} \quad (19)$$

$b_{i,j}(i=1,\cdots,\mathrm{L}, j=1,\cdots,n)$ are scale factors, $\tau_j(j=1,\cdots,n)$ are location factors, $\theta_{i,j}(i=1,\cdots,n, j=1,\cdots,\mathrm{L})$ are network weights, L is the number of wavelet neuron, $\mathbf{H}$ is set of the parameters of wavelet neural network, and it is consists of $b_{i,j}(i=1,\cdots,\mathrm{L}, j=1,\cdots,n)$, $\tau_j(j=1,\cdots,n)$ and $\theta_{i,j}(i=1,\cdots,n, j=1,\cdots,\mathrm{L})$.

It is possible to recover the source from the post-nonlinear mixture (11) using only the source statistical independence assumption [20]. In order to separate the independent sources from their post nonlinear mixtures, the outputs of the separation system is expected to be mutually statistically independent. For this purpose, a measure of independence between random variables must be utilized. Indeed, if the y_i are independent, one has [21]

$$\mathrm{M}(h_1(y_i), h_2(y_j)) =$$

$$\mathrm{E}[h_1(y_i)h_2(y_j)] - \mathrm{E}[h_1(y_i)]\mathrm{E}[h_2(y_j)] = 0 \quad \forall i \neq j \quad (20)$$

Thus, a cost function for post nonlinear blind sources separation is defined:

$$J(\mathbf{H}) = \sum_{i=1}^{n} \sum_{j \neq i}^{n} [\mathrm{M}(h_1(y_i), h_2(y_j))]^2$$

$$= \sum_{i=1}^{n} \sum_{j \neq i}^{n} [\mathrm{E}(h_1(y_i)h_2(y_j)) - \mathrm{E}(h_1(y_i))\mathrm{E}(h_2(y_j))]^2 \quad (21)$$

Where $\mathbf{H}$ is set of the parameters of wavelet neural network.

The minimization of the cost function in (21) can give the correct separation results for post nonlinear mixtures. In this section, we present the particle swarm optimization-based method to fulfill the search of the optimal parameter set of the separation model based on the cost functions specified in (21). The EDA using opposition-based learning for post-nonlinear blind source separation can be implemented as the following iterative procedure:

1) Set the iteration number t to zero, an initial population Q(0) are created from a random initial set of parameters;
2) Calculate opposite population $\mathrm{Q}'(0)$ by the opposite vector;
3) Evaluate individuals in $\{\mathrm{Q}(0) \cup \mathrm{Q}'(0)\}$, and select N fittest individuals as initial population H(0);
4) Build Gaussian probabilistic distribution p(x) from the individuals in H(t) by estimating mean and standard deviation;
5) Generate population Q(t+1) by sampling Gaussian probabilistic distribution p(x);
6) Calculate opposite population $\mathrm{Q}'(t+1)$ by the opposite vector;
7) Evaluate individuals in $\{\mathrm{Q}(t+1) \cup \mathrm{Q}'(t+1)\}$, and select N fittest individuals as population H(t+1);
8) Let t=t+1;
9) Go to step 4), and repeat until convergence.
10) Output the individual with the best fitness value in H and compute the separated signals.

4. Computer Simulation Results

In the section, the performance of the estimation of distribution algorithm utilizing opposition-based learning (EDAOL) is compared with that of the original estimation of distribution algorithm (EDA). The following some well-known benchmark functions have been used to test.

$$f_1(\mathbf{x}) = \sum_{i=1}^{n} x_i^2, \; -100 \le x_i \le 100$$

$$f_2(\mathbf{x}) = \sum_{i=1}^{n} (x_i^2 - 10\cos(2\pi x_i) + 10), \; -5.12 \le x_i \le 5.12$$

$$f_3(\mathbf{x}) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1, \; -600 \le x_i \le 600$$

$$f_4(\mathbf{x}) = \sum_{i=1}^{n}\left(\sum_{j=1}^{i} x_j\right)^2, \ -100 \le x_i \le 100$$

$$f_5(\mathbf{x}) = \sum_{i=1}^{n}|x_i| + \prod_{i=1}^{n}|x_i|, \ -10 \le x_i \le 10$$

$$f_6(\mathbf{x}) = \frac{\sin^2\sqrt{x_1^2 + x_2^2} - 0.5}{[1 + 0.001(x_1^2 + x_2^2)]^2} + 0.5, \ -100 \le x_i \le 100$$

All test functions have a global minimum with a fitness value of 0. Population size N=100. All functions were implemented in 20 dimensions except for the two-dimensional function 6. The initial population was generated from a uniform distribution in the ranges specified below. All experiments were repeated for 50 runs. The maximum number of iterations is set to 1000 in each running. Tables 1 listed the mean fitness value and standard deviation of the best solutions averaged over 50 trails on f_1~f_6 functions. According to Table 1, It can be known that the proposed algorithm can reach a better solution with a quicker speed and a higher precision. Clearly, the algorithm outperforms the standard EDA for the benchmark functions.

Table 1. The best fitness values with the standard deviation for f_1~f_6 function.

	Algorithm	Dim	Mean	Standard Deviation
f_1	EDA	20	0.4649e-6	0.7988e-6
	EDAOL	20	0.8469e-10	0.7517e-10
f_2	EDA	20	107.9121	91.4011
	EDAOL	20	93.3420	80.9838
f_3	EDA	20	0.9043	0.9201
	EDAOL	20	0.1087	0.5073
f_4	EDA	20	89.0113	160.9315
	EDAOL	20	0.7079e-7	0.4893e-6
f_5	EDA	20	0.7820e-5	0.5387e-5
	EDAOL	20	0.2131e-5	0.2496e-5
f_6	EDA	2	0. 4515e-8	0.3873e-8
	EDAOL	2	0. 2487e-9	0.1426e-9

And a simulation was conducted to test the algorithm to blind separation of independent sources from their post-nonlinear mixture. Nonlinear function $h_1(\cdot)$ and $h_2(\cdot)$ in (21) is selected as $h_1(x) = x^3$, $h_2(x) = x - \tanh(x)$. Consider the mixing case of two independent random signals. The mixing matrix A was randomly generated, nonlinear function $f_1(x) = x^3$, $f_2(x) = x^3$. Sources signals $s_1(t) = 0.5*[1 + \sin(6\pi t)]\cos(100\pi t)$ and $s_2(t) = \sin(20\pi t)$ are shown in Figure 4, mixtures are shown in Figure 5, Figure 6 are separated signals.

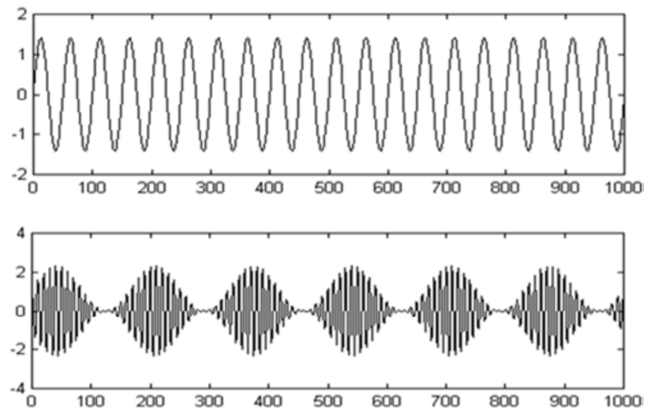

Figure 4. Source signals.

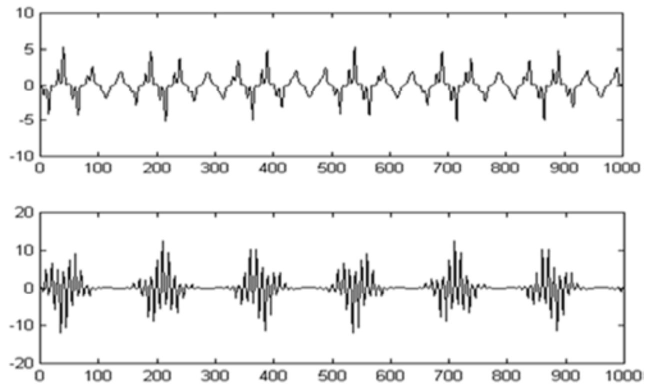

Figure 5. Mixture signals.

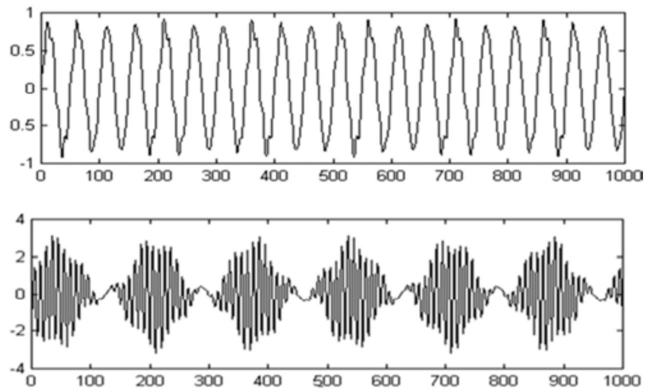

Figure 6. Separated signals.

5. Conclusions

In this paper, an estimation of distribution algorithm utilizing opposition-based learning and a method to blind source separation in post-nonlinear mixtures is presented. In the proposed algorithm, opposite population is generated from the current population by calculating opposite numbers, and the best individuals in the population with the current population and the opposite population are selected to form the next population based on fitness values. The demixing system of the post-nonlinear mixtures is modeled using a multi-input multi-output parameterized wavelet neural network whose parameters can be determined under the criterion of independence of its outputs. A criterion of independence based on higher order moments is used to measure the statistical dependence of the outputs of the demixing system, and the proposed algorithm is utilized to minimize the criterion. The proposed algorithm is compared with the version of the original estimation of distribution algorithm on some well-known benchmarks, and used to a post-nonlinear blind sources separation problem with two independent random signals. Simulation results show that the algorithm outperforms the original estimation of distribution algorithm, and is capable of separating independent sources from their post-nonlinear mixtures.

Acknowledgements

This work is supported by Natural Science Foundation of Guangdong Province, China, under Grant Nos. 2014A030313524 and 2014A030310349; by Science and Technology Projects of Guangdong Province, China, under Grant Nos. 2016B010127001 and 2015A010103020, and Science and Technology Projects of Guangzhou under Grant Nos. 201607010191 and 201604016045. Authors of the paper express great acknowledgment for these supports.

References

[1] Jordi Solé-Casals, Cesar F. Caiafa A Fast Gradient Approximation for Nonlinear Blind Signal Processing, Cognitive Computation, 2013, 5 (4): 483-492.

[2] Leonardo Tomazeli Duarte, Christian Jutten, SaÏd Moussaoui. A Bayesian Nonlinear Source Separation Method for Smart Ion-Selective Electrode Arrays. IEEE Sensors Journal. 2009, 9 (12), 1763-1771.

[3] Amit Singer, Ronald R. Coifman. Non-linear independent component analysis with diffusion maps. Applied and Computational Harmonic Analysis. 2008, 25 (2): 226-239.

[4] David N. Levin. Performing Nonlinear Blind Source Separation with Signal Invariants. IEEE Transactions on Signal Processing, 2010, 58 (4): 2131-2140.

[5] David N. Levin. Model-Independent Method of Nonlinear Blind Source Separation. International Conference on Latent Variable Analysis & Signal Separation, 2017: 310-319.

[6] H Sprekeler, T Zito, L Wiskott. An Extension of Slow Feature Analysis for Nonlinear Blind Source Separation, Journal of Machine Learning Research, 2014, 15 (3): 921-947.

[7] B Ehsandoust, M Babaie-Zadeh, C Jutten, Blind Source Separation in Nonlinear Mixture for Colored Sources Using Signal Derivatives, Springer International Publishing, 2015, 9237: 193-200.

[8] B Ehsandoust, B Rivet, C Jutten, M Babaie-Zadeh, Nonlinear blind source separation for sparse sources, European Signal Processing Conference, 2016: 1583- 1587.

[9] P. Larranraga, J. A. Lozano, Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation, Genetic Algorithms & Evolutionary Computation, 2002, 64 (5): 454-468.

[10] Pelikan, M., Hierarchical Bayesian Optimization Algorithm: Toward a New Generation of Evolutionary Algorithms, Springer, 2005.

[11] BV Ha, P Pirinoli, RE Zich, M Mussetta, F Grimaccia, Modified bayesian optimization algorithm for sparse linear antenna design, Progress in Electromagnetics Research B, 2013, 54 (54): 385-405.

[12] BV Ha, RE Zich, P Pirinoli, SV Hum, Application of Modified Bayesian Optimization Algorithm to the design of reflectarray antenna, International Conference on Numerical Electromagnetic Modeling & Optimization for Rf, 2014: 1-4.

[13] S. Shakya. DEUM: A Framework for an Estimation of Distribution Algorithm based on Markov Random Fields. PhD thesis, The Robert Gordon University, Aberdeen, UK, April 2006.

[14] X Hao, J Tian, HW Lin, T Murata, An Effective Markov Random Fields based Estimation of Distribution Algorithm and Scheduling of Flexible Job Shop Problem, Ieej Transactions on Electronics Information & Systems, 2014, 134 (6): 796-805

[15] Tizhoosh, H. R.: Opposition-Based Learning: A New Scheme for Machine Intelligence. In: Int. Conf. on Computational Intelligence for Modelling Control and Automation, Vienna, Austria, vol. I, pp. 695–701 (2005).

[16] PAN Bosman, J Grahl, D Thierens, Enhancing the Performance of Maximum–Likelihood Gaussian EDAs Using Anticipated Mean Shift, Lecture Notes in Computational Science & Engineering, 2008, 5199: 133-143.

[17] M Nakao, T Hiroyasu, M Miki, H Yokouchi, M Yoshimi, Real-coded Estimation of Distribution Algorithm by Using Probabilistic Models with Multiple Learning Rates, Procedia Computer Science, 2011, 4 (2): 1244-1251.

[18] S Shahraki, MRA Tutunchy, Continuous Gaussian Estimation of Distribution Algorithm, Springer Berlin Heidelberg, 2013, 8 (4): 772-775.

[19] A. Antonios K. Alexandridis, Achilleas D. Zapranis. Wavelet neural networks: A practical guide, Neural NetworksVolume 42, June 2013, Pages 1-27.

[20] Wang F S, Li H W, Shen Y T. An overview on nonlinear blind source separation: Theory and algorithms. Signal Processing. 2005, 21 (3): 282-288 (Chinese).

[21] Gao Ying Xie shengli A Blind Source Separation Using Particle Swarm Optimization. IEEE 6th CAS Workshop/Symposium on Emerging Technologies, Vol. 1, 2004, 297-300.