A Nonlinear Blind Source Separation System Using Particle Swarm Optimization Algorithm

Takuya Kurihara and Kenya Jin’no

Nippon Institute of Technology
4–1 Gakuendai, Miyashiro-machi, Minami-Saitama, Saitama 345-8501, Japan
E-mail: e1072215@estu.nit.ac.jp, jinno@nit.ac.jp

Abstract  Blind source separation (BSS) is a technique for recovering original source signals from mixed signals without the aid on information of the source signals. The system restores the original source signals using the probability of the distribution of the original signal. In this paper, we consider the case where the original source signals are nonlinearly mixed. In general, the separation of the nonlinear mixed signals is difficult. In order to solve this problem, we apply a radial basis function (RBF) network with the nonlinear BSS system. The RBF network can approximate the nonlinear mapping. Therefore, the inverse mapping of the nonlinear mixture system is approximated by the RBF network. For the system to approximate the inverse mapping, it is necessary to adjust the parameters of the RBF network. We assume the original source signals to be independent of each other. In this case, if the mixed signals can be separated, the higher-order cross-moment of the output signals is decreased. In order to adjust the parameters of the RBF network, particle swarm optimization is used. We confirm the separation performance by numerical simulations. Simulation results indicate that the proposed approach has good performance.

Keywords: nonlinear BSS, particle swarm optimization, RBF network, network structure, cross-moment

1. Introduction

Blind source separation (BSS) is a technique for recovering the original source signals from mixed signals without the aid on information on the source signals. BSS is a difficult problem. The main difficulty of BSS is its indetermination. In general, BSS searches for the solution in a limited region. One of the typical approaches of BSS is independent component analysis (ICA)[1]. The BSS technique and ICA method are applied to sound signal processing, electroencephalographic (EEG)[2], noise removal of the observed signal[3], and fault diagnosis[4], for example.

In ICA, each signal is assumed to be independent and to be non-Gaussian. ICA separates the signals by maximizing the statistical independence. Several techniques for maximizing statistical independence have been proposed, for example, maximization of the cumulant[5], minimization of nonlinear correlation[6], and maximization of negentropy[7].

In general, in BSS, the mixed signal is assumed to be composed of a linear combination of unknown independent signals. However, we consider the nonlinear mixed signal case. Such a problem is called nonlinear BSS. In general, the separation of nonlinear mixed signals by linear BSS is difficult[8]. Therefore, development of a nonlinear BSS method is required. There are some processing techniques for solving nonlinear BSS, for example, methods using a self-organizing map[9], multilayer perceptron[10], and radial basis function (RBF) network[11]. In addition, kernel ICA using a kernel trick has been proposed[12].

We demonstrate the separation of mixed source signals by fast ICA[5]. Fast ICA is one of the conventional linear ICA methods. Figure 1 shows the mixed source signals and the separated signals. In this case, the source signals are mixed by a linear operation. The result indicates that fast ICA can separate the source signals completely. On the other hand, Fig. 2 shows the case where the source signals are mixed by a nonlinear operation. In this case, the nonlinear operation is invertible, therefore, this nonlinearity can be classified into the easy case. However, fast ICA cannot separate the nonlinear mixed signals. Namely, the linear BSS system cannot separate the nonlinear mixed signals.
A nonlinear mixture model for BSS can be described as

$$x(t) = f(s(t))$$ \hspace{1cm} (1)$$

where \(x(t) = (x_1(t), x_2(t), \ldots, x_n(t))^T\) is the observed signal vector, \(s(t) = (s_1(t), s_2(t), \ldots, s_n(t))^T\) is the source signal vector generated from an independent source signal, superscript \(T\) denotes the transposition, and \(f(\cdot)\) is an unknown mixing function mapped from \(\mathbb{R}^n\) to \(\mathbb{R}^n\). We assume that \(f(\cdot)\) has an inverse function. Equation (1) corresponds to the left part of Fig. 3.

The right part of Fig. 3 corresponds to a separating function \(g(\cdot)\). A nonlinear separating system can be written as

$$y(t) = g(x(t), \theta)$$ \hspace{1cm} (2)$$

where \(y(t) = (y_1(t), y_2(t), \ldots, y_n(t))^T\) is a separated signal vector, \(g(\cdot)\) is a separating function mapped from \(\mathbb{R}^n\) to \(\mathbb{R}^n\), and \(\theta\) is a parameter vector of the separating function \(g(\cdot)\).

The purpose of nonlinear BSS is to search for the map \(g(\cdot)\). A typical solution is the inverse of the mixing function \(f(\cdot)\). However, we cannot calculate the inverse function because the mixing function is unknown. The ICA algorithm is one of the solutions for approximating the inverse function. In the ICA algorithm, each source signal is assumed to be independent of each other. The task is to find a map \(g(\cdot)\) that is independent of the output signals \(y(t)\). However, ICA cannot be used to determine the order or the power of the signal.

If the mapping is a linear transformation, ICA can solve the problem because there is a unique solution. On the other hand, the nonlinear mapping exists indefinitely if the solution space has not restriction[13].

### 3. Radial Basis Function Network

The RBF network approximates a nonlinear mapping[14]. Therefore, the inverse mapping of the nonlinear mixture system can be approximated by the RBF network. In this paper, we consider a nonlinear BSS system using the RBF network.

Figure 4 illustrates the \(n\)-input and \(n\)-output RBF network model employed for nonlinear BSS. The RBF network approximates a nonlinear mapping. In this work, we endeavor to design a nonlinear BSS system that can separate nonlinear mixed signals. However, dealing with all nonlinear mixing functions is very difficult. Therefore we classify the nonlinear mixing classes. As a first step in our study, we paid attention to an invertible nonlinear map. One of the nonlinear mixing classes is “post-nonlinear mixing” where in nonlinear distortion occurs after linear mixing[8]. Post-nonlinear mixing occurs in a sensor. We treat more general nonlinear mixing. For instance, there is a nonlinear distortion before and after linear mixing. This case corresponds to nonlinear distortions of the signal source and the sensor. However, sensors with irreversible characteristics also exist. Even in this case, separation of the signal is possible if the invertible map is within the range of the signal. We believe that such a system exists in reality.

The purpose of this paper is to propose nonlinear BSS using the particle swarm optimization (PSO) method. In our system, an RBF network is employed in the signal separation system. The RBF network can approximate nonlinear mapping. Therefore, the inverse mapping of the nonlinear mixture system is approximated by the RBF network. For the system to approximate the inverse mapping, it is necessary to adjust the parameters of the RBF network. We use the PSO algorithm to adjust parameters of the RBF network. The PSO algorithm is one of the heuristic optimization algorithms. Each particle moves in the solution space and searches for the optimal value. By using the numerical simulation results, we confirm the signal separation performance of our system.
network consists of three layers: an input layer, a hidden layer, and an output layer. The hidden layer employs an RBF unit. The output of the output layer is a linear combination of the RBF units.

The RBF network that maps from $\mathbb{R}^n$ to $\mathbb{R}^n$ is described by

$$y_i = \beta_i + \sum_{j=1}^{m} \alpha_{ij} K_j(x)$$  \hspace{1cm} (3)

where $K_j(x)$ denotes a radial function and $x$ represents the distance from the origin. $\alpha_{ij}$ is the weight coefficient between the $j$th radial function and the $i$th output, and $\beta_i$ is the $i$th offset.

We can select $K_j(x)$ from various functions. In this paper, we employ a Gaussian function as the function $K_j(x)$.

$$K_j(x) = \exp \left( -\frac{||x - \mu_j||^2}{\sigma_j^2} \right)$$  \hspace{1cm} (4)

$\mu_j$ denotes the $j$th center vector, $\sigma_j$ is a parameter of the $j$th RBF expansion. Other functions can be used as the RBF, namely, a linear function ($K(r) = r$), cubic function ($K(r) = r^3$), thin plate function ($K(r) = r^2 \log(r)$), multiquadric function ($K(r) = \sqrt{r^2 + 1}$), or inverse multiquadric function ($K(r) = 1/\sqrt{r^2 + 1}$)[11].

4. Particle Swarm Optimization

Searching for an optimal value of a given evaluation function in various problems is very important in engineering fields. In order to solve such optimization problems speedily, various heuristic optimization algorithms have been proposed. PSO is one of the metaheuristic algorithms for optimization problems[15][16]. Each particle has a location and velocity. Moreover, the particle memorizes its own best location. Each particle moves toward a combination of its own best location and the best location in the swarm. The information translation structure of particles can be regarded as a network structure. Several researchers have studied the relationship between the search performance and the network structure[17][18]. A ring structure is effective for searching for the optimum value of a multimodal function. Therefore, we pay attention to the ring structure of PSO in this paper.

The dynamics of the PSO algorithm is described as

$$\begin{align*}
v_{j}^{t+1} &= w v_{j}^{t} + c_1 r_1 (p_{best}^{t} - x_{j}^{t}) + c_2 r_2 (l_{best}^{t} - x_{j}^{t}) \\
x_{j}^{t+1} &= x_{j}^{t} + v_{j}^{t+1}
\end{align*}$$  \hspace{1cm} (5)

$x_{j}^{t} = (x_{j1}^{t}, x_{j2}^{t}, \ldots, x_{jN}^{t}) \in \mathbb{R}^N$ denotes the location of the $j$th particle at the $t$th iteration in the $N$-dimensional space, and $v_{j}^{t} = (v_{j1}^{t}, v_{j2}^{t}, \ldots, v_{jN}^{t}) \in \mathbb{R}^N$ denotes the velocity vector of the $j$th particle at the $t$th iteration. The PSO has three parameters, $w$, $c_1$, and $c_2$. $w \geq 0$ is an inertia weight coefficient, and $c_1 \geq 0$ and $c_2 \geq 0$ are acceleration coefficients. $r_1 \in [0, 1]$ and $r_2 \in [0, 1]$ are two separately generated uniformly distributed random numbers in the range $[0, 1]$.

$$p_{best}^{t} = (p_{best}^{t1}, p_{best}^{t2}, \ldots, p_{best}^{tN}) \in \mathbb{R}^N$$ represents the location that gives the best value of the evaluation function at the $t$th iteration. $p_{best}^{t}$ is the personal best. $p_{best}^{t}$ can be given as

$$p_{best}^{t} = x_{j}^{t}, \quad k = \arg \min_{r, \tau \leq t} f(x_{j}^{t})$$  \hspace{1cm} (6)

$$l_{best}^{t} = (l_{best}^{t1}, l_{best}^{t2}, \ldots, l_{best}^{tN}) \in \mathbb{R}^N$$ represents the location that gives the best value of the evaluation function at the $t$th iteration in the neighborhood of the $j$th particle. $l_{best}^{t}$ is the local best. $l_{best}^{t}$ can be given as

$$l_{best}^{t} = p_{best}^{t}, \quad l = \arg \min_{k \in Nr_j} f(p_{best}^{t})$$  \hspace{1cm} (7)

where $Nr_j$ is a set of particles in the neighborhood of the $j$th particle. The neighborhood is determined by the network topology of the particles. Figure 5 shows two network topologies. Figure 5(a) shows a full connection topology that corresponds to conventional PSO. Figure 5(b) shows the ring connection topology. Comparing these structures, the full connection exhibits quicker convergence. On the other hand, the ring structure exhibits an extraordinary search performance for a multimodal function[17][18].

5. Nonlinear BSS System

In this section, we describe our proposed nonlinear separation system. We assume that the signals have the following three properties. First, the mixing system is invertible. This is the condition to a separable
mapping exists. Second, the original source signals are independent of each other. This is a necessary condition for using ICA. Third, the time-averaged value of the original source signal is 0. ICA does not estimate the signal power. Hence, it is not possible to estimate the time average of the original source signal. Therefore, the system assumes that the time average of the original source signal is 0.

5.1 Separating system

Figure 6 shows our proposed nonlinear BSS system. The system consists of two parts: a whitening part and the RBF network. The parameters \( \sigma, \mu, \) and \( \alpha \) of the RBF network are adjusted using the PSO algorithm. \( \beta \) is set to a value the average value of the separation signal becomes 0.

In the whitening operation, the variance of the signal becomes 1 and the covariance becomes 0. This operation is described as

\[
\begin{align*}
  z &= \sqrt{V}^{-1} x \\
  V &= E[xx^T]
\end{align*}
\]  \hspace{1cm} (8)

where \( x \) is the original source signal vector and \( z \) denotes a whitened signal vector. Note that the average \( x \) is 0. \( V \) is a variance-covariance matrix of \( x \). \( \sqrt{V} \) satisfies \( \sqrt{V^T} \sqrt{V} = V \). If the original source signals are independent, the cross-correlation is 0. Therefore, the whitening operation is very important for ICA.

The whitening operation at the left part in Fig. 6 is a preprocess of signal \( x(t) \). It gives an appropriate initial value for the RBF network. The role of the whitening operation at the right part in Fig. 6 is to restrict the solution space.

5.2 Evaluation function

The cross-moment is employed to evaluate the learning. The cross-moment measures the independence of the output signals. We define an evaluation function using the cross-moment as

\[
C(y; \theta) = \sum_{i_1=1}^{k} \sum_{i_2=1}^{k} \sum_{i_n=1}^{k} (E [y_{i_1} \cdots y_{i_n}] - E [y_{i_1}] \cdots [y_{i_n}])^2 \hspace{1cm} (10)
\]

where \( y; \theta \) means that the generated signal \( y \) depends on the parameter vector \( \theta \). \( k \) denotes the maximum order for the evaluation.

On the basis of the above evaluation function, the system searches for the optimum parameters with which the evaluation value is 0. The reasons for this approach are as follows. If two stochastic variables \( x \) and \( y \) are independent, the following equation is satisfied:

\[
E[f(x)g(y)] - E[f(x)]E[g(y)] = 0 \hspace{1cm} (11)
\]

where \( f \) and \( g \) denote functions.

If the stochastic variables are independent, the value of Eq. (11) becomes 0. However, this condition is only a sufficient condition, therefore, if the value of Eq. (11) is 0, the stochastic values may not be independent. Thus, the summation of the higher-order cross-moments is applied to the evaluation function.

5.3 Adjustment algorithm

5.3.1 Particle swarm optimization

PSO searches for the minimum value of the evaluation function \( C(y) \) by updating \( a_{ij}, \mu_j, \) and \( \sigma_j \) of the RBF network. The initial values of the parameters are determined at random. After a certain number of iterations, the searching process is terminated.

PSO has the parameters \( c_1, c_2, \) and \( w \) in Eq. (5), where \( c_1 = c_2 = 1.49 \) and \( w = 0.730 \). These parameter values are recommended in Ref. [19].

5.3.2 Steepest descent method

As one of the algorithms for the adjustment of the parameters of the RBF network, we use the steepest descent method. This method is an algorithm using the first derivative the evaluation function \( C(y; \theta) \). The parameters of the RBF network are updated by

\[
\begin{align*}
  a_{ij} &\leftarrow a_{ij} - \eta \frac{\partial C(\theta)}{\partial a_{ij}} \\
  \sigma_j &\leftarrow \sigma_j - \eta \frac{\partial C(\theta)}{\partial \sigma_j} \\
  \mu_j &\leftarrow \mu_j - \eta \frac{\partial C(\theta)}{\partial \mu_j}
\end{align*}
\]  \hspace{1cm} (13)

where \( \eta \) is the learning coefficient. When the number of iterations reaches the maximum, the steepest descent method terminates the search.

6. Simulation

In this section, we carry out some numerical simulations to separate blind sources by the proposed procedure. In order to evaluate the separation ability, the
mean square error is applied, which is derived from two signals, $s(t)$ and $y(t)$.

$$mse = \frac{1}{n} \sum_{i=1}^{n} \min_{j} \text{error}_{i,j}$$

$$error_{i,j} = \min_{a} \mathbb{E}[(s_i - ay_j)^2]$$

$$= \mathbb{E}[s_i^2] - \frac{\mathbb{E}[s_i y_j^2]}{\mathbb{E}[y_j^2]}$$

6.1 Linear mixture case

We consider the following two-channel linear mixture system.

$$x_0(t) = B_0 s_0(t)$$

$$B_0 = \begin{bmatrix} 0.5 & 0.5 \\ 0.7 & 0.3 \end{bmatrix}$$

The original source signals $s_0(t)$ consist of a sinusoidal signal and a triangular waveform signal:

$$s_0(t) = \begin{bmatrix} \sin(2 \pi 90t) \\ \text{tri}(t, \tau) \end{bmatrix}$$

where the function $\text{tri}(t, \tau)$ denotes a periodic triangular wave with period $\tau = 0.002$. Figure 7(a) illustrates the original source signals $s(t)$, and Fig. 7(b) illustrates the mixed signals $x(t)$. We carry out numerical simulations by the standard PSO method, the Ring-PSO method and the steepest descent method to compare their performances. The standard PSO has the full-connection structure. For the numerical simulations, the same initial values are used. The simulation conditions are shown in Table 1. The number of particles is 40. $\eta$ for the steepest descent method is 0.0005. The maximum number of iterations for the steepest descent method is 500.

Figures 7(c), 7(d), and 7(e) respectively show the signals separated by the standard PSO, the Ring-PSO, and the steepest descent method. The evaluation value and mean square error of signals $s(t)$, $x(t)$, and each simulation result are shown in Table 2. The average and minimum evaluation values and mean square errors of each simulation are shown. In these cases, the separation of the signals was successful in all trials. These results indicate that the proposed method has the ability to separate linear mixed signals. However, the average evaluation function value of the steepest descent method is poor. This is because there are many local optima of the solution space. The performance is poor because the steepest descent method cannot escape from local solutions.

In order to confirm the performance of our proposed method, we carry out the following four simulations.

6.2 The case of a nonlinear mixture using a piecewise linear function

First, we consider the following two-channel nonlinear mixture system with a piecewise linear function.

$$x_1(t) = f(B_1 s(t))$$

$$B_1 = \begin{bmatrix} 0.7 & 0.3 \\ 0.3 & 0.1 \end{bmatrix}$$

$$f(x) = (f(x_1), \ldots, f(x_n))^T$$

$$f(x) = \begin{cases} 0.5x & 0 \leq x \\ x & x < 0 \end{cases}$$

$f(\cdot)$ is a piecewise linear function with different gradients in the positive region and negative region. $f(\cdot)$ is the vector function that maps each element of the vector. The piecewise linear function is applied to each element of the input vector. By using such a nonlinear mixture system, the mixed signals $x_1(t)$ is generated. The original source signals $s_1(t)$ consist of a sinusoidal signal and an amplitude-modulated signal:

$$s_1(t) = \begin{bmatrix} \sin(2 \pi 90t) \\ \sin(2 \pi 25t) \sin(2 \pi 800t) \end{bmatrix}$$

Table 1 Simulation parameters

<table>
<thead>
<tr>
<th>Number of trials</th>
<th>50</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum number of iterations for PSO</td>
<td>2000</td>
</tr>
<tr>
<td>Inertia weight coefficient $w$</td>
<td>$w = 0.730$</td>
</tr>
<tr>
<td>Acceleration coefficient $c_1, c_2$</td>
<td>$c_1 = c_2 = 1.49$</td>
</tr>
</tbody>
</table>

Table 2 Simulation results in linear mixture case

<table>
<thead>
<tr>
<th>Evaluation function</th>
<th>average</th>
<th>min</th>
<th>$mse$</th>
<th>average</th>
<th>min</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original source signals $s_0(t)$</td>
<td>0.000</td>
<td>-</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mixed signals $x_0(t)$</td>
<td>0.089</td>
<td>0.129</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Standard PSO</td>
<td>0.000</td>
<td>0.000</td>
<td>0.004</td>
<td>0.000</td>
<td></td>
</tr>
<tr>
<td>Ring-PSO</td>
<td>0.000</td>
<td>0.000</td>
<td>0.011</td>
<td>0.000</td>
<td></td>
</tr>
<tr>
<td>Steepest descent method</td>
<td>20.36</td>
<td>0.000</td>
<td>0.163</td>
<td>0.000</td>
<td></td>
</tr>
</tbody>
</table>
Figure 8(a) illustrates the original source signals \( s_1(t) \), and Fig. 8(b) illustrates the mixed signals \( x_1(t) \).

In order to compare the performances, we carry out numerical simulations by the standard PSO method, the Ring-PSO method, and the steepest descent method. These methods use the same initial values. The simulation conditions are shown in Table 1. The number of particles is 40. \( \eta \) for the steepest descent method is 0.0005. The maximum number of iterations for the steepest descent method is 500.

Figures 8(c), 8(d), and 8(e) respectively show the signals separated by the standard PSO, the Ring-PSO, and the steepest descent method. The simulation results are shown in Table 3. The table shows each evaluation value and its mean square error. In Table 3, the average and minimum values are represented. These results indicate that the evaluation value and mean square error of the PSO method are small compared with those of the steepest descent method.

6.3 The case of a nonlinear mixture using a sigmoid function

Next, we consider the following two-channel nonlinear mixture system with a sigmoid system.

\[
x_2(t) = B_2 \tanh(B_1 s_2(t))
\]

\[
\tanh(x) = (\tanh(x_1), \cdots, \tanh(x_n))^T
\]

where \( \tanh(\cdot) \) is a sigmoid function that is applied to each element of the input vector. \( \tanh(\cdot) \) is the vector function that maps each element of the vector using \( \tanh(\cdot) \). \( B_1 \) and \( B_2 \) represent the following mixture matrices.

\[
B_1 = \begin{bmatrix} 0.8 & 2.2 \\ 0.4 & 2.6 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0.1 & -0.9 \\ 0.7 & 0.3 \end{bmatrix}
\]

The original source signals \( s_2(t) \) consist of a sinusoidal signal and a rectangular waveform signal.

\[
s_2(t) = \begin{bmatrix} \sin(2\pi 90t) \\ \text{sgn}(\sin(2\pi 155t)) \end{bmatrix}
\]

The mixed signals \( x(t) \) are generated using such a nonlinear mixture system. Figure 9(a) illustrates the original source signals \( s_2(t) \), and Fig. 9(b) illustrates the mixed signals \( x_2(t) \).

In order to compare the performances, we carry out numerical simulations by the standard PSO method,
the Ring-PSO method, and the steepest descent method. These methods use the same initial values. The simulation conditions are shown in Table 1. Moreover, the number of particles is changed from 10 to 80 in increments of 10 units. \( \eta \) for the steepest descent method is 0.0005. The maximum number of iterations for the steepest descent method is 500.

Figure 10 shows the relationship between the evaluation function value and the number of particles. The simulation results indicate that the search performance is improved when the number of particles is increased. In addition, Fig. 10 indicates that Ring-PSO yields a small evaluation value. Figures 9(c), 9(d), and 9(e) respectively show the signals separated by the standard PSO, the Ring-PSO, and the steepest descent method. These are the results for the case of 40 particles. The simulation results are shown in Table 4. The average and minimum values are presented. These results indicate that the evaluation value obtained by the Ring-PSO method is small compared with that of the steepest descent method. Therefore, the Ring-PSO method exhibits high performance.

6.4 The case of a nonlinear mixed using a sigmoid function without preprocessing

We investigate the effect of the whitening operation in preprocessing. In order to confirm the effect of preprocessing, we carry out a numerical simulation without preprocessing under the same conditions as in the case of the nonlinear mixture using a sigmoid function. Also, the number of particles is 40.

Table 4  Simulation results for the case where the two signals are mixed nonlinearly by a sigmoid function

<table>
<thead>
<tr>
<th></th>
<th>Evaluation function</th>
<th>( \text{mse} )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>average</td>
<td>min</td>
</tr>
<tr>
<td>Original source ( s_3(t) )</td>
<td>0.000</td>
<td>-</td>
</tr>
<tr>
<td>Mixed signals ( x_3(t) )</td>
<td>1.542</td>
<td>0.250</td>
</tr>
<tr>
<td>Standard PSO</td>
<td>0.096</td>
<td>0.000</td>
</tr>
<tr>
<td>Ring-PSO</td>
<td>0.050</td>
<td>0.000</td>
</tr>
<tr>
<td>Steepest descent method</td>
<td>18.551</td>
<td>0.000</td>
</tr>
</tbody>
</table>

Figures 11(a), 11(b), and 11(c) show the signals separated by the standard PSO, the Ring-PSO, and the steepest descent method, respectively, without preprocessing. The evaluation value and mean square error of each simulation result are shown in Table 5. Figure 11 and Table 5 indicate that the performance is inferior to that indicated by Fig. 9 and Table 4. These results indicate that preprocessing is effective for signal separation.

6.5 The case where three signals are mixed nonlinearly

Next, we consider the following three-channel nonlinear mixture system.

\[
x_3(t) = B_4 \tanh (B_3 s_3(t))
\]  (30)
Fig. 9 The case where two signals are mixed nonlinearly by a sigmoid function

Table 5 Simulation results for the case where the two signals are mixed nonlinearly without preprocessing

<table>
<thead>
<tr>
<th>method</th>
<th>Evaluation function average</th>
<th>min</th>
<th>mse average</th>
<th>min</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard PSO</td>
<td>0.020 0.000</td>
<td>0.251 0.042</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ring-PSO</td>
<td>0.000 0.000</td>
<td>0.250 0.036</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Steepest descent method</td>
<td>7.496 0.009</td>
<td>0.151 0.020</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The original source signals $s_3(t)$ consist of a sinusoidal signal, a rectangular waveform signal, and a triangular waveform signal.

$$s_3(t) = \begin{bmatrix} \sin(2\pi 90t) \\ \text{sgn}(\sin(2\pi 155t)) \\ \text{tri}(t, 0.002) \end{bmatrix}$$

Figure 12(a) illustrates the original source signals $s_3(t)$, and Fig. 12(b) illustrates the mixed signals $x_3(t)$. 

$B_3$ and $B_4$ represent the following mixture matrices.

$$B_3 = \begin{bmatrix} 0.4 & 1.2 & 0.4 \\ 0.8 & 1.0 & 0.2 \\ 0.2 & 1.6 & 0.2 \end{bmatrix}, \quad B_4 = \begin{bmatrix} 0.1 & 0.2 & 0.7 \\ 0.8 & 0.1 & 0.1 \\ 0.3 & 0.2 & 0.5 \end{bmatrix} \quad (31)$$

The original source signals $s_2(t)$ consist of a sinusoidal signal, a rectangular waveform signal, and a triangular waveform signal.

$$s_2(t) = \begin{bmatrix} \sin(2\pi 90t) \\ 0 \\ 0 \end{bmatrix}$$

(b) Mixed signals $x_2(t)$
Table 6  Simulation parameters for the case where the three signals are mixed nonlinearly

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of trials</td>
<td>20</td>
</tr>
<tr>
<td>Number of hidden neurons</td>
<td>9</td>
</tr>
<tr>
<td>Maximum number of iterations for PSO</td>
<td>4000</td>
</tr>
<tr>
<td>Inertia weight coefficient $w$</td>
<td>$w = 0.750$</td>
</tr>
<tr>
<td>Acceleration coefficient $c_1$, $c_2$</td>
<td>$c_1 = c_2 = 1.49$</td>
</tr>
</tbody>
</table>

Table 7  Simulation results for the case where the three signals are mixed nonlinearly

<table>
<thead>
<tr>
<th>Evaluation function</th>
<th>$\mu$, $\sigma$</th>
<th>$\text{mse}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average, min</td>
<td>$E$, $\sigma$</td>
<td>$\mu$, $\sigma$</td>
</tr>
<tr>
<td>Standard PSO</td>
<td>$0.000$</td>
<td>$-1.100$</td>
</tr>
<tr>
<td>Ring-PSO</td>
<td>$3.278$</td>
<td>$0.280$</td>
</tr>
<tr>
<td>Steepest descent</td>
<td>$296.0$</td>
<td>$0.000$</td>
</tr>
</tbody>
</table>

To compare the performances, we carry out numerical simulations by the standard PSO method, the Ring-PSO method, and the steepest descent method. These methods use the same initial values. The simulation conditions are shown in Table 6. For the steepest descent method, the maximum number of iterations is 0.0005. The maximum number of iterations for the steepest descent method is 500.

Figures 12(c), 12(d), and 12(e) respectively show the signals separated by the standard PSO, the Ring-PSO and the steepest descent method. The evaluation value and mean square error of the signals $s_1(t)$, and $x_2(t)$ are shown in Table 7. In Table 7, the average and minimum values are presented. Even when the three signals are mixed, the system is able to separate them, but the performance is inferior to that of the case where two signals are mixed. However, the PSO method exhibits better performance than the steepest descent method.

7. Conclusions

In this paper, we proposed a nonlinear BSS algorithm consisting of an RBF network. The parameters of the RBF network are adjusted by the PSO algorithm. Compared with the conventional gradient method, we confirmed that the proposed algorithm exhibits higher performance when identical initial values are used. However, the system frequently falls into local solutions in the numerical simulations. One of the reasons why the system is caught in such local solutions is the characteristic of the evaluation function. Therefore, improving the evaluation function is one of our future tasks.

References

Fig. 12 The case where three signals are mixed nonlinearly


Takuya Kurihara received his B. Eng. and M. Eng. degrees from Nippon Institute of Technology, Saitama, Japan, in 2011 and 2013, respectively. He is presently working toward his Dr. Eng. degree at Nippon Institute of Technology. His research interests are in blind source separation.

Kenya Jin’no received his B. E., M. E. and Ph. D. degrees in electrical engineering from Hosei University, Tokyo, Japan, in 1991, 1993 and 1996, respectively. From 1994 to 1996, he was a Research Fellow of the Japan Society for the Promotion of Science. From 1996 to 1998, he was an Assistant Professor in the Department of Electrical and Electronics Engineering at Sophia University, Tokyo, Japan. From 1998 to 2004, he was an Associate Professor in the Department of Electrical and Electronics Engineering, Nippon Institute of Technology, Saitama, Japan. He has received research encouragement awards from IEICE in 1996. From 2004 to 2008, he was an Associate Professor in the Department of Networks and Multi-Media Engineering, Kanto-Gakuin University, Yokohama, Japan. From 2008 to 2009, he was a Research Fellow of the JST ERATO Aihara Complexity Modeling Project, Tokyo, Japan. In 2009, he joined the Department of Electrical and Electronics Engineering, Nippon Institute of Technology, Saitama, Japan where he is currently a Professor. His research interests include artificial neural networks, soft computing, high-dimensional nonlinear systems, image processing, complex systems, and evolutionary computation. Dr. Jin’no is a member of RISP, IEEE, INNS, IEICE, and IPSJ.

(Received May 15, 2013; revised September 6, 2013)