MATEC Web of Conferences **61**, 01014 (2016) DOI: 10.1051/matecconf/20166101014
APOP2016

A Nonlinear Blind Source Separation Method Based On Radial Basis Function and Quantum Genetic Algorithm

Pidong Wang[1,2], Ruoyun Tang[1,3], Di Zhao[1] , Hongyi Li[1,3,*], and Jiaxin Chen [4]

[1]*LMIB, School of Mathematics and Systems Science, Beihang University, Beijing 100191, China*
[2]*School of Reliability and Systems Engineering, Beihang University, Beijing 100191, China*
[3]*School of Software Engineering, Beihang University, Beijing 100191, China*
[4]*School of Computer Science and Engineering, Beihang University, Beijing 100191, China*
**Corresponding author (E-mail:Hongyili_buaa@163.com)*

Abstract. Blind source separation is a hot topic in signal processing. Most existing works focus on dealing with linear combined signals, while in practice we always encounter with nonlinear mixed signals. To address the problem of nonlinear source separation, in this paper we propose a novel algorithm using radial basis function neutral network, optimized by multi-universe parallel quantum genetic algorithm. Experiments show the efficiency of the proposed method.

Keywords. EMI signal, Non-linear blind source separation, Radial basis function, Quantum genetic algorithm

1 Introduction

In practice, the nonlinear mixed signals are widespread in blind source separation(BSS), which traditional methods based on linear mixture assumption like ICA are unable to deal with. According to [1,10-12], supposing that x, y are two independent random variables, and f, g are two nonlinear functions, then f(x) and g(y) are independent, too. This fact means that the nonlinear function of source signals could possibly be recovered with the independence assumption of source signals. [2] discusses the existence and uniqueness of solutions to nonlinear ICA and concludes that the solution exists but more assumptions should be added to confirm the uniqueness of solution.

Recently, a method utilizing the radial basis function neutral network (RBFNN) and chaotic self-adaptive particle swarm optimization was introduced in [3]. RBFNN does not deed to determine the mixture model. Theoretically, it is unsupervised and can approximate an arbitrary function [4- 5]. However, RBFNN always converges to a local minimum by adopting general optimization method, such as the gradient descent algorithm. Optimization algorithm searching for global minimum such as Genetic Algorithm [9-12] also doesn't perform well due to large amounts of control parameters, complex calculation and being sensitive to initial values. Quantum genetic algorithm (QGA) combine the features of quantum probability amplitude, superposition, polymorphism and group optimization, which could effectively improve the efficiency. Moreover, the multi-universe parallel quantum genetic algorithm (MPQGA) [6] integrates the concept of subgroup with QGA. We utilize the RBFNN to process signals, and employ fourth-order joint cumulants as independent criterion for separation to optimize parameters.

The rest of this article is as follows: in the second section presents the preliminaries. In section 3, we describe our proposed method in great detail. Then, in section 4, experiments results on synthetic data are provided to demonstrate the performance of the proposed method. Finally, we draw conclusions in the last section.

2 Preliminaries

In this section, we will briefly introduce the nonlinear mixed model, radial basis function neural network, and the multi-universe parallel quantum genetic algorithm.

2.1 Nonlinear mixed model

Nonlinear mixing model for nonlinear BSS could be expressed as x(t)=f(s(t)), where s(t) means the source signal, x(t) is the mixed signal, and $f(\cdot)$ is a unknown nonlinear function.

Supposing $f(\cdot)$ is reversible, and its inverse function exists, denoted by $g(\cdot,\theta)$, where θ is the undetermined parameter. Thus, we obtain the following

$$y(t) = g(x(t),\theta) = g(f(s(t)),\theta) = s(t) \qquad (1)$$

[a] Corresponding author: author@e-mail.org

where $y(t)$ could regarded as an estimation for the source signal $s(t)$.

It could be inferred that if $g(\cdot,\theta)$ could be approximated, $y(t)$ could be obtained from the observed signal. And RBFNN could approximate an unknown function, making it suitable for nonlinear BSS.

2.2 Radial basis function neural net

Broomhead put forward Radial basis Function Neural Network(RBFNN) in 1988 [7]. RBFNN is a feed-forward neural networks with ability of local approximation.

Fig. 1 shows the structure of an N-dimensional input, N-dimensional output network model. It consists of 3 layers; an input layer with N nodes, a hidden layer with one M nodes, and an output layer with N nodes. Nodes in hidden layer are called radial basis neurons and have local responses to the input layer. Nodes in output layer are called linear neurons and conduct linear accumulation on the output from hidden layer.

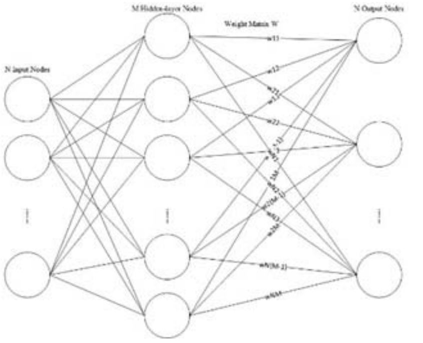

Figure. 1. The topology structure of a typical RBFN

RBFNN is usually utilized to approximate an unknown function $Q: R^n \to R^n$, which could be expressed as $U(X) = WK(X,P)$, where $W = [w_{ij}]$ is an $N \times N$ weight matrix of the output layer, $K(X,P)$ is the core function vector of this network composed of partly corresponding functions, and $X = [x_1, x_2, \Lambda\ , x_N]^{\mathrm{T}}$ is the input vector.

In this paper, we choose the Gaussian kernel as

$$K(X,P)=[1,\exp(-(X-\mu_1)^T(X-\mu_1)/\sigma_1^2),\exp(-(X-\mu_M)^T(X-\mu_M)/\sigma_M^2)]^{\mathrm{T}}$$

, where $X = [x_1, x_2, \Lambda\ , x_N]^{\mathrm{T}}$ is the input vector, and , μ_i, σ_i are the center and width parameter of the i-th neuron. $P = (\mu_1, \Lambda\ , \mu_M, \sigma_1, \Lambda\ , \sigma_M)$ is the vector parameter for the kernel. And the first component of $K(X,P)$ is set to 1.

2.3 Fourth-order joint cumulant

For a zero-mean random variable, its fourth-order joint cumulant is

$$cum(y_1^2, y_2^2) = E\{y_1^2 y_2^2\} - E\{y_1^2\}E\{y_2^2\} - 2(E\{y_1 y_2\})^2 \quad (2)$$.

According to [8], kurtosis is not appropriate for nonlinear problems, then the fourth-order cumulant is proposed as a criterion of independence. It tends to be 0, when the corresponding variables tend to be separated. Therefore it could be utilized for objective function of RBFNN.

2.4 Multi-universe parallel quantum genetic algorithm

It is difficult to obtain global optimal solution in the field of NP-hard problem like nonlinear BSS by using the gradient descent method. Multi-universe parallel quantum genetic algorithm (MPQGA) [6] is a kind of improvement for quantum genetic algorithm. This algorithm groups the individuals into several independent sub-groups, which are called as universes, according to their topology structure. Multi-state gene quantum coding is used to express individuals, which is called qubit coding. Quantum revolving door and quantum NOT gate are used to perform evolution and mutation operation on individuals in the same universe. Best immigration and quantum cross operation are performed between universes. More concrete content about MPQGA such as multiple universe parallel topology, Qubit coding of Chromosome and Information Exchanging Strategy are referred in [18].

3 Preliminaries A nonlinear BSS method based on radial basis function and quantum generic algorithm

In this paper, we use the topology structure as shown in Fig. 2.

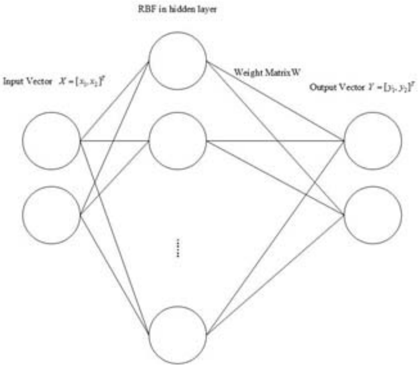

Figure. 2 Selected topology structure

The overall process of our proposed nonlinear BSS model based on RF and MPQGA is shown in Fig. 3. Assuming that nonlinear BSS problem is positive definite, input vector is consist of two mixed signals. After the preprocessing of centralization and whitening, mixed signal vector becomes $X = [x_1, x_2, \Lambda\ , x_N]^{\mathrm{T}}$, where $x_i = [x_{i,1}, x_{i,2}, \Lambda\ , x_{i,t}]^{\mathrm{T}}$ means a mixed signal with temporal structure, and $x_{i,t}$ means the value of signal x_i at time *t*.

The cluster step utilizes mixed signal at time *t*, i.e. $x_{:,t}=[x_{1,t},x_{2,t},\Lambda,x_{N,t}]^{\mathrm{T}}$, to get an *M*-dimensional clustering center μ, where the width parameter corresponding to the *i*-th center μ_i is set to mean value of the distances from the three nearest cluster centers. Then the hidden layer nodes of RBFNN, the nuclear function vector could be computed as follows:

$$K(X,P)=[1,\exp(-(X-\mu_1)^T(X-\mu_1)/\sigma_1^2),\exp(-(X-\mu_M)^T(X-\mu_M)/\sigma_M^2)]^{\mathrm{T}} \quad (3)$$

where *X* vector variable will accept the input vector of RBFNN as value.

After this step, we start to optimize RBFNN optimized by MPQGA iteratively. The input vector is the mixed signal $X=[x_1,x_2,\Lambda,x_N]^{\mathrm{T}}$. Then output vector of hidden layer nodes is:

$$[1,\exp(-(x-\mu_1)^T(x-\mu_1)/\sigma_1^2),\exp(-(x-\mu_M)^T(x-\mu_M)/\sigma_M^2)]^{\mathrm{T}} \quad (4)$$

And the output vector $\mathrm{y}=[y_1,y_2,\Lambda,y_N]^{\mathrm{T}}$ could be computed through $N\times(M+1)$ dimension weight matrix $W=[w_{ij}]$ from hidden layer to output layer. After that, we calculate the objective function $fitness(y_1,y_2)=1/|\,cum\{y_1^2,y_2^2\}\,||$ to estimate independence of signal separation, which is the feedback for RBF neural network to iterate again. After reaching the iterative target, $\mathrm{y}=[y_1,y_2,\Lambda,y_N]^{\mathrm{T}}$ will be regarded as the output vector of the RBF neural network and the estimation of source signal.

More details about Multi-universe parallel quantum genetic algorithm (MPQGA) to optimize the weight matrix $W=[w_{ij}]$ could refer to the [18].

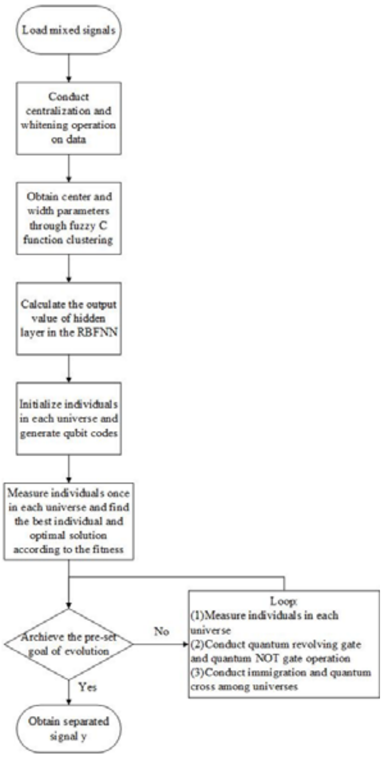
Load mixed signals
Conduct centralization and whitening operation on data
Obtain center and width parameters through fuzzy C function clustering
Calculate the output value of hidden layer in the RBFNN
Initialize individuals in each universe and generate qubit codes
Measure individuals once in each universe and find the best individual and optimal solution according to the fitness
Archieve the pre-set goal of evolution
No
Loop:
(1)Measure individuals in each universe
(2)Conduct quantum revolving gate and quantum NOT gate operation
(3)Conduct immigration and quantum cross among universes
Yes
Obtain separated signal y

Figure. 3 Flow chart of the proposed method

4 Experimental results and analysis

Simulation experiment is done using Matlab2013a. The source signals are $s_1(t)=0.5(1+\sin(7\pi t))\cos t$ and $s_2(t)=\sin(22\pi t)$. The mixing function is $f(t)=s^3$ and the mixed signal vector is $x(t)=A_2 f(A_1 s(t))$, where A_1 and A_2 are random mixing matrix and $s(t)=[s_1(t),s_2(t)]^{\mathrm{T}}$ is source signal vector.

4.1 Fourth-order joint cumulant

For a zero-mean random variable, its fourth-order joint cumulant is

$$cum(y_1^2,y_2^2)=E\{y_1^2y_2^2\}-E\{y_1^2\}E\{y_2^2\}-2(E\{y_1y_2\})^2 \quad .(5)$$

According to [8], kurtosis is not appropriate for nonlinear problems, then the fourth-order cumulant is proposed as a criterion of independence. It tends to be 0, when the corresponding variables tend to be separated. Therefore it could be utilized for objective function of RBFNN.

4.2 Source signals and mixed signals

The figure of source signals and mixed signals is as follows.

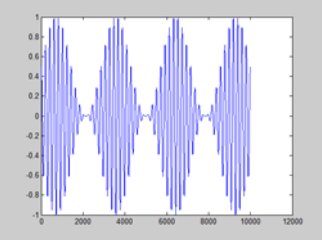

Figure. 4 Source signal $s_1(t)$

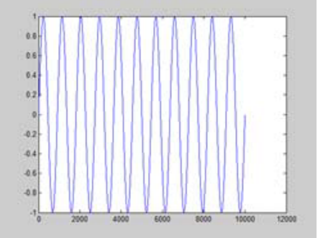

Figure. 5 Source signal $s_2(t)$

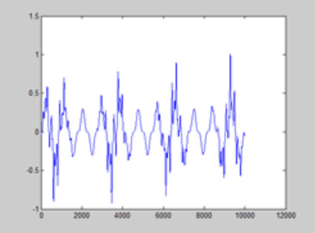

Figure. 6 Mixed signal $x_1(t)$

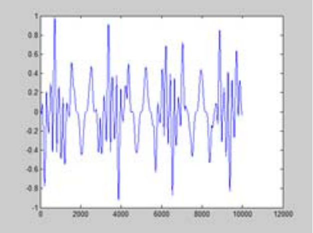

Figure. 7 Mixed signal $x_2(t)$

4.3 Experiment Results

4.3.1 Evaluate function

Similar coefficient is used to measure similarity between source signals and estimated signals for solving sequential uncertainty of output vector. The similarity coefficient between the source signal s_i and estimated signal y_j is defined as $\rho_{ij}=|\,s_i\cdot y_j\,|/\sqrt{s_i^2\cdot y_j^2}$.

The value tends to 1 when source signals and estimated signals tends to similar. The uncertainty could be eliminated by choosing the corresponding source signal and estimated signal with high similarity coefficient.

4.3.2 Experimental result

After 20000 generation, the result is like the following.

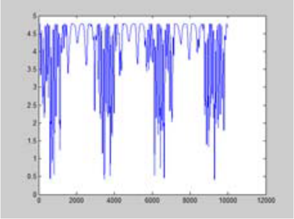

Figure. 8 Estimated signal y_1

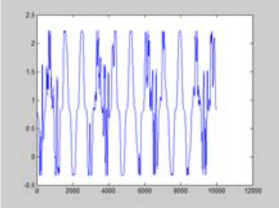

Figure. 9 Estimated signal y_2

Table 1 shows the similarity coefficients between each source signal and estimated signal is called original table of similarity coefficients.

The first row of Table 1 presents two estimated signals output by RBFNN optimized by MPQGA. And the first column presents two source signals.

Table 1. The similarity coefficients between each source signal and estimated signal

ρ_{ij}	y_1	y_2
s_1	0.0362	0.2270
s_2	0.0089	0.8898

Table 2. The similarity coefficients between the source signal and its corresponding estimated signal

$s_i\_y_j$	$s_1\_y_1$	$s_2\_y_2$
ρ_{ij}	0.0362	0.8898

The first row of Table 2 presents the chosen source signal and its corresponding estimated signal, e.g. $s_1\_y_1$ means the estimated signal, y_1 is the estimated signal of source signal s_1. The similarity between them is 0.0362. As the two tables shows, our method performs well on source signal s_2 while bad on s_1, due to the disadvantages of GA and search space fixed to [-1,2].

4.3.3 Comparison experiment to RBFNN optimized by natural gradient algorithm

For comparison, RBFNN optimized by natural gradient algorithm (NGA) is experimented for same parameters. Results are as the following.

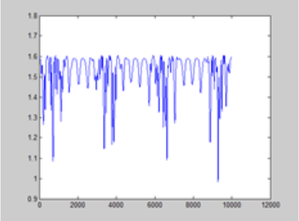

Figure.10 Estimated signal y_1

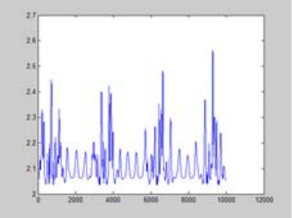

Figure. 11 Estimated signal y_2

Table 3 Original table

ρ_{ij}	y_1	y_2
s_1	0.5360	0.0226
s_2	0.6304	0.3572

Table 4 Corresponding table

$s_i\_y_j$	$s_1\_y_1$	$s_2\_y_2$
ρ_{ij}	0.5360	0.3572

The comparison table of the two methods is as the following.

Table 5 Comparisons between MPQGA and NGA

$s_i\_y_j$	$s_1\_y_1$	$s_2\_y_2$	$s_1\_y_1$	$s_2\_y_2$
Method	MPQGA	MPQGA	NGA	NGA
ρ_{ij}	0.0362	0.8898	0.5360	0.3572

The first row of figure indicates estimated signals computed from methods the second row indicates, such as $s_1\_y_1$ and MPQGA in type means the corresponding result s_1 and y_1 from the method optimized by MPQGA.

It could be seen that effect of our method is much better than the natural gradient one on s_2, though both low on s_1.

4.3.4 Comparison experiment to RBFNN optimized by MPQGA emphasizing search space

We found a drawback in experiment above about the same size of search space for each gene, caused by fixed explanation to codes of MPQGA, which leads to unbalance between size of search space and precision of search yielding local optimum. Experiment in 4.2.2 set the search space [-1,2] and precision 0.0001 to emphasize precision. A comparison experiment is done to show the difference between emphasizing precision and emphasizing search space, based on same parameters except for the explanation to codes. Results are as the following.

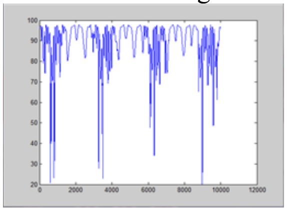

Figure.12 Estimated signal y_1

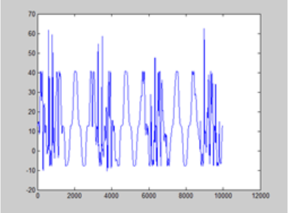

Figure.13 Estimated signal y_2

Table 6 Original table

ρ_{ij}	y_1	y_2
s_1	0.335	0.107
s_2	0.781	0.212

Table 7 Corresponding table

$s_i\_y_j$	$s_1\_y_1$	$s_2\_y_2$
ρ_{ij}	0.1078	0.7818

Compare to the experiment emphasizing search precision before,It could be seen that the MPQGA emphasizing search space is more powerful on the estimation of 1 s , but the similarity coefficient is still low because GA cannot balance search space and search precision adaptively.

Table 8 Comparisons between methods emphasizing precision and space

$s_i\_y_j$	$s_1\_y_1$	$s_2\_y_2$	$s_1\_y_1$	$s_2\_y_2$
Method	Precision	Precision	Space	Space
ρ_{ij}	0.0362	0.8898	0.1078	0.7818

5 Conclusion

Appropriate result is obtained in our experiment and shows our method's effectiveness. However, our method has high time complexity, and requires a large amount of sample data, which would be considered in future.

Acknowledgement

This work was supported by the National Natural Science Foundation of China (No.61379001) and the Key Project of Beihang University (No. 201412) ("Exploration and Practice of Innovation Ability Cultivation in Mathematical Education").

References

1. A. Papoulis, S. U. Pillai. Probability, random variables, and stochastic processes. Tata McGraw-Hill Education, (2002)
2. A. Hyvärinen A, P. Pajunen. Nonlinear independent component analysis: Existence and uniqueness results. Neural Networks, **12(3)**, 429-439 (1999)
3. M. Li, X. Huang, H. Liu, B. Liu, Y. Wu. Prediction of the gas solubility in polymers by a radial basis function neural network based on chaotic self-adaptive particle swarm optimization and a clustering method. Journal of Applied Polymer Science,**130(5)**, 3825-3832 (2013)
4. X. M. Wang. Nonlinear Blind Source Separation using GA optimized RBF-ICA and its Application to the Image Noise Removal. International Conference on Biotechnology, Chemical and Materials Engineering, (2011)
5. T. Kurihara, K. Jinno. Analysis of convergence property of PSO and its application to nonlinear blind source separation. IEEE Congress on Evolutionary, (2013)
6. J. Yang, Z. Zhuang, L. Shi. Multi-universe parallel quantum genetic algorithm. ACTA ELECTRONICA SINICA., **32(6)**, 923-928 (2004)
7. D. S. Broomhead, D.Lowe. Multivariate functional interpolation and adaptive networks. Complex System, **2**, 321-355 (1998)
8. J. A. Yang, Y. Zou, Z. Q. Zhuang. Nonlinear blind source separation algorithm using multi-universe parallel quantum genetic algorithm. Journal of Electronics and Information Technology, 2004, 26(8): 1210-1217.
9. G. Li, K. H. Lee, K. S. Leung, Genetic algorithm based on independent component analysis for global optimization. Parallel Problem Solving from Nature-PPSN IX, Springer Berlin Heidelberg, Iceland, 172-181 (2006)
10. H. Li, C. Wang, D. Zhao. An Improved EMD and Its Applications to Find the Basis Functions of EMI Signals. Mathematical Problems in Engineering, (2015)
11. H. Li, Z. Song, D. Zhao, P. Wang, J. Chen. A Single Channel EMI Signal Separation Method Based on Directly-mean Empirical Mode Decomposition. Journal of Infromation & Computation Science, **12(17)**, 6333-6340 (2015)
12. D. Zhao, H. Li, On the computation of inverses and determinants of a kind of special matrices. Applied Mathematics and Computation, **250**, 721-726 (2015)