Available online at www.sciencedirect.com
SCIENCE DIRECT®
ELSEVIER
Neurocomputing 69 (2006) 884–887
NEUROCOMPUTING
www.elsevier.com/locate/neucom

Letters

Optimal selection of time lags for TDSEP based on genetic algorithm

Zhan-Li Sun[a,b], De-Shuang Huang[a,*,1], Chun-Hou Zheng[a,b], Li Shang[a,b]

[a]*Intelligent Computing Lab, Hefei Institute of Intelligent Machines, Chinese Academy of Sciences, P.O. Box 1130, Hefei, Anhui 230031, China*
[b]*Department of Automation, University of Science and Technology of China, Hefei, Anhui 230026, China*

Received 26 May 2005; received in revised form 8 June 2005; accepted 8 June 2005
Available online 10 October 2005
Communicated by R.W. Newcomb

Abstract

In this letter, a two-step learning scheme for the optimal selection of time lags is proposed for a typical temporal blind source separation (TBSS), Temporal Decorrelation source SEParation algorithm (abbreviated as TDSEP). Given the time lags, the time-delayed second-order correlation matrices are first diagonalized simultaneously. Then, a genetic algorithm is used to update the time lags. Finally, experimental results demonstrate that the proposed method can efficiently accomplish the aforementioned task.
© 2005 Elsevier B.V. All rights reserved.

Keywords: Blind source separation; Time lag; Genetic algorithm

1. Introduction

Blind source separation (BSS) has become an important research area in the last few years since it has wide potential applications in signal processing and data analysis [4]. For the linear BSS, the observed signal vector $\mathbf{x}(t) = [x_1(t),\ x_2(t), \cdots, x_n(t)]^{\mathrm{T}}$ is assumed to be a mixture of unknown source signal vector $\mathbf{s}(t) = [s_1(t),\ s_2(t), \cdots s_n(t)]^{\mathrm{T}}$ formed by the following model:

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t), \quad t = 1, \cdots, T. \tag{1}$$

For simplicity, here, the mixing matrix $\mathbf{A}$ is assumed to be an unknown square matrix. The source signals $s_j(t)$, $j = 1, \cdots, n$ are supposed to be mutually statistically independent. The task for the linear BSS is to recover $\mathbf{s}(t)$ from the given mixed signals $\mathbf{x}(t)$. Compared to the methods using higher-order statistics, BSS using signal time structure is more stable and the independent components (ICs) $s_j(t)$, $j = 1, \cdots, n$ need not to be nongaussian. A typical temporal blind source seperation (TBSS) algorithm is the so-called Temporal Decorrelation source SEParation method (abbreviated as TDSEP) [8], which is realized through an approximate simultaneous diagonalization [1] of several time-delayed second-order correlation matrices. However, for most TBSS methods, how to select the optimal time lags is an important problem. To our knowledge, there still has not been a good and efficient method proposed to solve this problem [5,7,8]. Therefore, in this letter, based on the genetic algorithm (GA) [6], we propose a novel unsupervised method to determine the optimal or approximate optimal time lags for the TDSEP algorithm. Moreover, it should be pointed out that the main idea for our proposed method can also be used to improve other TBSS algorithms, such as the extension of the TDSEP algorithm [2].

2. Optimal selection of time lags by the genetic algorithm

The time-lagged covariance matrix of the time series $\mathbf{x}(t)$ can be defined as

$$\mathbf{C}_\tau^{\mathbf{x}} = \mathbf{E}\{\mathbf{x}(t)\mathbf{x}(t+\tau)\}, \tag{2}$$

where $\mathbf{E}$ denotes expectation and τ is a certain time lag, $\tau = 1, 2, 3, \cdots, M$. What we do is to find a matrix $\mathbf{B}$ so that in addition to making the instantaneous covariances of $\mathbf{y(t)} = \mathbf{Bx(t)}$ go to zero, the lagged covariances are made

*Corresponding author.
E-mail addresses: zhlsun2006@yahoo.com.cn (Z.-L. Sun), dshuang@iim.ac.cn (D.-S. Huang).
[1]This work was supported by the National Science Foundation of China (Nos. 60472111 and 60405002).

0925-2312/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.neucom.2005.06.010

zero as well [4]:

$$E\{y_i(t)y_j(t+\tau)\} = 0, \quad i,j = 1,\cdots,n, \quad \tau = 1,\cdots,M. \tag{3}$$

To reduce the computational complexity, the data $\mathbf{x}(t)$ are first whitened by a whitening transformation:

$$\mathbf{Z}(t) = \mathbf{W}\mathbf{x}(t). \tag{4}$$

The whitening matrix $\mathbf{W}$ can be determined by

$$\mathbf{W} = \mathbf{D}^{-1/2}\mathbf{V}^T, \tag{5}$$

where $\mathbf{D} = \mathrm{diag}\,(d_1, d_2, \cdots, d_n)$ is the diagonal matrix of the eigenvalues of $\mathbf{C}_0^{\mathbf{x}} = \mathbf{E}\{\mathbf{x}\mathbf{x}^T\}$ and $\mathbf{V}$ is the matrix whose columns are eigenvectors of $\mathbf{C}_0^{\mathbf{x}}$. Subsequently, we should simultaneously diagonalize all the corresponding lagged covariance matrices:

$$\sum_{\tau(\mathbf{z})} = \sum_{\tau=1}^{M} \langle \mathbf{z}(t)\mathbf{z}(t+\tau)\rangle, \tag{6}$$

where $\langle\cdot\rangle$ denotes a time average. As done in [8], we define the following cost function:

$$\mathbf{y} = \mathbf{Q}\mathbf{z}(t), \tag{7}$$

$$\mathbf{J}(\mathbf{Q},\tau) = \sum_{k=1}^{M}\sum_{i\neq j} \langle y_i(t)y_j(t+\tau_k)\rangle^2. \tag{8}$$

The orthogonal rotation matrix $\mathbf{Q}$ can be obtained by minimizing $\mathbf{J}(\mathbf{Q}, \tau)$ with the approximate simultaneous diagonalization method.

The update of the time lag τ by the GA is given as follows. The key issues are how to encode a solution as a chromosome, how to define a fitness function to evaluate each individual and how to apply genetic operators to the encoded chromosome. As shown in Fig. 1, the first n_d time lags are encoded as a chromosome, where each gene represents a time lag. We make use of binary symbols, 1 or 0, to denote whether the time lag is adopted or not in computing the time-delayed second-order correlation matrices. Our objective is to make the number of time lags selected as few as possible under the given accuracy. To measure the separation performance, a robust approximation of negentropy in [3] is used here.

$$J(\mathbf{y}) = \sum_{i=1}^{n} [E\{G(y_i)\} - E\{G(\upsilon_i)\}]^2, \tag{9}$$

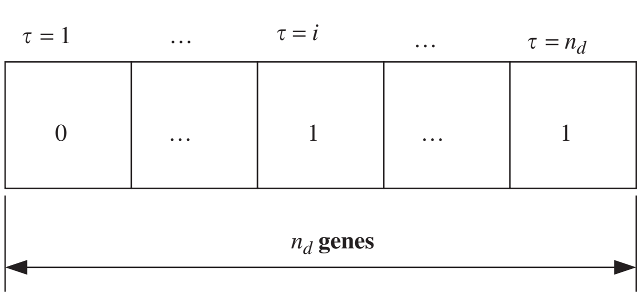

Fig. 1. The encoding scheme of individual for the first n_d time lags used in the TDSEP algorithm.

where υ_i is a Gaussian variable of zero mean and unit variance (i.e., standardized). Generally, a larger negentropy means that the separated signals are more mutually independent. The function $G(\cdot)$ is chosen as

$$G(u) = \frac{1}{a_1}\log\,\cosh(a_1 u), \tag{10}$$

where $1 \leqslant a_1 \leqslant 2$. As for the choice of $G(\cdot)$, please refer to Ref. [3]. Thus, we design a fitness function as follows:

$$f(J, M_i) = \frac{n_d J(\mathbf{y})}{M_i}, \quad i = 1,\cdots,n_p, \tag{11}$$

where M_i is the sum of all gene values in one individual and n_p the number of populations.

In addition, we adopt similar genetic operators as in [6]. For the selection operator, the individual with maximal fitness of the current population is first selected as a component of the next generation without performing crossover and mutation, then the mating parents of next generation are chosen from the remaining individuals. The single-point stochastic crossover is adopted here because of its smaller number of computations and easy realization [6]. The mutation operator is always used to keep the diversity of population, which independently employs the logical NOT operation by the mutation probability p_m.

In the light of the above analyses, the steps of determining the optimal time lags automatically for the TDSEP can be summarized briefly as follows:

Step 1. Set the number of observations T, length of an individual n_d, number of generations n_g, parameter a_1, number of populations n_p, crossover probability p_c and the mutation probability p_m.

Step 2. Whiten the data $\mathbf{x}(t)$, for every individual of one generation, compute the rotation matrix $\mathbf{Q}$ by an approximate simultaneous diagonalization of $M_i(i = 1,\cdots,n_p)$ time-delayed second-order correlation matrices.

Step 3. Update the parameter τ by the genetic operators, and then enter the next generation.

Step 4. Repeat steps 2 and 3 until the last generation is reached.

3. Experimental results

The proposed method was tested on the linear mixtures of three standard sound signal files (handel, chirp and gong) that come with matlab. The mixture matrix (randomly chosen) is given as follows:

$$A = \begin{pmatrix} 2.0211 & 0.2723 & -1.6106 \\ 0.5018 & 0.3368 & -1.0075 \\ -1.9983 & 0.1378 & -0.5144 \end{pmatrix}. \tag{12}$$

The parameters used in experiments are given in Table 1.

To verify the optimal time lags selected by our proposed method, we carried out two sets of experiments. On the one hand, as done in [8], the first k smallest positive integers are chosen as the time lags without optimal selection. To make the experimental result more convincing, we repeated the

Table 1
The parameters used in experiments

n_d	n_g	a_1	T	p_c
100	500	1.5	500	0.85
p_m	n_p			
0.05	100			

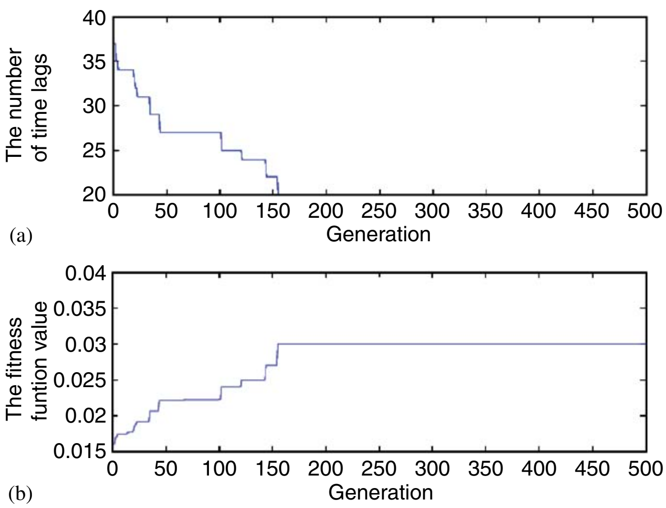

Fig. 2. Curves of the number of time lags (a) and the fitness function (b) versus generations of the GA.

TDSEP algorithm on the mixture when the parameter k varies from 1 to $T-1$. As a result, when the first 40 time lags were used, the negentropy of the separated signals reaches its maximum (0.060). On the other hand, for our proposed method, after 500 iterations, the curves of the number of time lags and the fitness function versus generations of the GA are shown as Fig. 2(a) and (b), respectively. From Fig. 2, we can see that after 155 iterations, the optimal number of time lags and the optimal fitness function start to converge. Finally, the optimized time lag number after convergence is 20 (only 20% of the original time lag number of one individual), which is less than the number directly obtained by the TDSEP. Moreover, the maximum negentropy obtained with these selected time lags is equivalent to that obtained by the TDSEP. So the computational complexity can be reduced when the time lags used in the TDSEP are preselected by our proposed method. This may be very interesting when the number of source signals is large.

In addition, the same conclusions can also be drawn when we change the mixing matrix and data. Due to the limitation of space, the related experimental results are not reported here.

4. Conclusions

Using the genetic algorithm, a novel optimal selection method of time lags for the TDSEP is proposed in this letter. The experimental results demonstrate the validity and feasibility of our proposed method.

Acknowledgments

The authors would like to thank the Editor-in-Chief and two anonymous reviewers who have given us many helpful comments and suggestions.

References

[1] J.-F. Cardoso, A. Souloumiac, Jacobi angles for simultaneous diagonalization, SIAM J. Mater Anal. Appl. 17 (1) (1996) 161–164.

[2] S. Harmeling, A. Ziehe, M. Kawanabe, K.-R. Müller, Kernel-based nonlinear blind source separation, Neural Comput 15 (5) (2003) 1089–1124.

[3] A. Hyvärinen, Fast and robust fixed-point algorithms for independent component analysis, IEEE Trans. Neural Networks 10 (3) (1999) 626–634.

[4] A. Hyvärinen, E. Oja, Independent component analysis: algorithms and applications, Neural Networks 13 (4–5) (2000) 411–430.

[5] K.-R. Müller, R. Vigario, F. Meinecke, A. Ziehe, Blind source separation techniques for decomposing event-related brain signals, Int. J. Bifurcat. Chaos 14 (2) (2004) 773–791.

[6] W.B. Zhao, D.S. Huang, The structure optimization of radial basis probabilistic neural networks based on genetic algorithms, in: Proceedings of the IJCNN'02, Honolulu, HI, USA, May 2002, pp. 1086–1091.

[7] A. Ziehe, M. Kawanabe, S. Harmeling, K.-R. Müller, A fast algorithm for joint diagonalization with non-orthogonal transformations and its application to blind source separation, J. Mach. Learn. Res. 5 (2004) 801–818.

[8] A. Ziehe, K.-R. Müller, TDSEP — an efficient algorithm for blind separation using time structure, in: Proceedings of the International Conference on ICANN, Perspectives in Neural Computing, Berlin, 1998, Springer, Berlin, pp. 675–680.

Zhan-Li Sun obtained his Bachelor's degree in Machinery and Electronic Engineer of the Huainan Industrial University in 1997, obtained his M.Sc. degree in Machinery and Electronic Engineer of the Hefei University of Technology in 2003, From March 2003 on, he is in pursuit for his Doctor's degree in Pattern Recognition & Intelligent System at the University of Science & Technology of China. His research interests include artificial neural networks, blind source separation, signal and image processing.

De-Shuang Huang (SM'98), a Professor and Ph.D. Advisor at the University of Science and Technology of China (USTC), Hefei, China and a Professor of the Graduate School of the Institute of Intelligent Machines, Chinese Academy of Sciences (CAS). From September 2000 to March 2001, he worked as a Research Associate at the Hong Kong Polytechnic University. From April 2002 to June 2003, he worked as a Research Fellow at the City University of Hong Kong. From October to December 2003, he worked as a Research Fellow at the Hong Kong Polytechnic University. From July to December 2004, he worked as the University Fellow at the Hong Kong Baptist University. Dr. Huang is currently a Senior Member of the IEEE. He has published over 190 papers.

Chun-Hou Zheng was born in Shandong province, China, in 1973. He received his M.Sc. degree in Control Theory & Control Engineering in 2001 from the Qu Fu Normal University, China. He is now in pursuit for his Ph.D degree in Pattern Recognition & Intelligent System at the University of Science and Technology of China. His research interests include artificial neural networks, intelligent computing, and intelligent information processing.

Li Shang received her B.Sc. and M.Sc. degrees at the Xi'an Mine University in June 1996 and June 1999, respectively. From September 2003, she is in pursuit for her Doctor's degree in Pattern Recognition & Intelligent System at the University of Science & Technology of China (USTC), Hefei, China. Now, she works at the Department of Automation of USTC. Her research interests include image processing, artificial neural networks and intelligent computing.