Bacteria Foraging Based Independent Component Analysis

D.P.Acharya, G.Panda, SMIEEE,
Department of ECE, NIT Rourkela,
d_p_acharya@rediffmail.com
ganapati.panda@gmail.com

S.Mishra, SMIEEE,
Department of EE, IIT Delhi,
sukumar@ee.iitd.ac.in

Y.V.S.Lakshmi
Center for Development of Telematics, Bangalore
lakshmi@cdotb.ernet.in

Abstract

The present paper proposes a bacteria foraging optimization based independent component analysis (BFOICA) algorithm assuming a linear noise free model. It is observed that the proposed BFOICA algorithm overcomes the long standing permutation ambiguity and recovers the independent components(IC) in a fixed order which depends on the statistical characteristics of the signals to be estimated. The paper compares the performance of BFOICA algorithm with the constrained genetic algorithm based ICA (CGAICA) and most popular fast ICA algorithm. The proposed algorithm offers comparable or even better performance compared to fast ICA algorithm and faster convergence and better mean square error performance compared to CGAICA.

1 Introduction

Independent Component Analysis (ICA) is a statistical signal processing technique in which the goal is to find a linear representation of nongaussian data so that the components are statistically independent or as independent as possible [1, 11]. All the existing methods for ICA do not find a global optimum but gets trapped in the local minimum. The non linear objective functions for ICA being multimodal, this problem becomes more prominent. Therefore, choice of good initial values is important in initializing these algorithms. Besides this, these algorithms have the ambiguities like scaling and permutation. In [2] attempts have been made to overcome the permutation ambiguity in the frequency domain only but the ambiguity remains still unresolved in the time domain.

GA has been applied to ICA problem in [3, 4, 5, 6] However all the applications of GA to BSS problem still have the permutation indeterminacy. A constrained GA based ICA (CGAICA) has been proposed in [9] which estimates all the ICs and resolves the permutation indeterminacy. The Bacterial Foraging Optimization (BFO) is a recently developed derivative free efficient evolutionary computational optimization technique which reveals the global optimum of the contrast function [7]. In this technique the foraging behavior of the *E.Coli* bacteria present in our intestine has been mimicked. This novel scheme has been successfully used for several applications in [8, 9]. Since BFO has been reported to have better performance than GA and GA has been applied successfully to ICA [10] it is quite interesting to study the application of BFO to ICA problem.

The present paper proposes a novel ICA algorithm based on BFO using the popular deflation approach to estimate all the ICs. Using BFOICA in a constrained manner overcomes the long standing permutation uncertainty and recovers the ICs in a fixed order which is dependent on the statistical characteristics of the signals. The proposed BFOICA algorithm has been shown to offer comparable or sometimes better performance as compared to the most popular fast ICA algorithm. It also converges faster and yields better performance than the CGAICA algorithm.

2 Independent Component Analysis

The basic idea of ICA is to minimize the dependency among the output components or maximize some measure of nongaussianity. The extraction of source is done by assuming the latent variables are non-Gaussian and statistically independent [11]. Suppose a set of observations of random variables is $(x_1(t), x_2(t), ..., x_n(t))$ where t is the time or the sample index and they are generated from a linear

0-7695-3050-8/07 $25.00 © 2007 IEEE
DOI 10.1109/ICCIMA.2007.126

mixture of sources $(s_1(t)\ s_2(t)...s_n(t))$ that are statistically independent. This is expressed in the following form.

$$[x_1(t)\ x_2(t)...x_n(t)]^T = A[s_1(t)\ s_2(t)...s_n(t)]^T \quad (1)$$

where A is some unknown mixing matrix and ^{T} stands for the transpose operator for a matrix.

Independent component analysis estimates both A and $s_i(t)$ when only the observations $x_i(t)$ are at hand. The ICs and the columns of A can only be estimated up to a multiplicative constant and their order of appearance is not known. For simplicity we have assumed a linear and noise free and instantaneous mixing model of ICA here.

The estimation of the data model of ICA is usually performed by formulating an objective (cost) function and then optimizing it. The characteristic of ICA is that maximizing non-Gaussianity of data allows the ICs to be obtained.

3 The Proposed Bacterial Foraging Optimization Based ICA Algorithm

Bacterial foraging is a new evolutionary computational method proposed by Passino [7].In this scheme, the foraging behavior of *E.coli* bacteria present in our intestines is mimicked. They undergo different stages such as chemotaxes, swarming, reproduction and elimination and dispersal. The detailed treatment of this new concept is presented in [7].

With BFO algorithm kurtosis [11] is used as the contrast function to be maximized and hence the requirement of minimization of the nutrient function J, defined as

$$J = \frac{1}{ContrastFunction} \quad (2)$$

Before presenting the observed mixed signal data for optimization the two preprocessing steps centering and whitening as in [11] are performed on it and then the following steps are carried out.

Step-1: Initialization

Number of parameters p to be optimized, number of bacteria S, swimming length N_s after which tumbling of bacteria will be undertaken in a chemotatic loop, number of iterations N_c to be under taken in a chemotatic loop $N_c > N_s$ number of reproduction steps N_{re}, the elimination and dispersal probability P_{ed}, the location of each bacterium $P(1-p,1-S,1)$ and the value of $C(i)$ are initialized for the optimization algorithm.

Step-2: Iterative algorithm for optimization

The section models the bacterial population chemotaxis, reproduction, elimination and dispersal (initially, $j = k = l = 0$).For the algorithm updating θ^i automatically resuits in updating of P

i) Elimination-dispersal loop: $l = l + 1$; ii) Reproduction loop: $k = k + 1$;

iii) Chemotaxis loop: $j = j + 1$

a) For $i = 1,2,.....,S,$ calculate cost function value for each bacterium ***i*** as follows.

*Compute value of cost function $J(i, j, k, l)$.

*let $J_{last} = J(i, j, k, l,)$ to save his value since we may find a better cost via a run. *End of for loop.

b) For $i = 1,2,....S$ take the tumbling /swimming decision

*Tumble: Generate a random vector $\Delta(i) \in \Re^p$ with each element $\Delta_m(i) m = 1,2,....p,$ a random number on [-1,1].

*Move: Let. $$\theta^i(j+1,k,l) = \theta^i(j,k,l) + C(i)\frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}} \quad (3)$$

Fixed step size in the direction of tumble for bacterium *i* is considered.

*compute $J(i, j+1, k, l)$,*Swim:i) Let $m = 0$;(counter for swim length)

ii) while $m < N_s$ (have not climbed down too long)

*Let $m = m + 1$, *If $J(i, j+1, k, l) < J_{last}$ (if doing better),

let $J_{last} = J(i, j+1, k, l)$ and $\theta^i(j+1, k, l) = \theta^i(j, k, l) + C(i)\frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}}$ (4)

and use this $\theta^i(j+1, k, l)$ to compute the new $J(i, j+1, k, l)$.

*Else, let $m = N_s$ .This is the end of the while statement.

c) Go to the next bacterium $(i+1)$ if $i \neq S$ (i.e go to b) to process the next bacterium.

iv) If $j < N_{c,}$ go to (iii).In this case, continue chemotaxis since the life of bacteria is not over.

v) Reproduction: a) For the given **k** and ***l***, and for each $i = 1,2,....S,$ let $J^i_{health} = \min_{j \in \{1...N_s\}} \{j\ (i, j, l)\}$ be the health of the bacterium ***I*** (a measure of how many nutrients it got over its life time and how successful it was at avoiding noxious substance).Sort bacteria in order of ascending cost J_{health} (higher cost means lower health).

b) The $S_r = S/2$ bacteria with highest J_{health} values die and other S_r bacteria with the best value split (and the copies that are made are placed at the same location as their parent)

vi) If $k < N_{re}$ go to (ii), in this case, we have not reached the number of specified reproduction steps, so we start the next generation in the chemotactic loop.

vii) Elimination & dispersal: $i = 1,2,...S$ with probability P_{ed} eliminate and disperse each bacterium to a random location on the optimization domain. The position of the bacteria w_1 at which global minimum value is obtained yields the first independent component.

Step-3: Evaluation of the Other Independent Components

The remaining ICs are evaluated by using the deflation approach as given in [10].

4 The Simulation Experiment

In the experimental studies for the verification of the validity and performance of the proposed BFO based ICA algorithm, programs for separating the signals blindly from their observed mixtures were written. The two signals were mixed by a known matrix A and the mixed signals were the inputs to the BFOICA algorithm for separation. Two different examples were taken to verify the separation capability of the proposed algorithm. Some more practical examples are excluded here due to lack of space. For a particular example, the parameters such as a number of bacteria (S), number of chemotactic steps (N_c), number of elimination and dispersal events (N_{ed}), number of reproduction steps (N_{re}), probability of element and dispersal (P_{ed}) and runlengthunit parameters are tuned, to get the proper separation. In this simulation for BFOICA we have considered the following typical values: $S = 8,\ p = 2, N_c = 8,\ N_s = 7,\ N_{re} = 4,\ N_{ed} = 4, P_{ed} = 0.25.$ The separation performance parameter, the mean square error (MSE) was evaluated. The separation was performed by using contrast function kurtosis. The minimum value of the nutrient function J is plotted against the number of its evaluations.

5 Results & Discussions

Example 1: A random binary wave and a sine wave with 400 samples as shown in fig 1 are mixed by the mixing matrix A and their mixtures are represented in fig 2.

$$A = \begin{pmatrix} 0.9121 & 0.2292 \\ 0.4763 & 0.7348 \end{pmatrix} \quad (5)$$

Using BFOICA the two ICs are recovered in the decreasing order of the value of their contrast function. For the case of kurtosis as contrast function the separated signals are depicted in fig 3.

Example 2: We consider another example with a sine wave and its third harmonic and mix these two by the same mixing matrix A as given in (5). The original signals and their mixtures are shown in fig 4 and fig 5. The separated signals are shown in fig 6.

5.1 Comparison of Different Optimization Schemes

Example 1 was repeated with fast ICA and CGAICA algorithms to have a comparison with BFOICA algorithm. Table 1 summarizes the typical MSE values estimated for the recovered ICs for fast ICA, CGAICA and BFOICA algorithms. Most commonly considered 15bit binary GA is adopted here with 20 chromosomes and crossover and mutation probabilities 0.85 and 0.1.The number of bacteria chosen for BFOICA was 40 with 30 chemotactic steps, 3 reproductions and 4 elimination and dispersal events. So it is clearly observed that BFOICA yields far better performance than CGAICA and comparable or sometimes better performance than the most popular fast ICA algorithm.GA being the most popular evolutionary computation algorithm we carried out a comparative study of the convergence of BFOICA with CGAICA algorithm. To have a common ground of comparison with CGAICA we studied the variation of the best value of nutrient function with the number of times the nutrient function is evaluated. Fig 7 shows the variation of the nutrient function J values with the number of J evaluations for random binary component. This clearly indicates that BFOICA algorithm has much faster convergence as compared to the GAICA algorithm. Needless to say that fast ICA has faster convergence than CGAICA or BFOICA algorithm.

5.2 Permutation Ambiguity

From the above two examples it was observed that using BFOICA (CGAICA also) ICs were recovered always in a fixed order in all runs of the simulation experiments. The IC for which the nutrient function has a global minimum value, appeared first and then appeared the IC with subsequent minimum value of the nutrient function J. So we can predict the order of the ICs if we relatively know about the value of their statistical property like kurtosis. However this is not the case with fast ICA or any other gradient based ICA.

6 Conclusions

The BFO is used to estimate the independent components from their observed mixtures which is tested using several examples. From the simulation results it is very clear that the BFOICA algorithm has faster convergence and better mean square error performance than as compared to the CGAICA algorithm. In comparison to the fast ICA algorithm it has good MSE. The permutation ambiguity present in ICA techniques is resolved by use of the BFOICA algorithm if we have relative knowledge of the statistical characteristics of the signals to be estimated. However care should be taken while adjusting the parameters for bacteria foraging optimization so that premature convergence in a local optimum does not occur.

7 References

[1] P. Comon, "Independent Component Analysis-A new concept?" *Signal Processing*, vol-36, pp.287-314,1994.

[2] H.Sawada, R.Mukai, S.Araki and S.Makino, "Solving Permutation and Circularity problem in Frequency-Domain Blind Source Separation," Proc. International Conf. on ICA 2004, Japan.

[3] M.Yoshioka and S. Omatu, "Signal separation method using genetic algorithms," in *Proc. IEEE Int. Joint Conf. Neural Networks*, vol.2, pp.909-912, 1998.

[4] Y.Tan and J.Wang, "Nonlinear Blind Source Separation Using Higher Order Statistics and a Genetic Algorithm," *IEEE Trans. On Evolutionary Computation*, vol.5, No.6, pp.600-611, Dec. 2001.

[5] F.Rojas, C.G.Puntonet, M.Rodriguez-Alvarez, I.Rojas, and R.Martin-Clemente, "Blind Source separation in post-nonlinear mixtures using competitive learning, simulated annealing and a genetic algorithm," *IEEE Trans. On Systems, Man and Cybernatics –Part C: Applications and Reviews*, vol.34,no.4, pp407-416, November 2004.

[6] R.Palaniappan and C.N.Gupta, "Genetic Algorithm based independent component analysis to separate noise from Electrocardiogram signals," *Proc. IEEE* , 2006

[7] K.M.Passino, "Biomimicry of Bacterial Foraging for distributed optimization and control," *IEEE Control Syst. Mag.*, vol.22, no.3, pp52-67, Jun.2002.

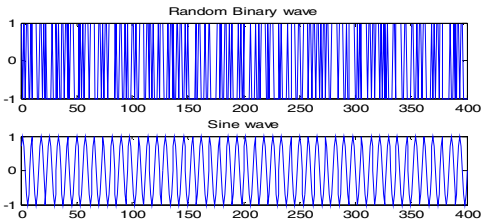
[8] S.Mishra, "A hybrid least square-fuzzy bacteria foraging strategy for harmonic estimation," *IEEE Trans. Evol.*

Comput., vol.9, no.1, pp.61-73,Feb.2005.

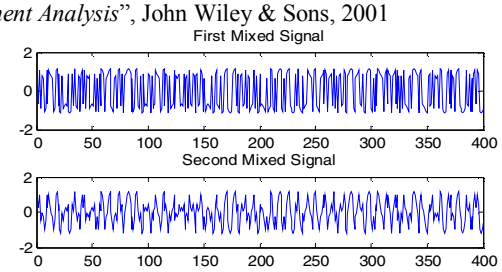
[9] S.Mishra and C.N.Bhende, "Bacterial Foraging Technique-Based Optimized Active Power Filter for Load Compensation," *IEEE Trans. Power Del.*, vol.22, no.1, pp.457-465, Jan.2007.

[10] D.P.Acharya, G.Panda and Y.V.S. Lakshmi, "Constrained Genetic Algorithm based Independent Component Analysis," Accepted for presentation in IEEE Congress on Evolutionary Computation, Singapore, Sep. 2007.

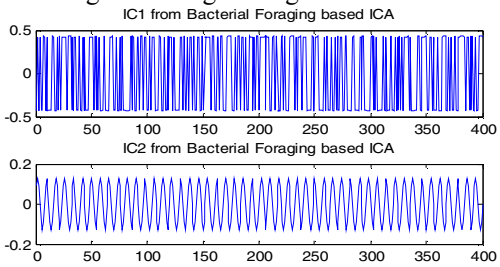
[11] A.Hyverinen., J.Kahrunen. and E. Oja, " *Independent Component Analysis*", John Wiley & Sons, 2001



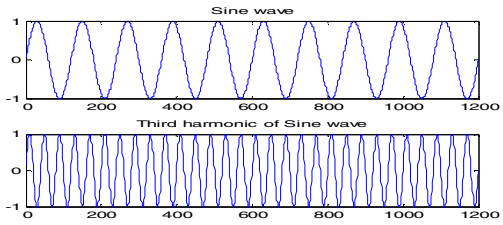
"Figure 1. Original Signals"



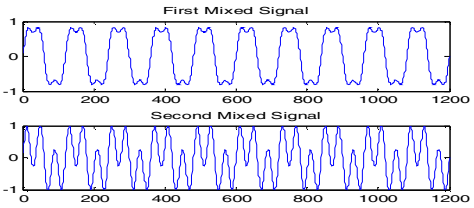
"Figure 2. Mixture of random binary and sine wave"



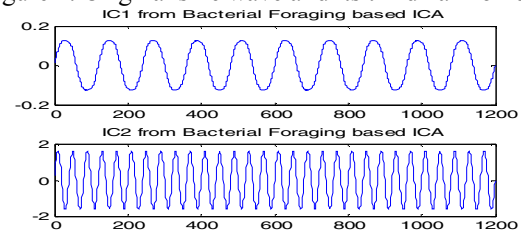
"Figure 3. The recovered independent components"



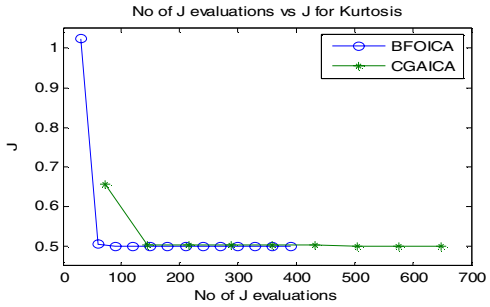
"Figure 4. Original sine wave and its third harmonic"



"Figure 5. Mixture of sine wave and its third harmonic"



"Figure 6 Recovered Independent Components"



"Figure 7. Minimum value of the nutrient function vs number of its evaluations for BFOICA (for kurtosis) and CGAICA"

Table 1 (MSE Comparison for different ICA algorithms)

Algorithm	MSE	
	Random Binary	Sine wave
Fast ICA	2.5004×10^{-9}	8.58×10^{-2}
CGAICA	2.1811×10^{-8}	8.58×10^{-2}
BFOICA	3.084×10^{-10}	8.58×10^{-2}