Proceedings of the International conference on Computing Technology and Information Management, Dubai, UAE, 2014

Biogeography-based Optimization Algorithm for Independent Component Analysis

Jehad Ababneh [a] [*], Jorge Igual [b]
[a] *A'Sharqiyah University, Ibra, Oman*
ababnehj@asu.edu.om
[*] *On sabbatical leave from Jordan University of Science and Technology, Irbid, Jordan*
[b] *Universidad Politecnica Valencia, Departmento Comunicaciones, Alcoy (Alicante), Spain*

ABSTRACT

Independent component analysis (ICA) is a signal processing technique that can be used to extract meaningful components from a dataset. Biogeography based optimization (BBO) algorithm is a recently developed stochastic optimization algorithm. In this paper, we report the use of the BBO algorithm to optimize a contrast function that measures the statistical independence of the recovered components in order to implement the ICA technique. The use of the BBO to implement the ICA technique is demonstrated on two benchmark data sets. The achieved results of using the BBO in the ICA technique are compared to that of the Fast ICA algorithm and using the particle swarm optimization (PSO) algorithm, and the differential evolutionary (DE) algorithm for ICA. Experimental results show that the BBO algorithm outperforms the Fast ICA and the DE algorithms in terms of the signal to interference ratio (SIR) of the recovered components while it outperforms the PSO algorithm in terms of the convergence speed. To both improve the convergence speed and the quality of the recovered components, the BBO and the PSO algorithms are jointly used.

KEYWORDS

Independent component analysis, Optimization, Biogeography based optimization algorithm, Particle swarm optimization, Evolutionary algorithm, FastICA algorithm.

1 INTRODUCTION

Independent component analysis (ICA) is a stochastic signal processing technique that can be used to extract meaningful components from a dataset or retrieve signals (sources) from their mixture [1-3]. The ICA technique has been widely used in many areas of science and engineering. The ICA relies on the principle assumption that all sources are independent of one another. Therefore, to implement the ICA technique, an objective or contrast function needs to be formulated to measure the statistical independence of the components. Then, an optimization technique is used to optimize this objective function such that the recovered components are as closely statistically independent from one another as possible. Therefore, the ICA technique can be formulated as an optimization problem.

The FastICA algorithm is an efficient algorithm for independent component analysis invented by Hyvarinen [1]. The algorithm maximizes non-Gaussianity as a measure of statistical independence based on a fixed-point iteration scheme. There are many other optimization algorithms that can be used to optimize the objective function in the ICA technique.

Differential evolutionary (DE) algorithm [4-5] is a popular optimization algorithm that has been developed to specifically search for global optima and avoids any local solutions. The DE algorithm uses mutation and crossover operations to search for the optimal solution. Another technique is particle swarm optimization (PSO) [6-8]. The PSO algorithm applies social interactivity between the different areas of the solution space to search for the optimum solution. The PSO algorithm is simpler in terms of coding than the DE algorithm. In addition, the PSO algorithm does not use complicated operations such as the mutation and crossover operations used in the DE algorithm. However, the PSO algorithm does not have strong exploitation ability and could be trapped in local optima.

A new promising stochastic optimization algorithm called biogeography-based optimization (BBO) has been recently developed [9-12]. The

BBO algorithm uses migration operations to share information between solution candidates in order to search for the optimum solution. The BBO algorithm has strong exploitation ability and has been successfully applied in many engineering areas [13-18].

In this paper, we report the use of the BBO algorithm to optimize a simple contrast function in the ICA technique. To the best of our knowledge, the BBO algorithm has never been used for ICA. The use of the BBO algorithm for the ICA is compared with the FastICA algorithm and the use of the PSO and DE algorithms for the ICA technique. Experimental results show that the BBO algorithm outperforms the Fast ICA and the DE algorithms in terms of the signal to interference ratio (SIR) of the recovered components while it outperforms the PSO algorithm in terms of the convergence speed. To both improve the convergence speed and the quality of the recovered components, the BBO and the PSO algorithms are jointly used.

The rest of this paper is organized as follows: an overview of the ICA and the BBO algorithms are provided in section 2 and section 3, respectively. Simulation results of using the BBO for ICA and performance comparison with the FastICA, PSO and DE are given in section 4. The conclusions of this paper are given in Section 5.

2 INDEPENDENT COMPONENT ANALYSIS

Independent Component Analysis (ICA) and its most popular application blind source separation (BSS) consists on the recovering of some independent sources up to a permutation, scaling and sign factor starting from instantaneous linear mixtures of them. The only assumption is that these sources are statistically independent. The formulation is:

$$\mathbf{x} = \mathbf{A}\mathbf{s} \tag{1}$$

where $\boldsymbol{x}$ is the observed vector, **s** is the source vector and **A** the mixing matrix.

This simple model has been extended to other kind of models, e.g., convolutive, nonlinear or noisy mixtures and many extensions can be found in the literature. It has been applied successfully in many real applications in spite of the very restrictive independence assumption. In addition, basic BSS is focused in the Blind term, i.e., in the fact that we have no prior information about the sources or mixing process.

The classical algorithms consider only the independence of the source signals. This condition implies that the joint distribution is the product of the marginal, i.e., $p(\mathbf{s}) = \prod_i p(s_i)$. Traditional solutions are based on the minimization-maximization of a function that measures the statistical independence of the recovered signals. As we do not know anything about the sources (their pdf are unknown), a pure statistical analysis of the problem is difficult and it is not possible to guarantee exactly the independence of the recovered signals. Some extensions including prior information have been proposed, depending on the application under consideration, e.g., see [2]; in this case, the problem is no more called blind.

In order to approximate the statistical independence of the sources, many approaches have been proposed, e.g., [3] maximizes contrast functions derived from mutual information, [19] presents an algebraic solution based on joint digitalization of fourth order cumulant matrices, [20] is based on information theory concepts. In all cases, the final algorithm involves the estimation of higher order moments since the independence requires that cross-cumulants must be zero. In practice, second order information is exhausted in a previous whitening step and after that, the higher order components are introduced directly or through non-linear transformations in order to estimate the remaining unknown orthogonal matrix. The de-mixing matrix is obtained as the product of this two matrices, the whitening one that decorrelates the signals and the orthogonal one that rotates them up to obtain the most independent sources. In mathematical notation:

$$\mathbf{y} = \mathbf{B}\mathbf{x} = \mathbf{U}\mathbf{W}\mathbf{x} \tag{2}$$

where **y** is the recovered (estimated) sources, **W** is the whitening matrix and **U** is the orthogonal matrix. Regardless of the contrast function we use, the final problem becomes an optimization problem in which the elements of the matrix **U** are searched such that the cost function is optimized in some sense. In this paper, the kurtosis is used in a contrast function to measure the independence. The kurtosis of a random variable *u* is defined as:

$$k(u) = E[u^4] - 3E^2[u^2] \qquad (3)$$

It is a normalized version of the fourth order moment. It is zero for Gaussian random variables and non-zero for non-Gaussian distributions. It is negative for flatter distributions such as the uniform random variable; these distributions are called sub-Gaussian or mesokurtic. It is positive for heavy tailed distributions such as the Laplacian random variable; in these cases, they are called super-Gaussian or leptokurtic. In summary, kurtosis can be used to measure the non-Gaussianity of the distribution. Knowing that when two random variables are mixed the distribution becomes more Gaussian, therefore, we can use the absolute value of the kurtosis or their square value as a contrast function to be maximum when the sources are unmixed. Hence, we may define a contrast function as

$$J_{kurtosis} = \sum_i k^2(y_i) \qquad (4)$$

where y_i is the *ith* recovered source. This contrast function is an approximate measure of the independence. Note that to enforce the independence restriction we should include all the higher order cumulants, not only fourth order (remember that second order are used to whitening the signals and usually third order statistics are not considered since the sources are symmetric). For practical considerations, this is not possible, not only because the high computational cost, but because an accurate estimation of higher order statistics higher than fourth is not possible.

3 THE BIOGEOGRAPHY-BASED OPTIMIZATION ALGORITHM

Biogeography-based optimization (BBO) is a recently developed algorithm for global optimization [9-12]. The algorithm is based on the biogeography theory which studies the geographical distribution of biological organisms in a habitat. To measure the suitability of a given habitat for residence for biological species, the term habitat suitability indexes (HSI) is used. A collection of variables called suitability index variables (SIVs) is used to characterize the habitability of a given habitat. In the BBO algorithm, an N-dimension candidate solution of the optimization problem is analogous to a habitat with N SIVs. The goodness of any solution is measured based on a given objective or fitness function that is analogous to the HSI.

To improve the candidate solutions, the migration and the mutation operations are used. The migration operator is used to change existing solutions. On the other hand, the mutation process is a probabilistic operator that randomly modifies habitat SIVs. Therefore, a habitat H is a collection of N SIVs that follow migration and mutation processes in order to improve its HSI. Details of these operations are in the original paper of the BBO algorithm [9].

The basic procedure of the BBO algorithm is described as follows:

Step1: The fitness (contrast) function is defined and a number of habitats each represented using an N-dimension vector is randomly initialized, where N represents the dimension of the optimization problem.

Step2: The fitness function (HSI) is evaluated for all habitats and then sorted from best to worst.

Step3: The HSI of each habitat is mapped to species count.

Step4: The immigration rate λ and emigration rate μ for each habitat are calculated based on its species count according to Equations (5) and (6).

Step 5: The migration and mutation algorithms are performed for all habitats.

Step 6: The termination criterion is checked and if it is not met, then go back to step 2.

The migration and mutation operations are briefly described in the following subsections.

3.1 Migration

In the migration operation, habitats share information based on the emigration rate μ and the immigration rate λ of each habitat. A habitat with few species (low HSI) is a bad solution of the optimization problem and is assigned a low μ and high λ. A habitat with more species (high HSI) is a good solution of the optimization problem and is assigned a high μ and low λ. Each habitat has its own μ and λ which are functions of the number of species S (species count) in the habitat and are given as:

$$\lambda(s) = I(1 - \frac{s}{s_{\max}}) \qquad (5)$$

$$\mu(s) = E(\frac{s}{s_{\max}}) \qquad (6)$$

where $s_{\max}$ is the largest possible number of species that a habitat can support; *I* is the maximum value of the immigration rate λ; and *E* is the maximum value of the emigration rate μ. Each solution is modified based on other solutions. The decision to modify a solution is controlled by the modification probability, P_{mod} which is a user-defined parameter which normally set to unity. If a solution is chosen to be modified, then its rate of immigration is used to decide whether or not to modify each SIV in this solution. If an SIV is selected to be modified, then the emigration rates μ of the other solutions are used to probabilistically determine which of the solutions should migrate an SIV to a chosen solution. A pseudo code of the migration operator is shown in Figure 1.

3.2 Mutation

In the BBO, the mutation is used to increase diversity through the population to obtain good solutions. Mutation modifies a habitat's SIV randomly based on mutation rate *m*. The mutation rate *m* of each solution is given as:

$$m(s) = m_{\max}\left(1 - \frac{P_s}{P_{\max}}\right) \qquad (7)$$

where $m_{\max}$ is maximum mutation rate which is a user defined parameter, P_s is the probability of species count, and $P_{\max}$ is the maximum probability of species count. Therefore, the good solution is less probably to mutate to another solution, while the bad solution is more probably to mutate to another solution. This opposite relationship comes into view in the mutation rate equation. Without this change, the solutions that have high probabilities tend to be more dominant in the population. This approach provides both high and low HSI solutions to mutate, in order to obtain better solution of the problem.

The mutation process gives the solutions the chance to be improved, but it can also ruin them. So, if mutation ruins the habitat that has the best solution, an elitism approach is used to keep it and revert back to it, if needed. For this reason, mutation process is applied as a high-risk process and is applied on both weak and good solutions. Since the medium quality solutions are already in improving stage, it is better to avoid mutating them. Performing mutation on a selected solution is done by replacing it with a new solution that is randomly generated. Otherwise, any other mutations that have been implemented in GA can be implemented for the BBO.

To prevent the best solutions in the population from being corrupted by migration and mutation step, some sort of elitism is incorporated to BBO algorithm. In this step, the best solutions are saved before beginning in modification. After completion of the modifications, the worst solutions are replaced with the saved solutions.

4 SIMULATION RESULTS

In this section, we simulate the use of the BBO algorithm to optimize the contrast function $J_{kurtosis} = \sum_i k^2(y_i)$ given in Eq. 4 in section 2 based on the kurtosis as a measure of statistical independence in order to implement the ICA technique. It is worth mentioning that we have used the PSO algorithm for ICA in our previous work based on mutual information contrast

```
For $i = 1$ to Ns (where Ns is the number of habitatas)
   If rand > $P_{\text{mod}}$ Continue
   End
          For $j = 1$ to $N$ (where $N$ is the number of variables)
                Select an SIV in $H_i$ with probability $\lambda_i$
                If an SIV is selected to be modified
                   For $k = 1$ to Ns
                        Select $H_k$ with probability $\mu_k$
                        If $H_k$ is selected
                                Replace the SIV$_j$ in $H_i$ with an SIV$_j$ from $H_k$
                        End
                   End
                End
          End
End
```

Figure 1 A Pseudo Code of the Migration Process Algorithm

function [21-22]. However, in this paper we use simpler contrast function. The simulation is made using two sets of benchmark data available in the ICALAB [23]. The first set is five sparse (smooth bell-shape) sources that are approximately independent. The second set is five fourth order colored sources with a distribution close to Gaussian. The two sets are mixed using two different randomly generated mixing matrixes before applying the ICA technique. Then, the ICA technique is applied to recover the original signals (sources) from their mixture. It is worth mentioning that a pre-processing process called whitening is normally performed as the first step in ICA to impose the decorrelation of the sources. Furthermore, it is assumed that the sources are zero mean and unit variance; otherwise they can be made to have these properties.

In this work, we compare the use of the BBO algorithm for ICA with the FastICA algorithm and with the use of the PSO and DE algorithms for ICA. The comparison is made based on the signal to interference ratio (SIR) of the recovered sources. The signal to interference value for the i^{th} recovered component y_i is defined as:

$$SIR(y_i) = 10\log_{10}\left[\frac{\left|\left\langle y_i, s_i\right\rangle\right|}{\left\langle y_i, y_i\right\rangle - \left|\left\langle y_i, s_i\right\rangle\right|}\right] \tag{8}$$

where $\langle f, g\rangle$ is the standard inner product of the two vectors f and g. In addition, a comparison of the average convergence speed of the BBO, PSO and the DE to optimize the $J_{kurtosis}$ contrast function is presented.

In our simulations, the parameters for the BBO were selected as follows: maximum immigration rate and maximum emigration rate of 1 and maximum mutation rate of 0.01. For PSO, constriction global topology PSO algorithm with constriction factor of 0.72984 was used [8]. We used, DE/rand/1/bin scheme with weighting factor of $F = 0.5$ and a crossover constant of $CR = 0.5$ for the DE algorithm. In all algorithms a population size of 25 was used.

All algorithms were run ten times for the same set of data and using the same mixing matrix. During the search, the main objective is to find the elements of the de-mixing matrix **U** that optimize the $J_{kurtosis}$ contrast function. Table 1 shows the average SIR of the recovered sources using the BBO, FasICA, PSO and DE algorithms for the first set of data (the sparse sources).

Table 1 Comparison of the Average SIR (dB) for the Sparse Signals Data Set

Signal	SIR (dB)				
	BBO	FastICA	PSO	DE	BBOPSO
1	43.0100	30.2796	55.6082	16.9834	**56.4218**
2	34.2693	27.2487	**36.3291**	15.9347	36.0879
3	42.2575	33.2825	52.2819	16.7115	**52.7823**
4	33.3364	19.5224	**35.6356**	16.0105	35.4270
5	36.5452	20.144	**40.6399**	16.7243	40.5017

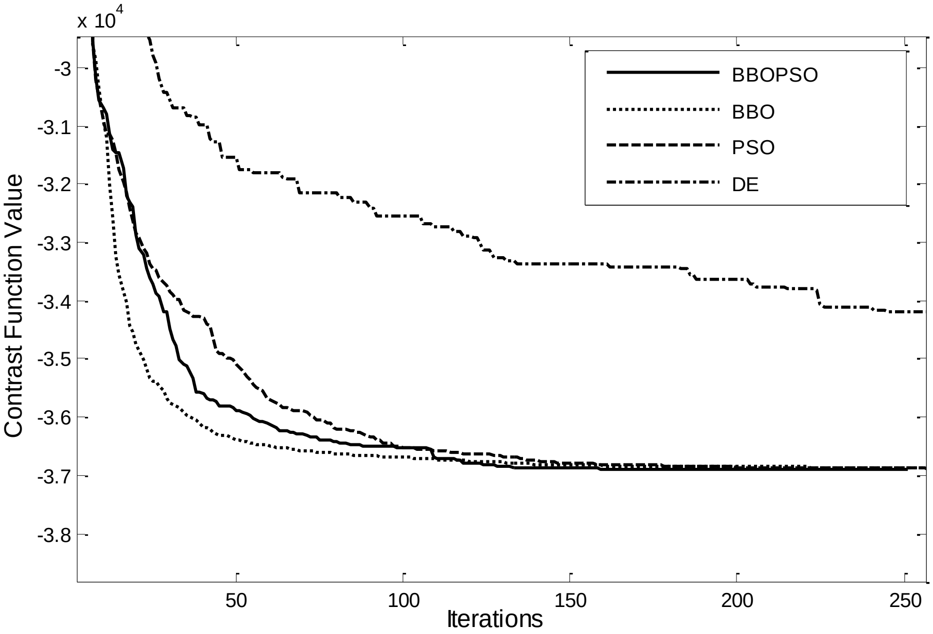

Figure 2. Convergence curves for the value of the contrast function for the sparse data set

Table 2 Comparison of the Average SIR (dB) the Fourth Order Colored Sources Data Set

Signal	SIR (dB)				
	BBO	FastICA	PSO	DE	BBOPSO
1	**14.5819**	Algorithm failed	14.4822	13.9383	14.4910
2	17.2179	Algorithm failed	17.5368	11.8345	**17.5695**
3	19.4137	Algorithm failed	19.7486	12.3451	**19.8158**
4	2.6924	Algorithm failed	2.6516	**3.4283**	2.6417
5	3.0989	Algorithm failed	3.0456	**4.4324**	3.0291

As can be seen in Table 1, the average SIR of the recovered signals using the BBO algorithm for ICA are higher than that of the FastICA and using the DE for ICA but lower than that using the PSO for ICA. However, the use of BBO for ICA has a better average convergence speed as shown in Figure 2.

The achieved results of using the BBO for ICA for this data set could be attributed to the fact that the BBO algorithm has good enough exploitation ability but does not have good enough exploration ability compared to that of the PSO algorithm. To simultaneously achieve better convergence speed and high SIR, we simultaneously used the PSO and the BBO for ICA. That is, in each iteration, we run independently the PSO and the BBO algorithm to optimize the contrast function. Then, at the end of each iteration, the fitness values of the corresponding candidate solutions of both algorithms are compared, and the candidate solution with better fitness value will be used in the next iteration in both algorithms. In this scenario, it is assumed that the candidate solutions in both algorithms are numbered or labeled and the numbering or labeling is tracked. The simultaneous use of BBO and PSO for ICA achieved higher convergence speed and slightly better SIR than that of the PSO algorithm as shown in Figure 2 and Table1, respectively.

The average SIR of the recovered sources using the BBO, FasICA, PSO, DE and using the BBOPSO algorithms for the second set of data (the fourth order colored sources) are given in Table 2.

As given in Table 2, the average SIR of the recovered signals achieved using the BBO algorithm for ICA are better than that of the FastICA and using the DE for ICA but are comparable to that of the PSO and the BBOPSO. In fact, the FastICA algorithm completely failed to recover any of the original sources

5 CONCLUSIONS

In this paper, the use of the BBO algorithm for the ICA technique is demonstrated on two benchmark data sets and is compared with that of the Fast ICA algorithm and using the PSO and the DE algorithms for ICA. The achieved results of using the BBO in the ICA showed that the BBO algorithm outperforms the Fast ICA and the DE algorithm in terms of the signal to interference ratio (SIR) of the recovered components while it outperforms the PSO algorithm in terms of the convergence speed. To both improve the convergence speed and the quality of the recovered components, the BBO and the PSO algorithms were jointly used.

7 REFERENCES

1. Oja, E., Hyvarinen A., Karhunen, J.: Independent Component Analysis. John Wiley & Sons, (2001).
2. Llinares, R., Igual, J.: Application of constrained independent component analysis algorithms in electrocardiogram arrhythmias. Artificial Intelligence in Medicine, 47, 121-133 (2009).
3. Comon, P.: Independent Component Analysis, a new concept?. Signal Processing, 36, 287-314 (1994).
4. Onwubolu, G., Babu, B.: New Optimization Techniques in Engineering. Berlin, Germany, Springer-Verlag, (2004).
5. Price, K., Storn, R.: Differential Evolution : Numerical Optimization Made Easy. Dr. Dobb’s Journal, 22 , 18–24 (1997).
6. Kennedy, J., Eberhart, R. C.: Particle swarm optimization. In: Proc. IEEE Conference on Neural Networks IV. pp.1942-1948, Piscataway, NJ (1995).
7. Shi, Y., Eberhart, R. C.: Parameter selection in particle swarm optimization. In: Proc. (EP98) Conference of Evolutionary Programming VII, pp. 591–600 (1998).
8. Clerc, M., Kennedy, J.: The particle swarm- explosion, stability, and convergence in a multidimensional complex space. IEEE Transactions on Evolutionary Computation. 6(1) 58–73 (2002).
9. Simon, D.: Biogeography-based optimization. IEEE Transactions on Evolutionary Computation 12 (6) 702–713 (2008).
10. Simon, D., Ergezer, M., Du, D., Rarick, R.: Markov models for biogeography-based optimization. IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics. 41 (1) 299-306 (2011).
11. Du, D., Simon, D., Ergezer, M.: Biogeography-based optimization combined with evolutionary strategy and immigration refusal. In: Proc. IEEE Conference on Systems, Man, and Cybernetics. pp. 1023-1028, (2009).
12. Ma, H., Ni, S., Sun, M.: Equilibrium species counts and migration model tradeoffs for biogeography-based optimization. In: Proc. IEEE Conference on Decision and Control. pp. 3306–3310 Shanghai (2009).
13. Johal, N., Singh, S., Kundra, H.: A hybrid FPAB/BBO algorithm for satellite image classification. International Journal of Computer Applications, 6 (5) 31-36 (2010).
14. Singh, U., Kumar, H., Kamal, T.: Design of Yagi-Uda antenna using biogeography based optimization. IEEE Transactions on Antennas and Propagation, 58 (10) 3375-3379 (2010).
15. Mukherjee, R., Chakraborty, S., Samanta, S.: Selection of wire electrical discharge machining process parameters using non-traditional optimization algorithms. Applied Soft Computing, 12 (8) 2506-2516 (2012).
16. Roy, P., Mandal, D.: Quasi-oppositional biogeography-based optimization for multi-objective optimal power flow. Electric Power Components and Systems, 40 (2) 236-256 (2011).
17. Wang, L., Xu, Y.: An effective hybrid biogeography-based optimization algorithm for parameter estimation of chaotic systems. Expert Systems with Applications, 38 (12) 15103-15109 (2011).

18. Wang, G., Guo, L., Duan, H., Liu, L., Wang, H.: Dynamic Deployment of Wireless Sensor Networks by Biogeography Based Optimization Algorithm. Journal of Sensor and Actuator Networks, 1 (2) 86-96 (2012).
19. Cardoso, J.F., Souloumiac, A.: Blind beamforming for non Gaussian signals. In: Proc. Inst. Elec. Eng. pt. F 140 pp. 362-370 (1993).
20. Bell A.J., Sejnowsky, T.J.: An information-maximization approach to blind separation and blind deconvolution. Neural Computation 7 1129-1159 (1995).
21. Igual, J., Ababneh, J., Llinares, R., Igual, C.: Using Particle Swarm Optimization for minimizing mutual information in Independent Component Analysis. In: Proc. 11th International Work-Conference on Artificial Neural Networks, IWANN 2011 Torremolinos-Málaga, Spain (2011).
22. Igual, J., Ababneh, J., Llinares, R., Miro-Borras, J., Zarzoso, V.: Solving Independent Component Analysis Contrast Functions with Particle Swarm Optimization. In: Proc. 20th ICANN, Thessaloniki, Greece (2010).
23. Cichocki, A., Amari, S., Siwek, K., Tanaka, T., Phan, A., Zdunek, R.: ICALAB – MATLAB Toolbox Ver. 3 for signal processing, http://www.bsp.brain.riken.jp/ICALAB.