Hindawi Publishing Corporation Mathematical Problems in Engineering Volume 2016, Article ID 5481602, 8 pages http://dx.doi.org/10.1155/2016/5481602



## Research Article

# **Glowworm Swarm Optimization and Its Application to Blind Signal Separation**

# Zhucheng Li<sup>1,2</sup> and Xianglin Huang<sup>1</sup>

<sup>1</sup>College of Computer Science, Communication University of China, Beijing 100024, China <sup>2</sup>Business College, Beijing Union University, Beijing 100025, China

Correspondence should be addressed to Zhucheng Li; lizhucheng@cuc.edu.cn

Received 31 December 2015; Revised 27 March 2016; Accepted 7 April 2016

Academic Editor: Marco Mussetta

Copyright © 2016 Z. Li and X. Huang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Traditional optimization algorithms for blind signal separation (BSS) are mainly based on the gradient, which requires the objective function to be continuous and differentiable, so the applications of these algorithms are very limited. Moreover, these algorithms have problems with the convergence speed and accuracy. To overcome these drawbacks, this paper presents a modified glowworm swarm optimization (MGSO) algorithm based on a novel step adjustment rule and then applies MGSO to BSS. Taking kurtosis of the mixed signals as the objective function of BSS, MGSO-BSS succeeds in separating the mixed signals in Matlab environment. The simulation results prove that MGSO is more effective in capturing the global optimum of the objective function of the BSS algorithm and has faster convergence speed and higher accuracy, compared with particle swarm optimization (PSO) and GSO.

#### 1. Introduction

"Cocktail party" problem can be seen as a classic example of blind signal separation: imagine being at a friend's party where, during your conversation with your friend, even though the sounds that reach your ears are a complicated mix of music, other people talking, wine glasses tinkling, and so on, you are able to understand your friend and enjoy the music at the same time. It is the task of blind signal separation (BSS) to recover unknown independent source signals obtained from sensors. The BSS technology has received considerable attention in recent years because of its significant potential applications such as sonar and radar signal processing [1, 2], wireless communication [3], geophysical exploration [4, 5], biomedical signal processing [6, 7], speech and image processing [8, 9], and machine fault diagnosis [10, 11]. BSS has two important components: the objective function and the optimization algorithm. The objective function is responsible for determining the statistical independence of separation signals, and the optimization algorithm is to ensure that the objective function value reaches its peak in subsequent updates. The convergence speed and accuracy of BSS mainly rely on the latter. Therefore, how to choose an appropriate algorithm is the crucial challenge of BSS.

Conventional optimization algorithms for BSS are based on gradient techniques, yet these methods would get a "poor" solution unless suitable initial parameters are given. However, it is very difficult to select these parameters because of the blind hypothesis. In particular, these algorithms cannot be used when the objective function is discontinuous and nondifferentiable. To solve the above problems, swarm-based algorithms have been gradually applied to BSS in the past few years, such as genetic algorithms (GA) [12, 13], particle swarm optimization (PSO) [12, 14], and artificial bee colony (ABC) [15]. Swarm-based algorithms belong to a family of natureinspired, population-based optimizations and the behavior of their agents is inspired by biological swarms like ants, fish, bees, frogs, and bacteria, which interact in accordance with certain behavioral law to cooperatively achieve some necessary tasks. Compared with conventional gradient-based approaches, these techniques for BSS are characterized by higher accuracy, efficiency, and robustness. However, there is

room for improvement of the performance of these optimization algorithms in terms of their tendency to fall into local optimum, convergence rate, and computational accuracy.

The GSO acronym can stand for two different swarm-based optimizations: Genetical Swarm Optimization [16, 17] and Glowworm Swarm Optimization. Genetical Swarm Optimization is a hybrid evolutionary algorithm that combines the well-known PSO and Genetic Algorithm (GA). Glowworm Swarm Optimization [18–23] proposed by Krishnanand and Ghose imitates the behavior that a glowworm carries a luminescence quantity called luciferin along with itself to exchange information with companions. GSO in this paper is only used for Glowworm Swarm Optimization.

GSO can effectively avoid missing the optimal solution because of intelligent changes of the decision radius and is very competent in capturing the global optimum of the objective function in finite-dimensional vector space. At present, GSO has been successfully applied in various fields, such as vehicle routing problem [24], dock scheduling problem [25], and wireless sensor networks [26]. Despite the above-mentioned advantages, the standard GSO has a trade-off between convergence speed and accuracy because of the fixed step-size (the suggested step-size is 0.03).

In this paper, we present a modified GSO (MGSO) algorithm to conquer the above defects and then apply MGSO to BSS; finally, the experiment proves the effectiveness of MGSO-BSS. The remainder of this paper is organized as follows: the next section gives a complete presentation of the basic GSO and describes the proposed methods; Section 3 introduces the working principle of BSS; in Section 4, seeking mode of new BSS algorithm based on MGSO is described; in Section 5, we carry out experiments to evaluate MGSO-BSS and analyze the simulation results; the last section contains the concluding remarks on this work.

#### 2. Basic Glowworm Swarm Optimization

2.1. Algorithm Representation. In GSO, a swarm of glowworms are initially deployed randomly in the solution space. Each glowworm represents a solution of objective function in the search space and carries a certain quantity of luciferin along with it. The luciferin level is associated with the fitness of the agent's current position. The brighter individual means a better position (is a better solution). Using a probabilistic mechanism, each agent can only be attracted by a neighbor whose luciferin intensity is higher than its own within the local-decision domain and then moves towards it. The density of a glowworm's neighbors affects its decision radius and determines the size of its local-decision domain: when the neighbor-density is low, the local-decision domain will enlarge in order to find more neighbors; otherwise, it will reduce to allow the swarm split into smaller groups.

The above process is repeated until the algorithm satisfies the termination condition. At this point, the majority of individuals gather around brighter glowworms. Briefly, the GSO involves five main phases: luciferin-update phase, neighborhood-select phase, moving probability-computer phase, movement phase, and the decision radius update phase.

#### 2.2. Basic GSO Algorithm

2.2.1. Luciferin-Update Phase. The luciferin update depends on the fitness value and previous luciferin value, and its rule [18–23] is given by

$$l_i(t+1) = (1-\rho)l_i(t) + \gamma \operatorname{Fitness}(x_i(t+1)). \tag{1}$$

Here,  $l_i(t)$  denotes the luciferin value of glowworm i at time t,  $\rho$  is the luciferin decay constant,  $\gamma$  is the luciferin enhancement constant;  $x_i(t+1) \in R^M$  is the location of glowworm i at time t+1, and Fitness( $x_i(t+1)$ ) represents the value of the fitness at glowworm i's location at time t+1.

2.2.2. Neighborhood-Select Phase. Neighbors  $N_i(t)$  [18–23] of glowworm i at t time consist of the brighter ones and can be written as

$$N_i(t) = \left\{ j : d_{ii}(t) < r_d^i(t); \ l_i(t) < l_i(t) \right\}. \tag{2}$$

Here,  $d_{ij}(t)$  represents the Euclidean distance between glowworms i and j at time t, and  $r_d^i(t)$  represents the decision radius of glowworms i at time t.

2.2.3. Moving Probability-Computer Phase. A glowworm uses a probability rule to move towards other glowworms having higher luciferin level. The probability  $P_{ij}(t)$  [18–23] of glowworm i moving towards its neighbor j can be stated as follows:

$$P_{ij}(t) = \frac{l_j(t) - l_i(t)}{\sum_{k \in N_i(t)} l_k(t) - l_i(t)}.$$
 (3)

2.2.4. Movement Phase. Suppose glowworm i selects glowworm  $j \in N_i(t)$  with  $P_{ij}(t)$ ; the discrete-time model of the movement of glowworm i is given by (4) as in [18–23]

$$x_{i}(t+1) = x_{i}(t) + s\left(\frac{x_{j}(t) - x_{i}(t)}{\|x_{j}(t) - x_{i}(t)\|}\right).$$
(4)

Here,  $\|\cdot\|$  represents the Euclidean norm operator, and s is the step-size.

2.2.5. Decision Radius Update Phase. In each update, decision radius of glowworm i is given as follows:

$$r_{d}^{i}(t+1) = \min \left\{ r_{s}, \max \left\{ 0, r_{d}^{i}(t) + \beta \left( n_{t} - \left| N_{i}(t) \right| \right) \right\} \right\}.$$
 (5)

Here,  $\beta$  is a constant,  $r_s$  denotes the sensory radius of glowworm i, and  $n_t$  is a parameter to control the neighbor number. Figure 1 shows the sensory radius and decision radius of glowworm i.

2.3. Modified Glowworm Swarm Optimization (MGSO). For the standard GSO, if the fixed step-size is large, each glowworm covers a large jump (equal to step-size). Therefore,

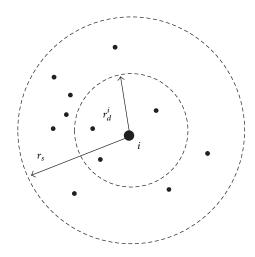


FIGURE 1: Sensory and decision radius of glowworm *i*.

these glowworms may move so fast as to miss the optimum solution in the updates. When the distance between a glowworm and its best neighbor is less than s in (4), the glowworm would oscillate. However, if the step-size decreases, the convergence rate becomes slow. Consequently, it is difficult to decide the most appropriate step-size. In this paper, the step-size is not fixed and varies for each glowworm in each iteration. Here, let the step-size of glowworm i be the function of update number t, a dynamic step-size strategy is proposed to accelerate the convergence speed in the early stage of search and improve the calculation accuracy in the later stage of search. The step-size function (showed in Figure 2) is expressed as follows:

$$s(t) = \mu e^{-\psi t} + \xi. \tag{6}$$

Here,  $\psi$  and  $\mu$  are positive factors, and  $\xi$  is a step minimum-threshold.

If the initial step-size in (6) is equal to or slightly bigger than the fixed step-size in (4), the real-time step-size in (6) would be smaller than that in (4) at the end. Therefore, the probability and amplitude of oscillation of the agent of MGSO near the optimal solution would be much smaller than those of the standard GSO. Particularly, when the parameter values in (6) are properly chosen, the oscillation can be ignored.

## 3. Blind Signal Separation

3.1. Mathematical Model. This section gives the basic formulations [27] of BSS and describes its major steps. The general system model for BSS is shown in Figure 3. In the figure,  $\mathbf{s}(k) = [s_1(k), s_2(k), \dots, s_n(k)]^T$  is an n-dimensional vector of unknown source signals and is instantaneously and linearly mixed by a random full-rank matrix  $\mathbf{A} \in \mathbf{R}^{m \times n}$ . Here, there is an assumption that each component of  $\mathbf{s}(k)$  is statistically independent.

The observed signals (also called mixed signals)  $\mathbf{x}(k) = [x_1(k), x_2(k), \dots, x_m(k)]^T$  can be expressed as a linear transformation of the source vector as follows:

$$\mathbf{x}(k) = \mathbf{A}\mathbf{s}(k) + \mathbf{n}(k). \tag{7}$$

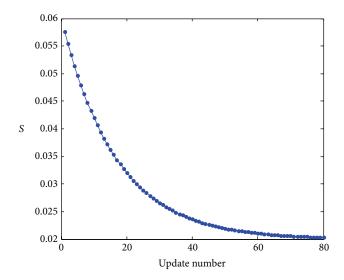


FIGURE 2: Curve of proposed step-size function.

Here,  $\mathbf{A} \in \mathbf{R}^{n \times m}$  is also unknown, and  $\mathbf{n}(k)$  is the *m*-dimensional vector of additive noise and is usually ignored.

The task of BSS is to obtain the separation matrix  $\mathbf{W} \in \mathbf{R}^{m \times n}$  based only on the observed signals by some algorithm. The recovery signals  $\mathbf{y}(k) = [y_1(k), y_2(k), \dots, y_n(k)]^T$  of source signals can be written as:

$$\mathbf{v}(k) = \mathbf{W}\mathbf{x}(k). \tag{8}$$

3.2. Ambiguities of Blind Signal Separation

*3.2.1. Amplitude of Separation Signal.* To clearly illustrate this problem, (7) can be written in the form

$$x_{i}(k) = \sum_{j=1}^{n} a_{ij} \mathbf{s}_{j}(k) = \sum_{j=1}^{n} \left(\frac{1}{\beta} a_{ij}\right) \cdot \left(\beta \mathbf{s}_{j}(k)\right). \tag{9}$$

Here,  $a_{ij}$  is the (i, j) element of mixing matrix **A**, and  $\beta > 0$  is a constant. The reason for this amplitude is that any scalar multiplier in the source  $s_j$  could always be cancelled by dividing the corresponding row  $a_i$  of **A** by the same scalar. Fortunately, this ambiguity of separation signal is insignificant in most applications.

3.2.2. Order of Separation Signal. Formula (7) can also be written in the form

$$\mathbf{x} = \mathbf{A}\mathbf{P}^{-1}\mathbf{P}\mathbf{s}.\tag{10}$$

Here, **P** is a permutation matrix. From (7), it can be concluded that **P** and its inverse  $\mathbf{P}^{-1}$  can be substituted in the model. The elements of  $\mathbf{P}\mathbf{s}$  are a set of the original independent variables, but in a different order. Then, the matrix  $\mathbf{A}\mathbf{P}^{-1}$  is just another unknown mixing matrix.

3.3. Measuring Algorithm Performance. The Central Limit Theorem, a classical result in probability theory, tells us that the distribution of a sum of several independent random

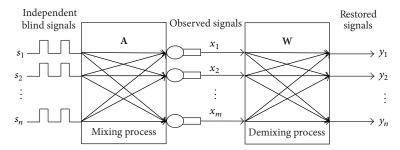


FIGURE 3: System model for BSS.

variables tends towards Gaussian distribution, under certain conditions. Thus, the distribution of a sum of several independent random variables is usually closer to the Gaussian distribution than any of the original random variables. According to this theory, performance of BSS algorithm can be measured by non-Gaussianity of the recovery signals, when the non-Gaussianity of signals reaches a maximum, the algorithm achieves high-quality separation of mixed signals. The classical measure of non-Gaussianity is kurtosis or the fourth-order cumulant [27]. The kurtosis of random variable y is classically defined as follows:

$$\operatorname{kurt}(y) = E[y^{4}] - 3(E[y^{2}])^{2}. \tag{11}$$

For most (but not quite all) non-Gaussian random variables, kurtosis is nonzero.

3.4. Signal Preprocessing. In general, kurtosis is difficult to compute from measured data as the data points required for reasonable accuracy are very large. To overcome this problem, the preprocessing steps need to be performed before the mixed signals are separated. This method can simplify BSS to the estimation of rotation angle of the joint probability density function (PDF). By this way, the computation cost can be reduced by approximately half. The preprocessing includes centering and whitening (sphering) of the data [27].

3.4.1. Centering. Subtracting its mean vector  $E[x_i]$  from the signal  $x_i$ , we can center the signal  $x_i$  into zero-mean variable. However, this does not mean that the mean could not be estimated. After estimating the mixing matrix **A** with centered data, we can complete the estimation by adding the mean vector of **s** back to the centered estimates of **s**. Centering is described as follows:

$$x_i = x_i - E\left[x_i\right]. \tag{12}$$

3.4.2. Whitening. Whitening is a critical procedure that reduces the number of parameters to be calculated. The observed signal  $\mathbf{x}$  is converted to a whitened vector  $\mathbf{v}$  with the help of a whitening matrix  $\mathbf{V}$  to ensure  $E[\mathbf{v}\mathbf{v}^T] = \mathbf{I}$  (identity matrix). The process can be mathematically expressed as follows:

$$\mathbf{v}(k) = \mathbf{V}\mathbf{x}(k) = \mathbf{V}\mathbf{A}\mathbf{s}(k). \tag{13}$$

There are many ways to whiten signals, such as principal components analysis (PCA) and singular value decomposition (SVD). The whitening matrix determined by the PCA approach is given as

$$\mathbf{V} = \mathbf{D}^{-1/2} \mathbf{E}^T. \tag{14}$$

Here, **D** is the diagonal matrix of eigenvalues of the covariance matrix  $\mathbf{C} = E\{\mathbf{x}(k)\mathbf{x}(k)^T\}$ , and **E** is the orthogonal matrix containing eigenvectors of **C**. After the mixed signals are whitened, instead of estimating the  $n^2$  coefficients of matrix **A**, we only need to estimate an orthogonal mixing matrix  $\mathbf{U} = \mathbf{V}\mathbf{A}$  which only has n(n-1)/2 parameters, about half the number of parameters in matrix **A**.

## 4. Seeking Mode of New BSS Algorithm Based on MGSO

It can be described as follows.

Step 1. Read the observed signals x, and then center and whiten them.

Step 2. Initialize parameters of  $\rho$ ,  $\gamma$ ,  $\beta$ ,  $n_t$ ,  $l_0$ , and s, and then generate a certain number of separation matrixes as glowworm individuals and initialize the position x(t) and decision radius  $r_d(t)$  of these glowworms, and then calculate the initial fitness value of each glowworm in the search space.

*Step 3.* Calculate the optimal location  $x_{\text{opt}}$  and the optimal fitness Fitness( $x_{\text{opt}}$ ) of these particles.

Step 4. Update luciferin l(t) according to (1), confirm the set of neighbors N(t) according to (2), and compute the probability of movement P(t) according to (3).

Step 5. Compute next position x(t + 1) and next decision radius  $r_d(t + 1)$  of each glowworm, and renew next step-size s(t + 1).

Step 6. Calculate the optimal fitness value of all glowworms. If this value is superior to  $Fitness(x_{opt})$ ,  $x_{opt}$  and  $Fitness(x_{opt})$  will be updated.

Step 7. Determine whether to satisfy the termination condition. If satisfied, jump out of the loop; otherwise, jump to Step 4.

TABLE 1: The values of the parameters of BSS-MGSO in this experiment.

Parameter	ρ	γ	β	$n_t$	$l_0$	ψ	μ	ξ
value	0.4	0.6	0.08	5	5	0.06	0.04	0.02

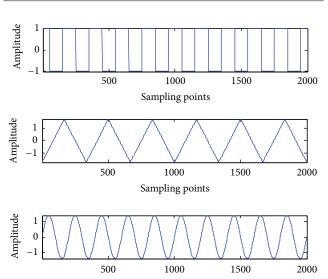


FIGURE 4: Source signals  $(s_1, s_2, s_3)$ .

Sampling points

## 5. Simulation Experiments

5.1. Experimental Environment and Parameter Setting. Three different digitized signals presented in Figure 4 are used in this simulation; these signals are mixed by the matrix

$$\mathbf{W} = \begin{pmatrix} 0.5669 & 0.9616 & 0.2359 \\ 0.5997 & 0.4416 & 0.7706 \\ 0.5913 & 0.6925 & 0.1600 \end{pmatrix}$$
 (15)

without delay as in formula (7) (**n** is ignored); the mixed signals are shown in Figure 5. The choice of parameters plays an important role in the performance of MGSO algorithm. The values of main parameters of MGSO are kept fixed in this experiment, and listed in Table 1.

5.2. The Results of Simulation and Analysis. Figure 6 is the restored signals obtained by GMSO after 15 updates; Figure 7 is two-dimensional map of  $\mathbf{W}_k(1,1)$  and  $\mathbf{W}_k(2,1)$  and approximately plots the track of glowworms seeking optimal solution. The fitness curves in Figure 8 mean that the standard PSO easily falls into a local optimum; the standard GSO has a good ability to search for the optimal value, but its convergence is not fast enough against others; MPSO has the best behavior on accuracy and speed. Figure 9 shows scatter plots of the source signal against its estimate using the MPSO algorithm. As can be seen from Figure 9, the source signals are well restored, but their orders are changed:  $y_1$  is the recovery signal of  $s_3$ ,  $y_2$  is the recovery signal of  $s_1$ , and  $s_2$  is the recovery signal of  $s_3$ .

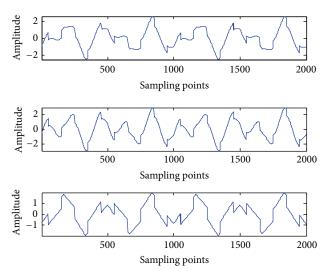


FIGURE 5: Mixed signals  $(x_1, x_2, x_3)$ .

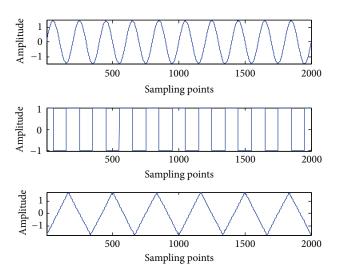


FIGURE 6: Recovery signals  $(y_1, y_2, y_3)$ .

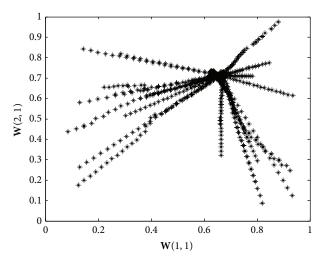


FIGURE 7: Emergence of the solution for MGSO.

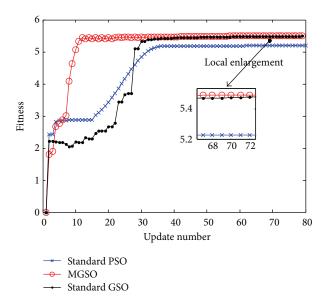
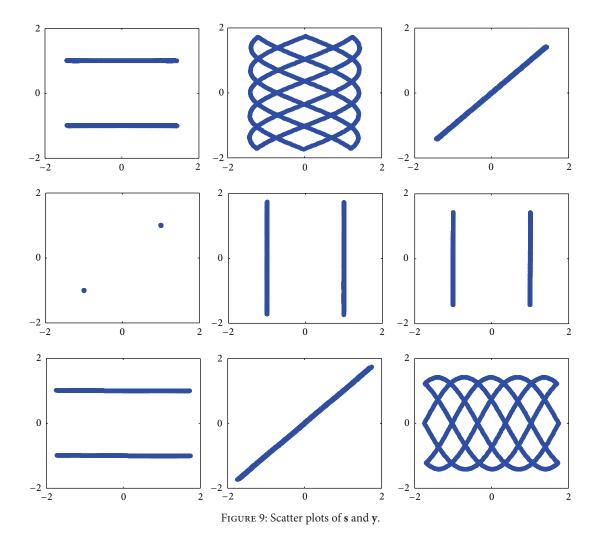


FIGURE 8: Fitness curves obtained by different optimization algorithms.



#### 6. Conclusions

To design an excellent optimization algorithm for BSS, a novel step adjustment strategy for the basic GSO is proposed in this paper. Based on independent component analysis (ICA), the modified GSO (MGSO) succeeds in separating the mixed signals using computer simulation. The experiment results show that the new MGSO-BSS has stronger global search ability and faster convergence rate than PSO-BSS and GSO-BSS, along with much higher accuracy. Therefore, it is concluded that the MGSO algorithm is more suitable for BSS.

### **Competing Interests**

The authors declare that they have no competing interests.

#### References

- [1] L. Jiang, L. Li, and G.-Q. Zhao, "Pulse-compression radar signal sorting using the blind source separation algrithms," in *Proceedings of the International Conference on Estimation, Detection and Information Fusion (ICEDIF '15)*, pp. 268–271, Harbin, China, January 2015.
- [2] H. Ghahramani, M. Barari, and M. H. Bastani, "Maritime radar target detection in presence of strong sea clutter based on blind source separation," *IETE Journal of Research*, vol. 60, no. 5, pp. 331–344, 2014.
- [3] J.-W. Huang, J.-C. Feng, and S.-X. Lü, "Blind source separation of chaotic signals in wireless sensor networks," *Acta Physica Sinica*, vol. 63, no. 5, Article ID 050502, 2014.
- [4] X.-W. Liu, W. Gao, N. Zhang, and W. Y. Liu, "ICA with banded mixing matrix based seismic blind deconvolution," *Progress in Geophysics*, vol. 4, article 21, 2007.
- [5] K.-H. Liu and W. H. Dragoset, "Blind-source separation of seismic signals based on information maximization," *Geophysics*, vol. 78, no. 4, pp. V119–V130, 2013.
- [6] V. Zarzoso, J. Millet-Roig, and A. K. Nandi, "Fetal ECG extraction from maternal skin electrodes using blind source separation and adaptive noise cancellation techniques," in *Proceedings of the Computers in Cardiology*, pp. 431–434, IEEE, Cambridge, Mass, USA, September 2000.
- [7] J. Metsomaa, J. Sarvas, and R. J. Ilmoniemi, "Multi-trial evoked EEG and independent component analysis," *Journal of Neuro-science Methods*, vol. 228, pp. 15–26, 2014.
- [8] K. Prakash and D. Hepzibha Rani, "Blind source separation for speech music and speech speech mixtures," *International Journal of Computer Applications*, vol. 110, no. 12, pp. 40–43, 2015.
- [9] T. Barnie and C. Oppenheimer, "Extracting high temperature event radiance from satellite images and correcting for saturation using independent component analysis," *Remote Sensing of Environment*, vol. 158, pp. 56–68, 2015.
- [10] V. H. Nguyen, C. Rutten, and J.-C. Golinval, "Fault diagnosis in industrial systems based on blind source separation techniques using one single vibration sensor," *Shock and Vibration*, vol. 19, no. 5, pp. 795–801, 2012.
- [11] D.-F. Luo, H.-J. Sun, and X.-L. Wen, "Research and application of blind signal separation algorithm to the aircraft engine vibration signal and fault diagnosis based on fast ICA," *Journal* of Convergence Information Technology, vol. 7, no. 10, pp. 248– 254, 2012.

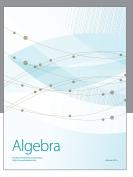
- [12] S. Mavaddaty and A. Ebrahimzadeh, "Blind signals separation with genetic algorithm and particle swarm optimization based on mutual information," *Radioelectronics and Communications Systems*, vol. 54, no. 6, pp. 315–324, 2011.
- [13] C. P. Dadula and E. P. Dadios, "A genetic algorithm for blind source separation based on independent component analysis," in *Proceedings of the IEEE International Conference on Humanoid, Nanotechnology, Information Technology, Commu*nication and Control, Environment and Management (HNICEM '14), pp. 1–6, November 2014.
- [14] A. N. Kumar and G. Jayakrishnan, "Application of ica for separation of artifacts in meg/eeg signals by blind source separation using improved particle swarm optimizer," *Procedia Engineering*, vol. 30, pp. 1020–1028, 2012.
- [15] Y.-X. Zhang, X.-M. Tian, and X.-G. Deng, "Blind source separation based on modified artificial bee colony algorithm," *Acta Electronica Sinica*, vol. 40, no. 10, pp. 2026–2030, 2012.
- [16] E. A. Grimaldi, F. Grimaccia, M. Mussetta, P. Pirinoli, and R. E. Zich, "A new hybrid genetical—swarm algorithm for electromagnetic optimization," in *Proceedings of the 3rd International Conference on Computational Electromagnetics and its Applications (ICCEA '04)*, pp. 157–160, November 2004.
- [17] E. Alfassio Grimaldi, F. Grimaccia, M. Mussetta, P. Pirinoli, and R. E. Zich, "Genetical swarm optimization: a new hybrid evolutionary algorithm for electromagnetic applications," in Proceedings of the 18th International Conference on Applied Electromagnetics and Communications (ICECom '05), pp. 1–4, Dubrovnik, Croatia, October 2005.
- [18] K. N. Krishnanand and D. Ghose, "Detection of multiple source locations using a glowworm metaphor with applications to collective robotics," in *Proceedings of the IEEE Swarm Intelligence* Symposium (SIS '05), pp. 87–94, IEEE, June 2005.
- [19] K. N. Krishnanand and D. Ghose, "Glowworm swarm based optimization algorithm for multimodal functions with collective robotics applications," *Multiagent and Grid Systems*, vol. 2, no. 3, pp. 209–222, 2006.
- [20] K. N. Krishnanand and D. Ghose, "Theoretical foundations for rendezvous of glowworm-inspired agent swarms at multiple locations," *Robotics and Autonomous Systems*, vol. 56, no. 7, pp. 549–569, 2008.
- [21] K. N. Krishnanand and D. Ghose, "Glowworm swarm optimization for multimodal search spaces," in *Handbook of Swarm Intelligence*, pp. 451–467, Springer, Berlin, Germany, 2011.
- [22] K. N. Krishnanand and D. Ghose, "Glowworm swarm optimization for simultaneous capture of multiple local optima of multimodal functions," *Swarm Intelligence*, vol. 3, no. 2, pp. 87–124, 2009.
- [23] K. N. Krishnanand and D. Ghose, "Glowworm swarm optimization for searching higher dimensional spaces," in *Innovations in Swarm Intelligence*, C. P. Lim, L. C. Jain, and S. Dehuri, Eds., vol. 248 of *Studies in Computational Intelligence*, pp. 61–75, Springer, Berlin, Germany, 2009.
- [24] M. Marinaki and Y. Marinakis, "A glowworm swarm optimization algorithm for the vehicle routing problem with stochastic demands," *Expert Systems with Applications*, vol. 46, pp. 145–163, 2016
- [25] B. Wu, C.-H. Qian, and W.-H. Ni, "Glowworm swarm optimization for cross dock scheduling problem," Computer Engineering and Applications, vol. 49, no. 6, 2013.
- [26] S. Mannar and S. N. Omkar, "Space suit puncture repair using a wireless sensor network of micro-robots optimized

- by Glowworm Swarm Optimization," *Journal of Micro-Nano Mechatronics*, vol. 6, no. 3-4, pp. 47–58, 2011.
- [27] A. Hyvärinen and E. Oja, "Independent component analysis: algorithms and applications," *Neural Networks*, vol. 13, no. 4-5, pp. 411–430, 2000.



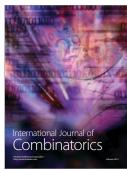














Submit your manuscripts at http://www.hindawi.com



