Multimodal Function Optimization Using an Improved Bat Algorithm in Noise-Free and Noisy Environments

Momin Jamil, Hans-Jürgen Zepernick and Xin-She Yang

Abstract Modern optimization problems in economics, medicine, and engineering are becoming more complicated and have a convoluted search space with multiple minima. These problems are multimodal with objective functions exhibiting multiple peaks, valleys, and hyperplanes of varying heights. Furthermore, they are nonlinear, non-smooth, non-quadratic, and can have multiple satisfactory solutions. In order to select a best solution among several possible solutions that can meet the problem objectives, it is desirable to find many such solutions. For these problems, the gradient information is either not available or not computable within reasonable time. Therefore, solving such problems is a challenging task. Recent years have seen a plethora of activities to solve such multimodal problems using non-traditional methods. These methods are nature inspired and are becoming popular due to their general applicability and effective search strategies. In this chapter, we assess the ability of an improved bat algorithm (IBA) to solve multimodal problems in noise-free and additive white Gaussian noise (AWGN) environments. Numerical results are presented to show that the IBA can successfully locate multiple solutions in both noise-free and AWGN environments with a relatively high degree of accuracy.

M. Jamil (✉)
Harman/Becker Automotive Systems GmbH,
Becker-Goering Str. 16, 76307 Karlsbad, Germany
e-mail: momin.jamil@harman.com

H.-J. Zepernick
Blekinge Institute of Technology, 371 79 Karlskrona, Sweden
e-mail: hans-jurgen.zepernick@bth.se

X.-S. Yang
Middlesex University, School of Science and Technology, London NW4 4BT, UK
e-mail: x.yang@mdx.ac.uk

S. Patnaik et al. (eds.), *Nature-Inspired Computing and Optimization*,
Modeling and Optimization in Science and Technologies 10,
DOI 10.1007/978-3-319-50920-4_2

1 Introduction

Most of the global optimization (GO) problems exhibit a highly rugged landscape. The challenge for any optimization algorithm is to find the highest or lowest point in such a landscape. In GO problems, the lowest or highest points refer to solution points in the function landscape. Over the years, nature-inspired population-based metaheuristic algorithms have been proven as an effective alternative to conventional optimization methods to solve modern engineering and scientific problems with complex search spaces and multiple optima. In case of multimodal problems with multiple optima, ideally, it is desirable to find all or as many as possible optima mainly for two reasons. First, to find all the optima can be dictated by a number of factors such as unavailability of some critical sources, or satisfaction of codal properties, etc. [37]. If such a situation arises, it is required to find or work on other available solutions. Second, the knowledge of the multiple solutions in a problem search space may provide valuable insight into the nature of the design space and, potentially, suggest alterative innovative solutions.

Population-based algorithms, such as differential evolution (DE), evolutionary strategies (ES), genetic algorithm (GA), and particle swarm optimization (PSO), have been extensively used to solve such problems. The intrinsic parallel nature of these algorithms enables them to simultaneously search multiple solution points in a search space. However, these algorithms tend to loose diversity and converge to a global best solution due to genetic drift [37]. Therefore, the two main challenges in these algorithms are (i) to maintain adequate population diversity so that multiple optimum solutions can be found, and (ii) how to preserve and maintain the discovered solution from one generation to another.

Iterative methods such as tabu search [9], sequential niche technique [2], and different niching methods [5, 10, 11, 20, 22, 25, 26, 28, 31, 40, 45] have been proposed to solve multimodal problems. Niches could aid differentiation of the species and maintain the diversity. These methods use various techniques that prevent algorithms to converge to the same solutions by prohibiting the algorithm from exploring those portions of the search space that have been already explored.

In recent years, Lévy flights also have been proposed within the context of metaheuristics algorithms to solve optimization problems [19, 29, 30, 35, 42]. It should be noted that by incorporating Lévy flights, the probability of visitation to new areas in a function landscape is increased. This, in turn, reduces the probability of search particles returning to the previously visited points in a search space. Thus, it can be argued that Lévy flights can be used as a search strategy in stochastic algorithms to solve complicated problems.

Animal foragers in nature hunt or search for food in dynamical environments without a priori knowledge about its location. Many studies show that search or flight behavior of animals and insects demonstrates the typical characteristics of Lévy flights [3, 4, 34]. These studies show that animals search for food in a random or quasi-random manner comprising active search phases and randomly alternating with phase of fast ballistic motion. The foraging path of an animal is random walk, in which the next move is based on the current location and the transition probability to the next location.

In this work, we investigate the effectiveness of replacing the dynamics of the bats within the originally proposed bat algorithm (BA) [44] by random sampling drawn from a Lévy distribution. This improved version of the BA also known as IBA is used to solve multimodal optimization problems. Its performance is experimentally verified on a well-chosen, comprehensive set of complex multimodal test problems varying both in difficulty and in dimension in both noise-free and additive white Gaussian noise (AWGN) environments. Without loss of generality, we will consider only minimization problems, with an objective, given a multimodal function, to locate as many global minima as possible.

The rest of the chapter is organized as follows. Section 2 presents an overview of the improved bat algorithm (IBA). In Sect. 3, experimental results on the performance of IBA on multimodal functions are presented. In Sect. 4, we compare the performance of IBA with other metaheuristic algorithms. In Sect. 5, results on the performance of IBA in a noisy environment are presented. Finally, Sect. 6 concludes the chapter.

2 Improved Bat Algorithm

In this section, an improved version of the BA [44] is formulated with the aim of making the method more practical for a wider range of problems, but without loosing the attractive features of the BA. The BA, mimicking the echolocation behavior of certain species of bats, is based on the following set of rules and assumptions [44]:

1. It is assumed that the bats know the difference between food/prey, background barriers, and use echolocation to sense the proximate distance from the prey;
2. The virtual bats are assumed to fly randomly with a frequency $f_{\min}$ with velocity $\mathbf{v}_i$ at position $\mathbf{x}_i$ by varying wavelength λ (or frequency f) and loudness A_0. Depending on the proximity from the target, the wavelength (or frequency) of emitted pulses and the rate of pulse emission $r \in [0, 1]$ can be adjusted by the bats;
3. It is further assumed that the loudness varies from a large (positive A_0) to a minimum value of $A_{\min}$;
4. Furthermore, ray tracing is not used in estimating the time delay and three-dimensional topography;
5. The frequency f is considered in the range $[0, f_{\max}]$ corresponding to the range of wavelengths $[0, \lambda_{\max}]$;

By making use of the above rules and assumptions, the standard bat algorithm (SBA) will always find the global optimum [44]. However, in SBA, bats rely purely on random walks drawn from a Gaussian distribution; therefore, speedy convergence may not be guaranteed [44].

In the improved version of the BA, the random motion of bats based on Gaussian distribution is replaced by Lévy flight. The motivation is that the power-law behavior of the Lévy distribution produces some members of the random population in the distant regions of the search space, while other members are concentrated around the mean of the distribution. The power-law behavior of the Lévy distribution also helps to induce exploration at any stage of the convergence, making sure that the system does not get trapped in local minima. The Lévy distribution also reduces the probability of returning to the previously visited sights, while the number of visitations to new sights is increased.

Secondly, we incorporate an exponential model of frequency tuning and pulse emission rate compared to a fixed setting of these parameters in BA. This incorporation into this model has two advantages: (i) frequency tuning and dynamic control of exploitation; and (ii) exploration by automatic switching to intensive exploitation. The frequency-based tuning and pulse emission rate change to imitate the behavior of a bat lead to good convergence. Pulse emission rate and loudness control mechanism in the BA help in maintaining a fine balance between exploration and exploitation. In fact, an automatic switch to more extensive exploitation instead of exploration can be achieved when bats are in the vicinity of the optimal solution. The basic steps of this improved version of the BA known as IBA are summarized in the pseudo-code shown in Procedure 1 [16].

The position and velocity of bat i are represented by the vectors $\mathbf{x}_i$ and $\mathbf{v}_i$, respectively. The pulse frequency for bat i at position $\mathbf{x}_i$ is denoted by f_i. The initial pulse emission rate, updated pulse emission rate after each iteration, initial loudness, and updated loudness after each iteration for bat i are represented by symbols $r_i^{t_0}$, r_i, $A_i^{t_0}$, and A_i in (1) and (2), respectively. The symbols γ and α in (1) and (2) are constants. The symbols $\beta \in [0, 1]$, f_{min}, and f_{max} in (3) denote a random number drawn from a uniform distribution, minimum, and maximum frequency of the emitted pulse, respectively. In (4), the symbol $\mathbf{x}_i^{\text{best}}$ represents the current best solution found by bat i by comparing all the solutions among all the *NB* bats and 'rand' is a random number drawn from a uniform distribution. The parameter δ in (6) scales the random step size S and is related to the problem dimension [43]. For most of the cases, $\delta = 1$ can be used; however, it can be set to smaller values when the dimension of the problem is small [43]. A smaller value of δ might hold for simple unimodal problems. However, for multimodal problems with multiple global minima scattered in the search space, a smaller step size might hinder the search process. The operator $\odot$ denotes entrywise multiplication.

Procedure 1: Pseudo-code of IBA based on Lévy Flights [16]

1: Initialize the positions $\mathbf{x}_i$, $i = 1, 2, ..., NB$, and velocity $\mathbf{v}_i$ of the *NB* bats in *D*-dimensional problem space.

2: Define the pulse frequency f_i at position $\mathbf{x}_i$.

3: Initialize the pulse emission rate r^{t_0} and the loudness A^{t_0}.

4: **for** all $\mathbf{x}_i$

5: Calculate objective function of all *NB* bats $F_i = f(\mathbf{x}_i)$

end for
Rank the position of the bats according to their fitness values.
Find the current best bat position $\mathbf{x}_i^{\text{best}}$.
$t = 1$
while (t $<$ Maximum number of iterations)
    $t = t + 1$
    Update r_i and A_i as

$$r_i^{t+1} = r_i^{t_0}[1 - \exp(-\gamma t)] \quad (1)$$

$$A_i^{t+1} = \alpha A_i^{t_0} \quad (2)$$

    for (loop over all *NB* bats and all *D* dimensions)
        Adjust frequency as

$$f_i = f_{min} + (f_{max} - f_{min})\beta \quad (3)$$

        Update velocities as

$$\mathbf{v}_i^t = \mathbf{v}_i^{t-1} + (\mathbf{x}_i^{t-1} - \mathbf{x}_i^{\text{best}}) f_i \quad (4)$$

        Update locations or new solutions as

$$\mathbf{x}_i^t = \mathbf{x}_i^{t-1} + \mathbf{v}_i^t \quad (5)$$

        if (rand $> r_i$)
            Generate new solution using Lévy Flight as

$$\mathbf{x}_i^t = \mathbf{x}_i^{\text{best}} + \delta \odot \mathbf{L}_\lambda(S) \quad (6)$$

        end if
        Check if the new solution (5) or (6) is within the problem domain.
        if (True)
            Move to next step.
        else
            Apply problem bound constraints.
        end
        Evaluate the objective function at new location (solution) generated as

$$F_k = f(\mathbf{x}_i) \quad (7)$$

        if ((rand $< A_i$) or ($F_k < F_i$))
            Accept the new solution.
        end if
        Rank the bats and find the current best $\mathbf{x}_i^{\text{best}}$.

31: **end for**
32: **end while**

In (6), the random step size S is drawn from a Lévy distribution which has infinite mean and variance:

$$L_{\lambda}(S) \sim \frac{1}{S^{\lambda+1}}|S| >> 0 \tag{8}$$

The probability of obtaining a Lévy random number in the tail of the Lévy distribution is given by (8). The choice of λ in (8) can have a significant impact on the search ability of the optimization algorithm based on Lévy flights. Therefore, some experimentation is required to find a proper value of λ [35, 43]. In order to determine a proper value of λ, five different benchmark functions were selected. These problems include unimodal (Beale), multimodal with a few local minima (Carrom table and Holder table), and multimodal functions with many local minima (Griewank, Eggholder) functions. For each of these problems, 50 independent trails were conducted for a fixed number of iterations with $\lambda = 1.3$, 1.4, 1.5, and 1.6. In order to determine an effective value of λ, the best values produced by the IBA for each landscape were recorded and averaged over the number of trails. The only value of λ that performed well on all landscapes was found to be 1.5. This value was subsequently chosen for all the tests presented later in this chapter. The value of $\lambda = 1.5$ has also been used in other metaheuristic algorithms based on Lévy flights [35, 43].

3 IBA for Multimodal Problems

In this section, the results obtained by applying IBA on a set of multimodal functions with multiple solutions are presented. The effectiveness of the IBA to handle multimodal problems is validated on a set of benchmark functions with different characteristics.

3.1 Parameter Settings

The universal set of initial values of parameters such as NB, $r_i^{t_0}$, r_i, $A_i^{t_0}$, and A_i that are applicable to all the benchmark functions used in this study do not exist. Therefore, finding effective values of these parameters requires some experimentation. The parameter settings listed in Table 1 are obtained from the trial experiments on a limited set of benchmark functions with two and higher dimensions. We have used different values of $NB = 50$, 100, 150, 200 and found that $NB = 100$ seems to be sufficient for most multimodal optimization problems considered in this work. For

Table 1 Parameter settings for IBA

Parameter	Value
Number of bats (population size) NB	100
Number of generations G	5000
Number of runs R	100
Initial loudness A^{t_0}	0.1
Initial pulse emission rate r^{t_0}	0.1
Constants $\alpha = \gamma$	0.9
Lévy step length λ	1.5
Minimum frequency f_{min}	0
Maximum frequency f_{max}	Dependent on problem domain size

tougher problems, larger NB can be used only if there is no better alternative, as it is more computationally expensive [43]. In summary, the parameter settings listed in Table 1 were used for all the experiments, unless we mention new settings for one or other parameters.

3.2 Test Functions

In various applications, the objective function exhibit multiple global minima. In order to experimentally evaluate the strengths and weaknesses of IBA, we have used a set of well-chosen test functions [12, 15, 21, 23], representing different characteristics and various levels of difficulty. The test suite includes functions having evenly and unevenly spaced multiple global optima, multiple global optima in the presence of multiple local minima, and deceptiveness. The test functions are listed in Table 2 with their respective domain sizes, number of minima (multimodal functions), and value of global minimum.

3.3 Numerical Results

Over the years, several different performance measures to evaluate the performance of stochastic algorithms have been reported in the literature [33]. However, there are no standard performance measures and in practice the results are reported as averages from certain number of independent runs for stochastic algorithms. Therefore, we will adapt commonly used performance measures and report results in terms of mean and standard deviation (SD) attained with a population size and number of generations.

Table 2 List of test functions

Function	Domain size	No. of minima	Global minima
f_1	[−10, 10]	1	−176.1375
f_2	[0, pi]	1	−1.0813
f_3	[0, pi]	1	0.9
f_4	[0, 10]	2	−0.6737
f_5	[−5, 5]	2	−4.5901
f_6	[−5, 5]	2	−0.8510
f_7	[−5, 5]	2	−1.0316
f_8	[−10, 10]	2	−10.8723
f_9	[−5, 5]	2	0
f_{10}	[−5, 0; 10, 15]	3	0.3979
f_{11}	[−10, 10]	4	−24.1568
f_{12}	[−10, 10]	4	−2.0626
f_{13}	[−5, 5]	4	0
f_{14}	[−10, 10]	4	−19.2085
f_{15}	[−10, 10]	4	−26.9203
f_{16}	[−10, 10]	4	−0.9635
f_{17}	[−1, 1]	6	−1
f_{18}	[−10, 10]	9	−176.541
f_{19}	[−10, 10]	9	−24.0624
f_{20}	[−20, 20]	12	−1
f_{21}	[−10, 10]	18	−186.7309
f_{22}	[−7, 7]	20	0
f_{23}	[0, 1]	25	−1
f_{24}	[0, 1]	25	−1

f_1: Levy 5, f_2: Michaelwicz, f_3: Periodic, f_4: Keane, f_5: Modified Ackley, f_6: Multimodal Yang, f_7: Six Hump Camel, f_8: Test Tube Holder, f_9: Trecanni, f_{10}: Branin, f_{11}: Carrom Table f_{12}: Cross-in-Tray, f_{13}: Himmelblau, f_{14}: Holder Table 1, f_{15}: Holder Table 2, f_{16}: Pen Holder f_{17}: Root 6, f_{18}: Hansen, f_{19}: Henrik-Madsen, f_{20}: Butterfly, f_{21}: Shubert f_{22}: Parsopoulos, f_{23}: Deb 1, f_{24}: Deb 2

The best fitness value produced by the IBA after each run was recorded. The mean fitness value and SD are presented in Table 3. From the results presented in Table 3, it can be seen that IBA performs equally well on almost all of the functions tested. The only two exceptions are f_{17} and f_{21} which show small deviation of $10E-4$ and $10E-5$ from the known global minima.

The minimum, maximum, and mean number of function evaluations (NFE) required by the IBA to converge to a solution averaged over the number of independent runs are also reported. From the results presented in Table 3, it can be seen that IBA can locate global minima with relatively few NFE (lower mean NFE values) with the exception of f_{20}. This function has 760 local minima and 18 global minima.

Table 3 Statistical results of 100 runs obtained by IBA for 2D functions (global minima ≥ 2). NFE_{min}: Minimum number of function evaluations; NFE_{max}: Maximum number of function evaluations; NFE_{mean}: Mean of number of function evaluations; Mean: Mean of fitness value; SD: Standard deviation of the fitness values

Function	NFE_{min}	NFE_{max}	NFE_{mean}	Mean	SD
f_4	800	2900	1817	−0.6737	$3.3474E-16$
f_5	3300	6700	5417	−4.5901	$8.9265E-16$
f_6	3300	11800	7294	−0.8512	$1.2274E-15$
f_7	1400	5500	3851	−1.0316	$1.3390E-15$
f_8	3700	12100	6572	−10.8723	$1.7853E-15$
f_9	2100	5500	3952	0	0
f_{10}	2800	8200	4899	0.3979	$5.5791E-17$
f_{11}	2500	5400	3774	−24.1568	$1.7853E-14$
f_{12}	3300	6400	4766	−2.0626	$8.9265E-16$
f_{13}	2600	6500	4343	0	0
f_{14}	2600	7500	4282	−19.2085	$2.8656E-14$
f_{15}	2400	5700	3910	−26.9203	$4.9989E-14$
f_{16}	2700	7700	5166	−0.9635	$2.23116E-15$
f_{17}	2200	77500	7244	−0.9995	$1.4824E-04$
f_{18}	4100	10100	6559	−176.5417	$5.7130E-14$
f_{19}	2600	10700	6343	−24.0624	$1.0712E-14$
f_{20}	7650	238800	65250	−1	0
f_{21}	4300	500000	125481	−186.7309	$2.8468E-05$
f_{22}	2900	8100	5322	0	0
f_{23}	1800	8400	3614	−1	$1.3530E-06$
f_{24}	1800	1800	8400	−1	0

The 18 global minima are classified into 9 different clusters. Each cluster consists of two solutions that are separated by small distance 0.98, while the distance between two clusters is 5.63 [13]. This function is difficult to optimize for any algorithm [13, 14] and IBA is no exception. Box plots are shown in Figs. 1 and 2 depicting the NFE required by the IBA to converge to a solution. The median and mean values of the NFE are depicted by the notch and '$\circ$,' whereas the whiskers above and below the box give additional information about the spread of the NFE. Whiskers are vertical lines that end in a horizontal stroke. The '+' markers in these plots represent the outliers that are beyond the lower and upper whiskers, i.e., NFE_{min} and NFE_{max} in Table 3.

Figures 3, 4, 5, 6, and 7 show the contour plots of functions f_{20}, f_{21}, f_{22}, f_{23}, and f_{24}, respectively. The number of global minima for f_{20} and f_{22} depends on the problem domain size. For a domain size of $[-20, 20]^D$ and $[-7, 7]^D$, where D represents the problem dimension, functions f_{20} and f_{22} have 12 and 20 global

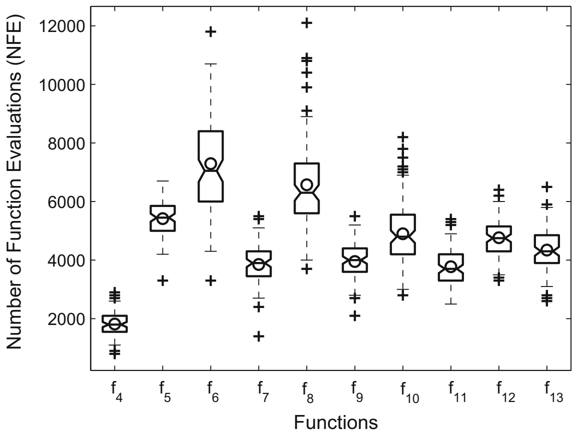

Fig. 1 Box plot depicting the NFE for function f_4 to f_{13}

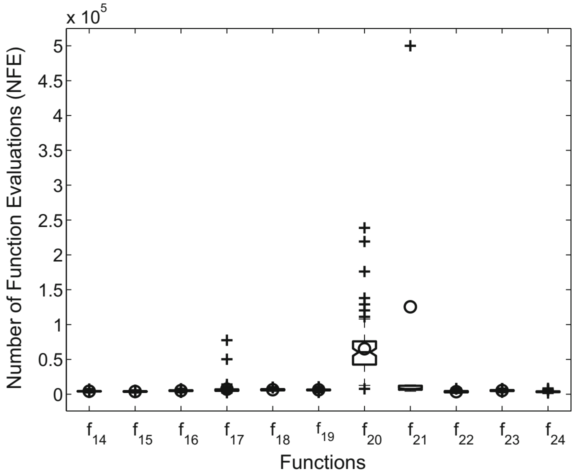

Fig. 2 Box plot depicting the NFE for function f_{14} to f_{24}

minima, respectively. The functions f_{23} and f_{24} have 25 global minima that are evenly and unevenly spread in the function landscape [3]. The '+' markers in these figures represent the global minima (solutions) located by the IBA in 100 independent runs with the parameter settings in Table 1. From the results presented in these figures, we can see that IBA is able to locate global minima with relatively high degree of accuracy. Due to the complex landscape of f_{20}, the number of bats $NB = 100$ is found to be not sufficient to locate all the global minima. We have found that with $NB = 125$, IBA was able to locate all the global minima for function f_{20}.

According to [36], for practical reasons, it is desirable to find all global minima. The ability and effectiveness of the IBA to accurately locate all of the multiple minima, a level of accuracy, $\epsilon \in (0, 1]$ is defined. If the Euclidean distance of a computed solution to a known global optimum is less than ϵ, then the solution is

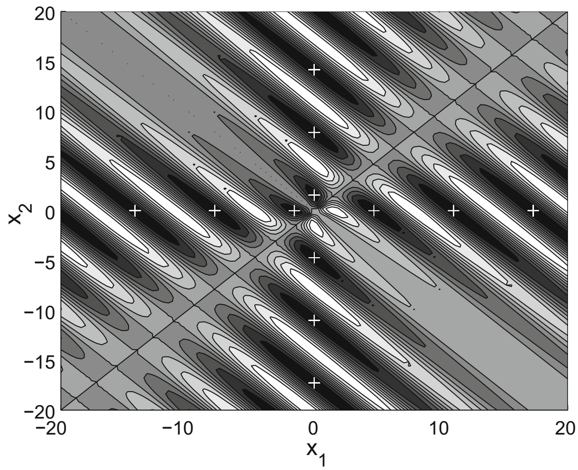

Fig. 3 Contour plot of function f_{20} (12 global minima indicated by '+' found by IBA after 100 independent runs)

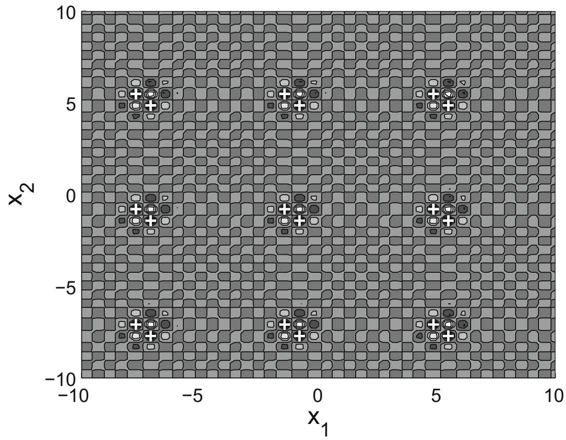

Fig. 4 Contour plot of function f_{21} (18 global minima indicated by '+' found by IBA after 100 independent runs)

considered as a global optimum. The ability of IBA to consistently locate all the solutions for each function for a given set of parameters is also measured. The number of runs converged on solutions in % in Tables 4 and 5 signifies the number of the independent runs converged on the solutions. From the results presented in these tables, it can be seen that IBA could find all the solutions for a specified value of ϵ. These results also demonstrate the performance consistency of the IBA which measures the ability of the algorithm to consistently locate all the solutions for each function for a given set of parameters.

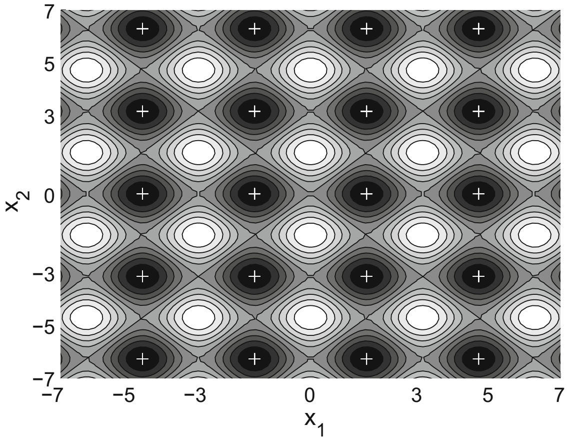

Fig. 5 Contour plot of function f_{22} (25 global minima indicated by '+' found by IBA after 100 independent runs)

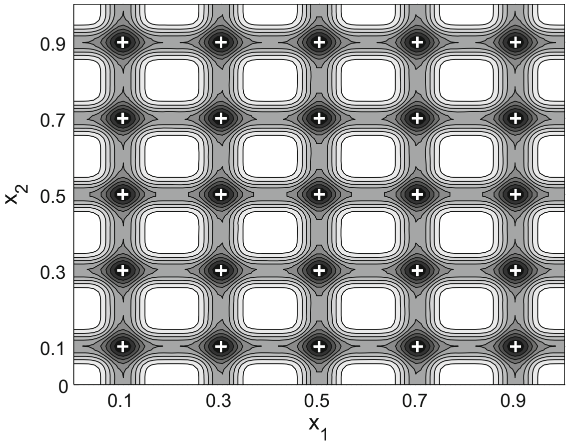

Fig. 6 Contour plot of function f_{23} (25 global minima indicated by '+' found by IBA after 100 independent runs). The minima are located at $\{x_1, x_2\} =$ 0.1, 0.3, 0.5, 0.7, 0.9

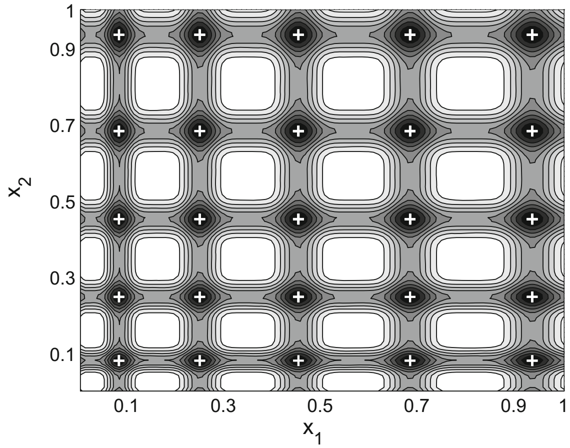

Fig. 7 Contour plot of function f_{24} (25 global minima indicated by '+' found by IBA after 100 independent runs). The minima are located at $\{x_1, x_2\} = 0.08, 0.246,$ $0.45, 0.681, 0.934$

Table 4 IBA performance on test functions for $\epsilon = 10^{-5}$

Function	Solutions	% Number of runs converged on solutions								
		1	2	3	4	5	6	7	8	9
f_4	2	45	55	–	–	–	–	–	–	–
f_5	2	49	51	–	–	–	–	–	–	–
f_6	2	43	57	–	–	–	–	–	–	–
f_7	2	50	50	–	–	–	–	–	–	–
f_8	2	47	53	–	–	–	–	–	–	–
f_9	2	57	43	–	–	–	–	–	–	–
f_{10}	3	33	30	37	–	–	–	–	–	–
f_{11}	4	28	23	28	21	–	–	–	–	–
f_{12}	4	25	21	26	28	–	–	–	–	–
f_{13}	4	40	11	21	28	–	–	–	–	–
f_{14}	4	23	21	29	27	–	–	–	–	–
f_{15}	4	23	31	21	25	–	–	–	–	–
f_{16}	4	29	26	17	28	–	–	–	–	–
f_{17}	6	24	19	21	19	11	6	–	–	–
f_{18}	9	5	11	8	12	13	16	10	14	11
f_{19}	9	5	16	11	13	19	15	6	8	7

4 Performance Comparison of IBA with Other Algorithms

In this section, we compare IBA with other metaheuristic algorithms, such particle swarm optimization (PSO) [18], differential evolution (DE) [32], and evolutionary strategy (ES) [38]. We abstain to introduce these algorithms and did not make a special effort to fine-tune different parameters of these algorithms. For DE, we have used the strategy DE/best/2/bin [32] with the weighting factor $F = 0.8$ and crossover constant $CR = 0.7$. We have used the standard version of PSO with global learning, i.e., no local neighborhoods, an inertial constant $\omega = 0.3$, a cognitive constant $c_1 = 1$, and a social constant for swarm interaction $c_2 = 1$. For ES, we have used an offspring of $\lambda = 10$ for each generation and standard deviation $\sigma = 1$ for changing solutions. In order to have a fair comparison, we have used the settings of Table 1 for the population size NB, fixed NFE, and R for these algorithms.

In Table 6, the number of minima found by each algorithm in 100 independent runs is shown in parentheses. The numerator in the parentheses represents the total number of minima found by each algorithm, while the denominator represents the actual number of minima. The boldface terms represent the best results. For example, in the case of the function f_{18}, IBA has found 9 out of 9 minima, i.e., (**9**/**9**), whereas the other algorithms found either fewer or none of the minima. These results show that IBA has an ability to find multiple solutions for all the functions tested.

Table 5 IBA performance on test functions for $\epsilon = 10^{-5}$

Solutions	% Number of runs converged on solutions				
	f_{20}	f_{21}	f_{22}	f_{23}	f_{24}
1	8	7	4	2	4
2	5	5	3	4	3
3	2	6	2	6	2
4	3	9	5	3	5
5	10	5	2	3	2
6	22	6	2	3	2
7	19	5	2	5	2
8	11	3	3	8	3
9	2	6	2	5	2
10	1	3	4	4	4
11	4	8	1	3	1
12	13	4	5	3	5
13	–	3	9	2	9
14	–	2	7	5	7
15	–	6	3	3	3
16	–	4	2	6	2
17	–	6	1	7	1
18	–	4	6	3	6
19	–	2	6	2	6
20	–	6	8	4	8
21	–	–	3	6	3
22	–	–	3	4	3
23	–	–	8	4	8
24	–	–	4	3	4
25	–	–	5	2	5

5 IBA Performance in AWGN

Optimization in a noisy environment occurs in various applications such as experimental optimization. The problem of locating either minima or maxima of a function is vital in applications such as design of wireless systems [8], spectral analysis and radio-astronomy [27], and semiconductor modeling and manufacturing [39]. Traditionally, the simplex method by Nelder and Mead [7, 24] has been used for optimization in noisy environments. However, in order to overcome the drawbacks and the deficiencies of simplex methods [17], different variants of the simplex method have been proposed [41]. More sophisticated methods and extensive studies in this direc-

Table 6 Comparison of IBA with DE, ES, and PSO. Mean: mean of best value; SD: standard deviation of the best value; SEM: standard error of mean

Function		IBA	DE	ES	PSO
f_4	Mean	-0.67	-0.67	-0.67	-0.6737
	SD	0	0	$2.13E-07$	$4.44E-09$
	Minima	**(2/2)**	**(2/2)**	(1/2)	(1/2)
f_8	Mean	-10.87	-10.87	-10.87	-10.87
	SD	0	$1.07E-14$	$2.50E-03$	$7.23E-04$
	Minima	**(2/2)**	**(2/2)**	(1/2)	(1/2)
f_{14}	Mean	-19.20	-19.20	-19.20	-19.20
	SD	0	0	$2.54E-05$	$9.50E-03$
	Minima	**(4/4)**	**(4/4)**	(2/4)	(2/4)
f_{16}	Mean	-0.96	-0.96	-0.96	-0.96
	SD	0	0	$5.41E-05$	$3.42E-06$
	Minima	**(4/4)**	**(4/4)**	(3/4)	(3/4)
f_{17}	Mean	-0.99	-1	-0.99	-0.99
	SD	$2.80E-07$	0	$9.57E-05$	$1.19E-04$
	Minima	**(6/6)**	**(6/6)**	(1/6)	(1/6)
f_{18}	Mean	-176.54	-176.54	-162.41	-176.39
	SD	0	0	11.33	$1.40E-01$
	Minima	**(9/9)**	**(9/9)**	(0/9)	(3/9)
f_{19}	Mean	-24.07	-24.06	-22.51	-24.05
	SD	0	0	1.18	$8.20E-03$
	Minima	**(9/9)**	**(9/9)**	(0/9)	(6/9)
f_{22}	Mean	0	0	$1.3723E-05$	$1.66E-05$
	SD	0	0	$1.41E-05$	$1.85E-05$
	Minima	**(12/12)**	(11/12)	(6/12)	(5/12)

tion are discussed in [1]. Different population-based algorithms, e.g., PSO, have also been used to optimize functions in AWGN and multiplicative noise environments [27].

Information about the function $f(x)$ is obtained in the form of $\widehat{f}(x)$, where $\widehat{f}(x)$ is an approximation of the true function value $f(x)$, corrupted by AWGN. The influence of AWGN on the values of the objective functions was simulated according to [6] and is given as

$$\widehat{f}(x) = f(x) + \eta, \quad \eta \sim N\left(0, \sigma^2\right) \tag{9}$$

where $\eta \sim N(0, \sigma^2)$ is a Gaussian distributed random variable with zero mean and variance σ^2.

5.1 Numerical Results

In this section, we study the ability of IBA to handle the two-dimensional multimodal function optimization in AWGN. However, these problems can also be extended to dimensions greater than 2. Due to space limitations, we selected 9 functions from those listed in Table 1. These include two functions with single global minimum and 7 functions with two or more than two global minima. Experiments were carried out for four different levels of $\sigma^2 = 0.01, 0.05, 0.07$, and 0.09. At each function evaluation,

Table 7 Statistical results of 100 runs obtained by IBA on selected functions in AWGN. Mean: mean of fitness value; SD: standard deviation of the fitness values; MAPE: mean absolute percentage error

Function		$\sigma^2 = 0.01$	$\sigma^2 = 0.05$	$\sigma^2 = 0.07$	$\sigma^2 = 0.09$
$f_1(-176.1375)$	Mean	−176.1830	−176.3646	−176.5443	−176.5442
	SD	0.0025	0.0121	0.0171	0.228
	MAPE	0.0258	0.1289	0.1799	0.2309
$f_2(-1.8013)$	Mean	−1.8468	−2.0283	−2.1206	−2.2095
	SD	0.0025	0.0122	0.0193	0.0242
	MAPE	2.5281	12.6022	17.7255	22.6617
$f_3(0.9)$	Mean	0.8556	0.6838	0.6053	0.5330
	SD	0.0027	0.0254	0.0386	0.0529
	MAPE	4.9308	24.0261	32.7394	46.5648
$f_5(-4.59012)$	Mean	−4.6349	−4.8137	−4.9029	−4.9957
	SD	0.0029	0.0118	0.0169	0.0258
	MAPE	0.9753	4.8701	6.8144	8.8361
$f_6(-0.851)$	Mean	−0.8950	−1.0741	−1.1595	−1.2473
	SD	0.0025	0.0153	0.0184	0.0234
	MAPE	5.1666	26.2138	36.2536	46.5684
$f_{11}(-24.1568)$	Mean	−24.2016	−24.3806	−24.4665	−24.5567
	SD	0.0026	0.0140	0.0156	0.0215
	MAPE	0.1856	0.9266	1.2822	1.6556
$f_{14}(-19.20850)$	Mean	−19.2530	−19.4329	−19.5236	−19.6095
	SD	0.0027	0.0137	0.0183	0.0229
	MAPE	0.2319	1.1683	1.6403	2.0875
$f_{17}(-1)$	Mean	−1.0424	−1.2112	−1.2946	−1.3851
	SD	0.0027	0.0137	0.0211	0.0310
	MAPE	4.2370	21.1208	29.4648	38.5136
$f_{18}(-176.5418)$	Mean	−176.5849	−176.7617	−176.8452	−176.9367
	SD	0.0025	0.0134	0.0160	0.0216
	MAPE	0.0244	0.1245	0.1719	0.2237

noise was added to the actual function according to (9) for different values of σ^2. For each variance value, 100 independent runs of IBA were performed.

Based on the mean absolute percentage error (MAPE) results presented in Table 7, it can be seen that increasing the value of σ^2 deteriorates the ability of IBA to locate global minimum/minima for unimodal (f_2 and f_3) and multimodal (f_6 and f_{17}) functions. For these functions, a performance degradation of more than 20 and 40% can be observed for $\sigma^2 = 0.09$ that may adversely effect the ability of IBA to accurately locate global optima for these functions. The term enclosed in the brackets in front of the function names represents the known value of the global minimum (unimodal) and minima (multimodal) for these functions.

Table 8 IBA performance on selected test function in AWGN with different σ^2

Function	Mean Euclidean distance of computed solution from known solution				
	σ^2	1	2	3	4
f_1	0.01	$8.2301E{-}04$	–	–	–
	0.05	0.0019	–	–	–
	0.07	0.0022	–	–	–
	0.09	0.0023	–	–	–
f_2	0.01	0.0540	–	–	–
	0.05	0.0132	–	–	–
	0.07	0.0160	–	–	–
	0.09	0.0201	–	–	–
f_3	0.01	0.0282	–	–	–
	0.05	0.0830	–	–	–
	0.07	1.1067	–	–	–
	0.09	2.2026	–	–	–
f_5	0.01	0.0119	0.0117	–	–
	0.05	0.0332	0.0294	–	–
	0.07	0.0288	0.0311	–	–
	0.09	0.0319	0.0379	–	–
f_6	0.01	0.0607	0.0708	–	–
	0.05	0.1403	0.1759	–	–
	0.07	0.1633	0.1727	–	–
	0.09	0.2278	0.2046	–	–
f_{11}	0.01	0.0048	0.0057	0.0060	0.0044
	0.05	0.0151	0.0162	0.0112	0.0133
	0.07	0.0179	0.0160	0.0149	0.0151
	0.09	0.0206	0.0158	0.0153	0.0178
f_{14}	0.01	0.0107	0.0094	0.0088	0.0081
	0.05	0.0215	0.0207	0.0199	0.0241
	0.07	0.0238	0.0262	0.0242	0.0210
	0.09	0.0291	0.0289	0.0243	0.0238

Next, in Tables 8, 9, and 10, we compare the mean Euclidean distance (the difference between the obtained and actual global minimum) of the computed solution from known a priori solution for four levels of σ^2. From the results presented in Tables 8, 9, and 10, we can see that as the value of σ^2 increases, the ability of the IBA to locate global minima decreases significantly for f_3 (periodic) and f_6 (Yang

Table 9 IBA performance for function f_{18} in AWGN with different σ^2

Solutions	$\sigma^2 = 0.01$	$\sigma^2 = 0.05$	$\sigma^2 = 0.07$	$\sigma^2 = 0.09$
1	$9.9842E-04$	0.0026	0.0017	0.1838
2	$6.9885E-04$	0.0018	0.0021	0.0025
3	$8.4551E-04$	0.0020	0.0020	0.0026
4	$8.6135E-04$	0.0017	0.0024	0.0023
5	$9.8587E-04$	0.0020	0.0023	0.0024
6	$7.9620E-04$	0.0014	0.0020	0.0029
7	$8.1328E-04$	0.0016	0.0021	0.0029
8	$9.2927E-04$	0.0021	0.0021	0.0020
9	0.0012	0.0012	0.0027	0.0020

Table 10 IBA performance for function f_{22} in AWGN with different σ^2

Solutions	$\sigma^2 = 0.01$	$\sigma^2 = 0.05$	$\sigma^2 = 0.07$	$\sigma^2 = 0.09$
1	0.0291	0.1353	0.0813	0.1324
2	0.0169	0.0692	0.0699	0.2266
3	0.0264	0.0488	0.1290	0.1272
4	0.0297	0.0623	0.1169	0.1436
5	0.0462	0.0804	0.0701	0.1254
6	0.0074	0.1318	0.1765	0.0758
7	0.0196	0.0686	0.0930	0.0554
8	0.0439	0.0775	0.0826	0.1054
9	0.0237	0.0655	0.0676	0.0512
10	0.0243	0.0870	0.1073	0.0927
11	0.0284	0.0927	0.0562	0.1019
12	0.0173	0.0856	0.1072	0.0889
13	0.0241	0.1015	0.1081	0.1140
14	0.0335	0.0971	0.0781	0.1124
15	0.0303	0.0782	0.1083	0.0626
16	0.0443	0.0647	0.0658	0.1866
17	0.0313	0.0291	0.0540	0.1225
18	0.0349	0.0649	0.0896	0.0873
19	0.0421	0.0986	0.0966	0.1733
20	0.0365	0.0825	0.1145	0.0874

multimodal) functions. For these two functions, the largest difference between the obtained and actual global minimum can be observed.

Interestingly, in the case of functions with more than one global minimum (with the exception of f_6 and f_{17}) such as f_5, f_{11}, f_{14}, f_{18}, and f_{23} with few or many local minima, IBA seems to perform relatively well in AWGN even for the highest value of σ^2. It seems that the noise plays a positive role for these multimodal functions by helping the IBA to escape local minima of the objective function. For these multimodal functions, the performance degradation is significantly low for the highest value of σ^2. On the contrary, in the case of unimodal problems such as f_2 and f_3, significant performance deterioration can be observed for the highest value of σ^2. One can hypothesize that for the unimodal functions high values of σ^2 can create deep troughs (for minimization problems) or crests (for maximization problems) in the function landscape that are mistaken by the IBA or any other optimization algorithm as a global minimum or maximum.

6 Conclusions

In this work, IBA has been presented as a viable alternative to existing numerical optimization methods for unimodal and multimodal problems. In this chapter, the ability of IBA to solve unimodal and multimodal problems in non-noise and AWGN has been investigated. Performance results have been reported for a set of test functions with varying levels of difficulty, different number of minima, and different level of noise variances. The experimental results indicate that IBA is very stable and efficient in the presence of noise. It is a very noise-tolerant method and can be used for minimization or maximization of noisy functions. It has performed exceptionally well even in the presence of noise with high variance. Conclusively, IBA appears to be a viable alternative technique for solving global optimization problems and may offer an alternative where other techniques fail. However, further research may be required to fully comprehend the dynamics and the potential limits of the IBA.

References

1. Arnold DV (2001) Local performance of evolution strategies in the presence of noise. PhD thesis, University of Dortmund, Germany
2. Beasley D, Bull DR, Martin RR (1993) A sequential Niche technique for multimodal function optimization. Evol Comput 1(2):101–125
3. Brits R, Engelbrecht AP, van den Bergh F (2007) Locating multiple optima using particle swarm optimization. Appl Math Comput 189(12):1859–1883
4. Brown C, Liebovitch LS, Glendon R (2007) Lévy flights in Dobe Ju/Hoansi foraging patterns. Hum Ecol 35(1):129–138
5. De Jong KA (1975) An analysis of the behavior of a class of genetic adaptive system. PhD thesis, University of Michigan, USA

6. Elster C, Neumaier A (1997) A method of trust region type for minimizing noisy functions. Computing 58(1):31–46
7. Fletcher R (1987) Practical methods of optimization, 1st edn. Wiley, Chichester
8. Fortune SJ, Gay DM, Kernighan BW, Landron O, Valenzulea RA, Wright MH (1995) WISE design of indoor wireless systems. IEEE Comput Sci Eng 2(1):58–68
9. Glover F (1989) Tabu search—part I. ORSA J Comput 1(3):190–206
10. Goldberg DE, Richardson J (1987) Genetic algorithms with sharing for multimodal function optimization. In: Proceedings of international conference on genetic algorithms and their application, USA, pp 41–49
11. Harik GR (1995) Finding multimodal solutions using using restricted tournament selection. In: Proceedings of international conference on genetic algorithms and their application, USA, pp 24–31
12. http://www-optima.amp.i.kyoto-u.ac.jp/member/student/hedar/Hedar_files/TestGO_files/Page364.htm
13. Iwamatsu M (2006) Multi-species particle swarm optimizer for multimodal function optimization. IEICE Trans Inf Syst (English Edition) E89-D(3):1181–1187
14. Iwamatsu M (2006) locating all the global minima using multi-species particle swarm optimizer: the inertia weight and the constriction factor variants. In: Proceedings of IEEE congress on evolutionary computation, Canada, pp 816–822
15. Jamil M, Yang XS (2013) A literature survey of benchmark functions for global optimisation problems. Int J Math Model Numer Optim 4(2):150–194
16. Jamil M, Zepernick HJ, Yang XS (2015) Synthesizing cross-ambiguity function using improved bat algorithm (Ed Xin-She Yang). Springer, Heidelberg, pp 179–202
17. Kelley CT (1995) Iterative methods for optimization. SIAM
18. Kennedy J, Eberhart RC (1995) Particle swarm optimization. In: Proceedings of IEEE international conference on neural networks, Australia, pp 1942–1948
19. Lee C-Y, Yao X (2004) Evolutionary programming using mutations based on Lévy probability distribution. IEEE Trans Evol Comput 8(1):1–13
20. Li J-P, Balaz ME, Parks GT (2002) A species conserving genetic algorithm for multimodal function optimization. Evol Comput 10(3):207–234
21. Madsen K, Žilinskas J Testing branch-and-bound methods for global optimization. http://www2.imm.dtu.dk/documents/ftp/tr00/tr05_00.pdf
22. Mahfoud SW (1995) Niching methods for genetic algorithms. PhD thesis, University of Illinois at Urbana-Champaign, USA
23. Mishra S Some new test functions for global optimization and performance of repulsive particle swarm method. http://mpra.ub.uni-muenchen.de/2718
24. Nelder JA, Mead R (1965) A simplex method for function minimization. Comput J 7(4):308–313
25. Parsopoulos K, Plagianakos V, Magoulas G, Vrahatis M (2001) Objective function "Stretching" to alleviate convergence to local minima. Nonlinear Anal 47(5):3419–3424
26. Parsopoulos KE, Vrahatis MN (2001) Modification of the particle swarm optimizer for locating all the global minima. In: Kurova V et al (eds) Artificial neural networks and genetic algorithms. Springer, Berlin, pp 324–327
27. Parsopoulos K, Vrahatis M (2002) Recent approaches to global optimization problems through particle swarm optimization. Nat Comput 1(2–3):235–306
28. Parsopolos K, Vrahatis M (2004) On the compuation of all minima through particle swarm optimization. IEEE Trans Evol Comput 8(3):211–224
29. Pavlyukevich I (2007) Non-local search and simulated annealing Lévy flights. J Comput Phys 226(2):1830–1844
30. Pavlyukevich I (2007) Cooling down Lévy flights. J Phys A: Math Theor 40(41):12299–12313
31. Petrowski A (1996) A clearing procedure as a Niching method for genetic algorithms. In: IEEE international conference on evolutionary computation, Japan, pp 798–803
32. Price K, Storn RM, Lampinen JA (2005) Differential evolution: a practical approach to global optimization. Natural computing series. Springer, New York

33. Rahnamayan S, Tizhoosh HR, Salama MMA (2008) Opposition-based differential evolution. IEEE Trans Evol Comput 12(1):64–79
34. Reynolds AM, Frey MA (2007) Free-flight odor tracking in dorsophila is consistent with an optimal intermittent scale-free search. PLoS One 2(4):e354
35. Richer TJ, Blackwell TM (2006) The Lévy particle swarm. In: Proceedings of IEEE congress on evolutionary computation, Canada, pp 808–815
36. Rönkkönen J (2009) Continuous multimodal gloabal optimization with differential evolution-based methods. PhD thesis, Lappeenranta University of Technology, Finland
37. Saha A, Deb K (2010) A bi-criterion approach to multimodal optimization: self-adaptive approach. In: Proceedings of international conference on simulated evolution and learning, India, pp 95–104
38. Schwefel H-P (1995) Evolution and optimum seeking. Wiley, New York
39. Stoneking D, Bilbro G, Trew R, Gilmore P, Kelley CT (1992) Yield optimization Using a GaAs process simulator coupled to a physical device model. IEEE Trans Microw Theory Tech 40(7):1353–1363
40. Tasoulis D, Plagianakos V, Vrahatis M (2005) Clustering in evolutionary algorithms to effectively compute simultaneously local and global minima. In: IEEE congress on evolutionary computation, Scotland, pp 1847–1854
41. Torczon V (1991) On the convergence of the multidimensional search algorithm. SIAM J Optim 1(1):123–145
42. Yang X-S, Deb S (2009) Cuckoo search via Lévy flights. In: Proceedings of world congress on nature and biological inspired computing, India, pp 210–214
43. Yang X-S, Deb S (2010) Engineering optimization by Cuckoo search. Int J Math Model Numer Optim 1(4):330–343
44. Yang XS (2010) A new metaheuristic bat-inspired algorithm. In: Gonzalez JR et al (eds) Nature inspired cooperative strategies for optimization. Studies in computational intelligence. Springer, Berlin, pp 65–74
45. Zaharie D (2004) Extension of differential evolution algorithm for multimodal optimization. In: International symposium on symbolic and numeric algorithms for scientific computing, Romania, pp 523–534

http://www.springer.com/978-3-319-50919-8

Nature-Inspired Computing and Optimization
Theory and Applications
Patnaik, S.; Yang, X.-S.; Nakamatsu, K. (Eds.)
2017, XXI, 494 p. 191 illus., 43 illus. in color., Hardcover
ISBN: 978-3-319-50919-8