

## Research Article

# A Novel Bat Algorithm Based on Differential Operator and Lévy Flights Trajectory

Jian Xie,<sup>1</sup> Yongquan Zhou,<sup>1,2</sup> and Huan Chen<sup>1</sup>

<sup>1</sup> College of Information Science and Engineering, Guangxi University for Nationalities, Nanning, Guangxi 530006, China

<sup>2</sup> Guangxi Key Laboratory of Hybrid Computation and IC Design Analysis, Guangxi University for Nationalities, Nanning, Guangxi 530006, China

Correspondence should be addressed to Yongquan Zhou; [yongquanzhou@126.com](mailto:yongquanzhou@126.com)

Received 16 August 2012; Accepted 1 February 2013

Academic Editor: Christian W. Dawson

Copyright © 2013 Jian Xie et al. 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.

Aiming at the phenomenon of slow convergence rate and low accuracy of bat algorithm, a novel bat algorithm based on differential operator and Lévy flights trajectory is proposed. In this paper, a differential operator is introduced to accelerate the convergence speed of proposed algorithm, which is similar to mutation strategy “DE/best/2” in differential algorithm. Lévy flights trajectory can ensure the diversity of the population against premature convergence and make the algorithm effectively jump out of local minima. 14 typical benchmark functions and an instance of nonlinear equations are tested; the simulation results not only show that the proposed algorithm is feasible and effective, but also demonstrate that this proposed algorithm has superior approximation capabilities in high-dimensional space.

## 1. Introduction

Nowadays, since the evolutionary algorithm can solve some problem that the traditional optimization algorithm cannot do easy, the evolutionary algorithms are widely applied in different fields, such as the management science, engineering optimization, scientific computing. More and more modern metaheuristic algorithms inspired by nature or social phenomenon are emerging and they become increasingly popular, for example, particles swarms optimization (PSO) [1], firefly algorithm (FA) [2, 3], artificial chemical reaction optimization algorithm (ACROA) [4], glowworm swarms optimization (GSO) [5], invasive weed optimization (IWO) [6], differential evolution (DE) [7–9], bat algorithm (BA) [2, 10], and so on [11–15]. Some researchers have proposed their hybrid versions by combining two or more algorithms.

Bat Algorithm (BA) is a novel metaheuristic optimization algorithm based on the echolocation behaviour of microbats, which was proposed by Yang in 2010 [2, 10]. This algorithm gradually aroused people’s close attention, and which is increasingly applied to different areas. Tsai et al. (2011)

proposed an improved EBA to solve numerical optimization problems [16]. A multiobjective bat algorithm (MOBA) is proposed by Yang (2011) [17], which is first validated against a subset of test functions, and then applied to solve multi-objective design problems such as welded beam design. In 2012, Bora et al. applied BA to solve the Brushless DC Wheel Motor Problem [18]. Although the basic BA has remarkable property compared against several traditional optimization methods, the phenomenon of slow convergence rate and low accuracy still exists. Therefore, in this paper, we put forward an improved bat algorithm based on differential operator and Lévy flights trajectory (DLBA), the purpose is to improve the convergence rate and precision of bat algorithm. At the end of this paper, we tested 14 typical benchmark functions and applied them to solve nonlinear equations; the simulation results not only showed that the proposed algorithm is feasible and effective, which is more robust, but also demonstrated the superior approximation capabilities in high-dimensional space.

The rest of this study is organized as follows. In Section 2, the basic bat algorithm and Lévy flights were described.

In Section 3, we gave the design framework of DLBA. The implementation and comparison of improved algorithm are presented in Section 4. Finally, we concluded this paper in Section 5.

## 2. Bat Algorithm, Lévy Flights, and Nonlinear Equations

*2.1. Behaviour of Microbats.* Most of microbats have advanced capability of echolocation. These bats can emit a very loud and short sound pulse; the echo that reflects back from the surrounding objects is received by their extraordinary big auricle. Then, this feedback information of echo is analyzed in their subtle brain. They not only can discriminate direction for their own flight pathway according to the echo, but also can distinguish different insects and obstacles to hunt prey and avoid a collision effectively in the day or night.

*2.2. Bat Algorithm (see [10]).* First of all, let us briefly review the basics of the BA for single-objective optimization. In the basic BA developed by Yang in 2010, in order to propose the bat algorithm inspired by the echolocation characteristics of microbats, the following approximate or idealised rules were used.

*IR1.* All bats use echolocation to sense distance, and they also “know” the difference between food/prey and background barriers in some magical way.

*IR2.* Bats fly randomly with velocity  $v_i$  at position  $x_i$  with a fixed frequency  $f_{\min}$ , varying wavelength  $\lambda$ , and loudness  $A_0$  to search for prey. They can automatically adjust the wavelength (or frequency) of their emitted pulses and adjust the rate of pulse emission  $r \in [0, 1]$ , depending on the proximity of their target.

*IR3.* Although the loudness can vary in many ways, we assume that the loudness varies from a large (positive)  $A_0$  to a minimum constant value  $A_{\min}$ .

In addition, for simplicity, they also use the following approximations: in general, the frequency  $f$  in a range  $[f_{\min}, f_{\max}]$  corresponds to a range of wavelengths  $[\lambda_{\min}, \lambda_{\max}]$ . In fact, they just vary in the frequency while fixed in the wavelength  $\lambda$  and assume  $f \in [0, f_{\max}]$  in their implementation. This is because  $\lambda$  and  $f$  are related due to the fact that  $\lambda f = v$  is constant.

In simulations, they use virtual bats naturally to define the updated rules of their positions  $x_i$  and velocities  $v_i$  in a D-dimensional search space. The new solutions  $x_i^t$  and velocities  $v_i^t$  at time step  $t$  are given by

$$\begin{aligned} f_i &= f_{\min} + (f_{\max} - f_{\min})\beta, \\ v_i^t &= v_i^{t-1} + (x_i^t - x_*)f_i, \\ x_i^t &= x_i^{t-1} + v_i^t, \end{aligned} \quad (1)$$

where  $\beta \in [0, 1]$  is a random vector drawn from a uniform distribution. Here,  $x_*$  is the current global best location (solution) which is located after comparing all the solutions among all the  $n$  bats.

For the local search part, once a solution is selected among the current best solutions, a new solution for each bat is generated locally using random walk:

$$x_{\text{new}} = x_{\text{old}} + \varepsilon A_t, \quad (2)$$

where  $\varepsilon \in [-1, 1]$  is a random number, while  $A_t = \langle A_i^t \rangle$  is the average loudness of all the bats at this time step.

Furthermore, the loudness  $A_i$  and the rate  $r_i$  of pulse emission have to be updated accordingly as the iterations proceed. These formulas are

$$A_i^{t+1} = \alpha A_i^t, \quad (3)$$

$$r_i^{t+1} = r_i^0 [1 - \exp(-\gamma t)], \quad (4)$$

where  $\alpha$  and  $\gamma$  are constants.

Based on these approximations and idealization, the basic steps of the bat algorithm can be summarized in the Pseudocode 1.

*2.3. Lévy Flights.* Lévy flights are Markov processes, which differ from regular Brownian motion, whose individual jumps have lengths that are distributed with the probability density function (PDF)  $\lambda(x)$  decaying at large  $x$  as  $\lambda(x) = |x|^{-1-\alpha}$  with  $0 < \alpha < 2$ . Due to the divergence of their variance,  $\lambda(x) \approx |x|^{-1-\alpha}$ , extremely long jumps may occur, and typical trajectories are self-similar, on all scales showing clusters of shorter jumps interspersed by long excursions [19]. Lévy flight has the following properties [20]:

- (1) *Stability:* distribution of the sum of independent identically distributed stable random variables equal to distribution of each variable.
- (2) *Power law asymptotics* (“heavy tails”).
- (3) *Generalized Central Limit Theorem:* The central limit theorem states that the sum of a number of independent and identically distributed (i.i.d.) random variables with finite variances will tend to a normal distribution as the number of variables grows.
- (4) *Which has an infinite variance with an infinite mean value.*

Due to these remarkable properties of stable distributions, it is now believed that the Lévy statistics provide a framework for the description of many natural phenomena in physical, chemical, biological, and economical systems from a general common point of view.

Furthermore, various studies have shown that the flight behaviour of many animals and insects has demonstrated the typical characteristics of Lévy lights. A recent study by Reynolds and Frye shows that fruit flies, or *Drosophila melanogaster*, explore their landscape using a series of straight flight paths punctuated by a sudden 90° turn, leading to a Lévy flight-style intermittent scale-free search pattern [21]. Studies on human behaviour such as the Ju’hoansi hunter-gatherer foraging patterns also show the typical feature of Lévy flights [22]. The conclusion that light is related to Lévy flights is proposed by Barthelemy et al. (2008) [23]. The

```

Objective function  $f(x)$ ,  $x = [x_1, x_2, \dots, x_d]^T$ 
Initialize the bat population  $x_i$  ( $i = 1, 2, \dots, n$ ) and  $v_i$ 
Define pulse frequency  $f_i$  at  $x_i$  initialize pulse rates  $r_i$  and the loudness  $A_i$ 
While ( $t < \text{Max number of iterations}$ )
  Generate new solutions by adjusting frequency,
  and updating velocities and locations/solutions (1)
  if ( $\text{rand} > r_i$ )
    Select a solution among the best solutions
    Generate a local solution around the selected best solution
  end if
  Generate a new solution by flying randomly
  if ( $\text{rand} < A_i$  &  $f(x_i) < f(x_*)$ )
    Accept the new solutions
    Increase  $r_i$  and reduce  $A_i$ 
  end if
  Rank the bats and find the current best  $x_*$ 
end while
Postprocess results and visualization

```

## PSEUDOCODE 1

study by Mercadier et al. shows that the Lévy flights of photons in hot atomic vapours (2009) [24]. Subsequently, such behaviour has been applied to optimization and optimal search, and preliminary results show its promising capability.

2.4. *Description of the Nonlinear Equations (see [25]).* The general form of nonlinear equations with real variables is described as follows:

$$g_i(X) = 0, \quad i = 1, 2, \dots, n, \quad (5)$$

where  $X = (x_1, x_2, \dots, x_n) \in D \subset R^n$ ,  $D = \{(x_1, x_2, \dots, x_n) \mid a_i \leq x_i \leq b_i, i = 1, 2, \dots, n\}$ .

In solving nonlinear equations process for DLBA, the fitness function can be constructed by

$$G(X) = \sum_{i=1}^n |g_i(x)|. \quad (6)$$

So, the solving of nonlinear equations can be translated into an optimization problem in domain  $D$ :

$$\begin{aligned} \min \quad & G(X) \\ \text{s.t} \quad & (x_1, x_2, \dots, x_n) \in D. \end{aligned} \quad (7)$$

Consequently, the optimal value of (7) is exactly the solution of (5).

### 3. DLBA

Inspired by Yang's method, we propose an improved bat algorithm based on differential operator and Lévy-flights trajectory (DLBA) based on the basic structure of BA and re-estimate the characters used in the original BA. In DLBA, not only the movement of the bat is quite different from the original BA, but also the local search process is different.

In DLBA, the frequency fluctuates up and down, which can change self-adaptively, and the differential operator is introduced, which is similar to the mutation operation of DE, the frequency  $f$  is similar to the scale factor  $F$  of DE/best/2. So, the frequency updated formulae of a bat are defined as follows:

$$f_{1i}^t = \left( (f_{1,\min} - f_{1,\max}) \frac{t}{n_t} + f_{1,\max} \right) \beta_1, \quad (8)$$

$$f_{2i}^t = \left( (f_{2,\max} - f_{2,\min}) \frac{t}{n_t} + f_{2,\min} \right) \beta_2, \quad (9)$$

where  $\beta_1, \beta_2 \in [0, 1]$  is a random vector drawn from a uniform distribution,  $f_{1,\max} = f_{2,\max}$ ,  $f_{1,\min} = f_{2,\min}$ ,  $n_t$  is a fixed parameter. In DLBA, the position  $x_i^t$  of each bat individual are updated with (10), which is different from original BA. This can preferably incorporate the echolocation characteristics of microbats:

$$x_i^{t+1} = x_{\text{best}}^t + f_{1i}^t (x_{r1}^t - x_{r2}^t) + f_{2i}^t (x_{r3}^t - x_{r4}^t), \quad (10)$$

where  $x_{\text{best}}^t$  is the current global best location (solution) which is located after comparing all the solutions among all the  $n$  bats in  $t$  generation,  $x_{r_i}^t$  is the bat individual in the bat swarm, and this can be achieved by randomization.

In addition, Lévy flight has the prominent properties increase the diversity of population, sequentially, which can make the algorithm effectively jump out of the local optimum. So, we let these bats perform the Lévy flights with (11) before the position updating:

$$x_i^t = \hat{x}_i^{t-1} + \mu \text{sign}[\text{rand} - 0.5] \oplus \text{Levy}, \quad (11)$$

where  $\mu$  is a random parameter drawn from a uniform distribution,  $\text{sign} \oplus$  means entrywise multiplications,  $\text{rand} \in [0, 1]$ , and random step length Levy obeys Lévy distribution:

$$\text{Levy} \sim u = t^{-\lambda}, \quad (1 < \lambda \leq 3). \quad (12)$$

```

Input: Objective function  $f(x)$ ,  $x = [x_1, x_2, \dots, x_d]^T$ ; algorithm's parameters  $n, f_{\min}, f_{\max}, \alpha, n_t$ ;
BEGIN
  For  $i = 1$  To  $n$  do
    Initialize  $x_i^t, r_i^0, A_i^0$ ;
    Calculating the initial fitness;
  End For
  While ( $t < \text{iter}_{\max}$ )
    Update position use formula (8)~(12);
    Evaluate fitness  $f(x_i)$  to find  $x_{\text{best}}$ ;
    For  $i = 1$  To  $n$  do
      If ( $\text{rand} > r_i$ )
        Generate a  $x_{\text{new}}$  around the selected best solution use formula (2);
      End If
    End For
    Evaluate fitness  $f(x_{\text{new}})$ ;
    For  $i = 1$  To  $n$  do
      If ( $\text{rand} < A_i$ )
        Generate a  $x_{\text{new}}$  around the selected best solution use formula (13);
      End If
    End For
    Evaluate fitness  $f(x_{\text{new}})$ ;
    If ( $f(x_{\text{best}}^{t+1}) < f(x_{\text{best}}^t)$ )
      Update  $r_i^t, A_i^t$  use formula (3), (14);
    End If
     $t = t + 1$ ;
  End While
END
Output:  $x_{\text{best}}, \text{fitness}_{\min}$ ;

```

## PSEUDOCODE 2

On the other hand, each bat should have different values of loudness and pulse emission rate, while the rate of pulse emission is relatively low and the loudness is relatively high. During the search process, the loudness usually decreases, while the rate of pulse emission increases. Bats' position variation is influenced by the pulse emission rate and loudness as well. Firstly, pulse emission rate  $r_i$  causes fluctuation of position using (2); sequentially, more and more new position can be explored. Secondly, loudness  $A_i$  is designed to strengthen local search and to guide bats find better solutions:

$$x_{\text{new}} = x_{\text{best}} + \eta r_t, \quad (13)$$

where  $\eta \in [-1, 1]$  is a random parameter, and  $x_{\text{best}}$  is the current global best location (solution) in whole bats swarm. While  $r_t = \langle r_i^t \rangle$  is the average pulse emission rate of all the bats at this generation.

The new rate of pulse emission  $r_i^t$  and loudness  $A_i^t$  at time step  $t$  are given by

$$r_i^{t+1} = r_i^t \left( \frac{t}{n_t} \right)^3, \quad (14)$$

where  $r_i^t$  is time varying; their loudness and emission rates will be updated only if the best solution of the current generation is better than the best solution of last generation, which means that these bats are moving towards the optimal solution. The pseudocode of DLBA can be depicted as in Pseudocode 2.

## 4. The Simulation and Analysis

**4.1. Parametric Studies.** The proposed DLBA is implemented in MATLAB, simulation platform: CPU Intel Xeon E5405@2.00 GHz; OS: Microsoft Windows Server 2003 Enterprise Edition SP2; RAM: 1GHz; Matlab Version: R2009a. The parameter setting for BA and DLBA are listed in Table 1, the parameters of BA are recommended in the original article.

The stopping criterion can be defined in many ways. We adopt two terminated criteria: we can use a given tolerance ( $\text{Tol} = 1.0e - 5$ ) for a test function that have certain minimum value in theory; on the contrary, each simulation run terminates when a certain number of function evaluations (FEs) have been reached. In this paper, FEs could be obtained by population size multiplied by the number of iteration. In our experiment, FEs = 24000, BA (= 300\*40\*2); DLBA (= 200\*40\*3).

**4.2. Benchmark Test Function.** In order to validate the validity of DLBA, we selected the 14 benchmark functions to experimentize. The benchmark set include unimodal, multimodal, high-dimensional, and low-dimensional unconstrained optimization benchmark functions, where  $f_1-f_3$  are unimodal functions,  $f_4-f_{14}$  are multimodal functions;  $f_1-f_9$  have certain theoretical minimum and  $f_{10}-f_{14}$  have uncertain theoretical minimum.

TABLE I: The parameter set of BA and DLBA.

Algorithms	$n$	$f_{\min}$	$f_{\max}$	$A_i^0$	$t_i^0$	$\alpha$	$\gamma$	$n_t$
BA	40	0	100	(1,2)	(0,0.1)	0.9	0.9	—
DLBA	40	0	1	(1,2)	(0,0.1)	0.9	—	5000

(1)  $f_1$ : sphere function (the first function of De Jong's test set),

$$f(x) = \sum_{i=1}^n x_i^2, \quad -10 \leq x_i \leq 10. \quad (15)$$

Here,  $n$  is dimensionality; this function has a global minimum  $f_{\min} = 0$  at  $x_* = (0, 0, \dots, 0)$ .

(2)  $f_2$ : Schwefel's problem 2.22,

$$f(x) = \sum_{i=1}^n |x_i| + \prod_{i=1}^n |x_i|, \quad -10 \leq x_i \leq 10, \quad (16)$$

whose global minimum is obviously  $f_{\min} = 0$  at  $x_* = (0, 0, \dots, 0)$ .

(3)  $f_3$ : Rosenbrock function,

$$f(x) = \sum_{i=1}^{n-1} \left[ (x_i - 1)^2 + 100(x_{i+1} - x_i^2)^2 \right], \quad (17)$$

$$-2.408 \leq x_i \leq 2.408,$$

which has a global minimum  $f_{\min} = 0$  at  $x_* = (1, 1, \dots, 1)$ .

(4)  $f_4$ : Eggcrate function,

$$f(x, y) = x^2 + y^2 + 25(\sin^2 x + \sin^2 y), \quad (18)$$

$$(x, y) \in [-2\pi, 2\pi].$$

This 2-dimensional test function obviously gets the global minimum  $f_{\min} = 0$  at  $(0, 0)$ .

(5)  $f_5$ : Ackley's function,

$$f(x) = -20 \exp \left[ -\frac{1}{5} \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2} \right] - \exp \left[ \frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i) \right] + 20 + e, \quad (19)$$

$$-30 \leq x \leq 30.$$

This function has a global minimum  $f_{\min} = 0$  at  $x_* = (0, 0, \dots, 0)$ , which is a multimodal function.

(6)  $f_6$ : Griewangk's function,

$$f(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1, \quad -600 \leq x_i \leq 600. \quad (20)$$

Its global minimum equal  $f_{\min} = 0$  is obtainable for  $x_* = (0, 0, \dots, 0)$ , the number of local minima for arbitrary  $n$  is

unknown, but in the two-dimensional case there are some 500 local minima.

(7)  $f_7$ : Salomon's function,

$$f(x) = -\cos\left(2\pi\sqrt{\sum_{i=1}^n x_i^2}\right) + 0.1\sqrt{\sum_{i=1}^n x_i^2} + 1, \quad -5 \leq x_i \leq 5, \quad (21)$$

which has a global minimum  $f_{\min} = 0$  at  $x_* = (0, 0, \dots, 0)$ , and is a multimodal function.

(8)  $f_8$ : Rastrigin's function.

$$f(x) = 10n + \sum_{i=1}^n [x_i^2 - 10\cos(2\pi x_i)], \quad -5.12 \leq x_i \leq 5.12, \quad (22)$$

whose global minimum is  $f_{\min} = 0$  at  $x_* = (0, 0, \dots, 0)$ ; for  $n = 2$ , there are about 50 local minimizers arranged in a lattice-like configuration.

(9)  $f_9$ : Zakharov's function,

$$f(x) = \sum_{i=1}^n x_i^2 + \left(\frac{1}{2} \sum_{i=1}^n i x_i\right)^2 + \left(\frac{1}{2} \sum_{i=1}^n i x_i\right)^4, \quad -10 \leq x_i \leq 10, \quad (23)$$

whose global minimum is  $f_{\min} = 0$  at  $x_* = (0, 0, \dots, 0)$ ; it is a multimodal function as well.

(10)  $f_{10}$ : Easom's function,

$$f(x, y) = -\cos(x) \cos(y) \exp[-(x-\pi)^2 + (y-\pi)^2], \quad (24)$$

$$-10 \leq x, y \leq 10,$$

whose global minimum is  $f_{\min} = -1$  at  $(\pi, \pi)$ ; it has many local minima.

(11)  $f_{11}$ : Schwefel's function,

$$f(x) = -\sum_{i=1}^n x_i \sin\left(\sqrt{|x_i|}\right), \quad -500 \leq x_i \leq 500, \quad (25)$$

whose global minimum is  $f_{\min} \approx -418.9829n$  occurring at  $x_* \approx (420.9687, \dots, 420.9687)$ .

(12)  $f_{12}$ : Shubert's function,

$$f(x, y) = \left[ \sum_{i=1}^5 i \cos(i+(i+1)x) \right] \cdot \left[ \sum_{i=1}^5 i \cos(i+(i+1)y) \right], \quad (26)$$

$$-10 \leq x, y \leq 10.$$

The number of local minima for this problem is not known but for  $n = 2$ , the function has 760 local minima, 18 of which are global with  $f_{\min} \approx -186.7309$ .

(13)  $f_{13}$ : Xin-She Yang's function,

$$f(x) = -\left(\sum_{i=1}^n |x_i|\right) \exp\left(-\sum_{i=1}^n x_i^2\right), \quad -10 \leq x_i \leq 10, \quad (27)$$

which has multiple global minima, for example, for  $n = 2$ , it has 4 equal minima  $f_{\min} = -1/\sqrt{e} \approx -0.6065$  at  $(0.5, 0.5)$ ,  $(0.5, -0.5)$ ,  $(-0.5, 0.5)$ , and  $(-0.5, -0.5)$ .

(14)  $f_{14}$ : "Drop Wave" function,

$$f(x) = -\frac{1 + \cos\left(12\sqrt{\sum_{i=1}^n x_i^2}\right)}{(1/2)\left(\sum_{i=1}^n x_i^2\right) + 2}, \quad -5.12 \leq x_i \leq 5.12. \quad (28)$$

This two-variable function is a multimodal test function, whose global minimum is  $f_{\min} = -1$  occurs at  $x_* = (0, 0, \dots, 0)$ .

**4.3. Comparison of Experimental Results.** The 2D landscape of Schwegel's function is shown in Figure 1, and this global minimum can be found after about 720 FEs for 40 bats after 6 iterations as shown in Figures 2, 3, and 4.

We adopt different terminated criteria aiming at different benchmark function, we perform 100 times independently for each test function, and the record is given in Tables 2 and 3.

From Table 2, we can see that the DLBA performs much better than the basic bat algorithm, which converges much faster than BA under the fixed accuracy  $\text{Tol} = 1.0e - 5$ . In our experiment, for the function  $f_4$ , we proposed that an algorithm only costs 6 generations under the best situation, and the average generations is 10. Furthermore, for the first group of test functions  $f_1 - f_9$ , the DLBA averagely expend 43.9 generations when each function attains its terminated criteria; however, the BA needs 200 generations invariably, and the convergence speed of DLBA advances 298.57% times. On other hand, the DLBA which obtained the accuracy of the solution is much superior to BA, which is more approximate to the theoretical value. In a word, it demonstrated that DLBA has fast convergence rate and high precision of the solutions.

In Table 3, the five functions ( $f_{10} - f_{14}$ ) independently runs 100 times under the 24000 Fes; we can clearly see the precision of DLBA is obviously superior to the bat algorithm. Some benchmark functions can easily attain the theoretical optimal value. In addition, the standard deviation of DLBA is relatively low. It shows that DLBA has superior approximation ability.

Observe Tables 2 and 3, no matter high-dimensional or low-dimensional, we can find that DLBA can quickly converge to the global minima. Furthermore, Figures 5, 6, 7, 8, and 9 show the convergence curves for some of the functions from a particular run of DLBA and BA, which end at 200 generations. The adaptive scheme generally converges faster than the basic scheme. Here, we select  $f_2$  ( $D = 20$ ),  $f_4$  ( $D = 2$ ),  $f_7$  ( $D = 5$ ),  $f_{11}$  ( $D = 2$ ), and  $f_{13}$  ( $D = 2$ ).

DLBA not only has superior approximation ability in low-dimensional space, but also has excellent global search ability in high-dimensional situation. Table 4 is the experimental

result that DLBA performs 50 times independently under the high-dimensional situation. As shown in Table 4, we can be conscious that the DLBA is effective under the multidimensional condition, and acquired solution has higher accuracy, even approximate the theoretical value.

Figures 10, 11, 12, 13, and 14 are the distribution map of optimal fitness; that is the selected functions independently perform 50 times under the multidimensional situation. Figure 13 show that two "straight line", we can see in the figure that DLBA reaches the global optimum(-1), however, BA fluctuates around 0. In order to display the fact of fluctuation, we magnify the two "straight line", and the amplifying effect are depicted in Figures 15 and 16. According to the experimental results which are obtained from selected test functions, DLBA presents higher precision than the original BA on minimizing the outcome as the optimization goal.

#### 4.4. An Application for Solving Nonlinear Equations

*Interval Arithmetic Benchmark (IAB).* We consider one benchmark problem proposed from interval arithmetic, the benchmark consists of the following system of [25]:

$$\begin{aligned} 0 &= x_1 - 0.25428722 - 0.18324757x_4x_3x_9, \\ 0 &= x_2 - 0.37842197 - 0.16275449x_1x_{10}x_6, \\ 0 &= x_3 - 0.27162577 - 0.16955071x_1x_2x_{10}, \\ 0 &= x_4 - 0.19807914 - 0.15585316x_7x_1x_6, \\ 0 &= x_5 - 0.44166728 - 0.19950920x_7x_6x_3, \\ 0 &= x_6 - 0.14654113 - 0.18922793x_8x_5x_{10}, \\ 0 &= x_7 - 0.42937161 - 0.21180486x_2x_5x_8, \\ 0 &= x_8 - 0.07056438 - 0.17081208x_1x_7x_6, \\ 0 &= x_9 - 0.34504906 - 0.19612740x_{10}x_6x_8, \\ 0 &= x_{10} - 0.42651102 - 0.21466544x_4x_8x_1. \end{aligned} \quad (29)$$

Some of the solutions obtained as well as the function values (which represent the values of the system's equations obtained by replacing the variable values) are presented in Table 5. From Table 5, we can obviously observe that the function value of each equation solved by DLBA is superior to which solved by EA [25], depending on the statistics of the 40 function values, the precision of function values is enhanced  $5.199820E + 06$  times by DLBA. The convergence curve of DLBA is depicted in Figure 17. Figure 17 show that DLBA has fast convergence rate for solving nonlinear equations. The achieved optimal fitness (the sum of function value with absolute value) in 50 times independent run is depicted in Figure 18. In order to reflect the precision of fitness, we magnify Figure 18, these optimal fitness that less than or equal to 0.001 are plotted in Figure 19. We can found that there are 32 times optimal fitness is less than 0.001.

TABLE 2: Comparison of BA and DLBA for several test functions under the first terminated criteria.

TC	BF	Method	Fitness				Iteration		
			Min	Mean	Max	Std	Min	Mean	Max
Tol = 1.0e - 5, iter <sub>max</sub> = 200	$f_1 (D = 30)$	DLBA	4.30568008E-08	4.80356915E-06	9.96163402E-06	3.02060716E-06	10	17.6	26
		BA	56.81732462	130.61959271	223.82694528	37.22960351	200	200	200
	$f_2 (D = 20)$	DLBA	5.66761254E-07	6.33091639E-06	9.88929757E-06	2.48867687E-06	21	29.4	37
		BA	22.77136624	93.35062167	2733.46090499	269.41410301	200	200	200
	$f_3 (D = 10)$	DLBA	5.66716240	7.39029403	8.28857171	0.37001430	200	200	200
		BA	82.24908050	294.95948479	640.76052284	109.67161818	200	200	200
	$f_4 (D = 2)$	DLBA	6.23475433E-09	3.93352701E-06	9.97022879E-06	2.77208752E-06	6	10	20
		BA	1.05265280E-04	0.20705750	1.00771980	0.23116047	200	200	200
	$f_5 (D = 5)$	DLBA	4.71737180E-07	6.40969812E-06	9.96088247E-06	2.71187009E-06	16	25.6	39
		BA	1.15883977	3.02092450	5.86431360	0.69052427	200	200	200
	$f_6 (D = 5)$	DLBA	1.38176777E-07	5.01027883E-06	9.92998256E-06	2.77338474E-06	11	18.4	56
		BA	0.06234957	8.42851961	29.90707979	6.12812390	200	200	200
	$f_7 (D = 5)$	DLBA	7.78490674E-07	6.57288799E-06	9.97486019E-06	2.36637154E-06	18	46	114
		BA	0.09987344	0.15318115	0.30215065	0.04923496	200	200	200
	$f_8 (D = 5)$	DLBA	5.99983281E-08	4.47052788E-06	9.85695441E-06	2.81442645E-06	15	33	87
		BA	7.24547772	17.11077150	25.90638177	4.37190073	200	200	200
	$f_9 (D = 5)$	DLBA	6.65070016E-09	3.88692603E-06	9.69841027E-06	2.75444981E-06	10	15.1	23
		BA	0.11106843	1.75623424	8.48211975	1.41343722	200	200	200

TABLE 3: Comparison of BA and DLBA for several test functions under the second terminated condition.

TC	BF	Method	Fitness			
			Min	Mean	Max	Std
FEs	$f_{10} (D = 2)$	DLBA	-1	-1	-1	9.06493304E - 17
		BA	-0.99964845	-0.98245546	-0.90221232	0.01759602
	$f_{11} (D = 2)$	DLBA	-418.98288727	-415.83523725	-300.54455266	15.53948263
		BA	-418.98287874	-406.98044144	-328.10751671	22.54615132
	$f_{12} (D = 5)$	DLBA	-186.73090883	-186.73090883	-186.73090883	4.31613544E - 13
		BA	-186.67541533	-184.01677767	-174.95599301	2.26333511
	$f_{13} (D = 2)$	DLBA	-0.60653066	-0.60653066	-0.60653066	8.63947249E - 16
		BA	-0.60652631	-0.60411559	-0.59656608	2.18888608E - 03
	$f_{14} (D = 5)$	DLBA	-1	-1	-1	0
		BA	-0.93624514	-0.80454167	-0.56977584	0.10270830

TABLE 4: Comparison of DLBA and BA under the high-dimensional situation.

BF	Method	Fitness			
		Min	Mean	Max	Std
$f_6 (D = 128)$	DLBA	0	0	0	0
	BA	2587.38759264	2811.40649000	2966.56723398	93.96861133
$f_9 (D = 256)$	DLBA	2.66885939E - 94	5.12093542E - 69	2.56023720E - 67	3.62071553E - 68
	BA	7503.39306546	8256.61302714	11506.79192998	613.37556609
$f_8 (D = 320)$	DLBA	0	0	0	0
	BA	4468.59039467	4723.44624200	5008.57109760	131.18818777
$f_{14} (D = 512)$	DLBA	-1	-1	-1	0
	BA	-1.48358700E - 03	-1.31084173E - 03	-1.20435493E - 03	5.99288716E - 05
$f_1 (D = 1024)$	DLBA	2.63790469E - 109	4.73742970E - 87	2.03861481E - 85	2.89031313E - 86
	BA	18463.48950157	19752.26686813	21239.23111285	567.27878361

TABLE 5: Outcome of DLBA for Interval Arithmetic Benchmark And in Comparison with EA.

Method Solutions	EA		DLBA	
	Variables values	Functions values	Variables values	Functions values
Sol. 1	0.0464905115	0.2077959240	0.2578335208	7.22997083E - 08
	0.1013568357	0.2769798846	0.3810968901	-2.8487905E - 07
	0.0840577820	0.1876863212	0.2787447331	-2.63974619E - 07
	-0.1388460309	0.3367887114	0.2006720536	3.06212480E - 06
	0.4943905739	0.0530391321	0.4452522319	7.74692656E - 07
	-0.0760685163	0.2223730535	0.1491853777	1.33597657E - 06
	0.2475819110	0.1816084752	0.4320098178	-7.87244534E - 09
	-0.0170748156	0.0878963860	0.0734062202	3.41255029E - 06
	0.0003667535	0.3447200366	0.3459672005	3.24064555E - 07
	0.1481119311	0.2784227489	0.4273251429	-1.18429644E - 06
Sol. 2	0.1224819761	0.1318552790	0.2578332642	-1.45675448E - 07
	0.1826200685	0.1964428361	0.3810968252	-3.41069930E - 07
	0.2356779803	0.0364987069	0.2787462357	1.19723259E - 06
	-0.0371150470	0.2354890155	0.2006687659	-1.76074071E - 08
	0.3748181856	0.0675753064	0.4452514819	2.89773434E - 07
	0.2213311341	0.0739986588	0.1491839999	9.16571896E - 08
	0.0697813043	0.3607038292	0.4319795315	-3.01421570E - 05
	0.0768058043	0.0059182979	0.0734021258	-4.53845535E - 07
	-0.0312153867	0.3767487763	0.3459672388	4.15572858E - 07
	0.1452667120	0.2811693568	0.4273281275	1.85998012E - 06
Sol. 3	0.0633944399	0.1908436653	0.2578328072	-5.95667763E - 07
	0.1017426933	0.2767897367	0.3810969084	-2.39480772E - 07
	-0.1051842285	0.3769063436	0.2787447816	-2.04170456E - 07
	-0.0477059943	0.2460900702	0.2006696507	6.71241513E - 07
	0.4149858326	0.0260337751	0.4452514467	-4.38810984E - 09
	0.1215195321	0.0256054760	0.1491841089	1.89336147E - 07
	0.2539777159	0.1761486401	0.4320126793	2.97845790E - 06
	0.0843972823	0.1349869851	0.0734028724	7.79163472E - 08
	-0.0534132992	0.3986395691	0.3459668311	3.30962522E - 09
	0.0880998746	0.3383563536	0.4273256312	-6.46813249E - 07
Sol. 4	0.1939820199	0.0603335280	0.2578382574	5.02240252E - 06
	0.0152114400	0.3633514726	0.3810978088	8.03072038E - 07
	0.1618654345	0.1097465792	0.2787430938	-1.98215448E - 06
	0.0056985809	0.9114653768	0.2006688671	3.21352180E - 08
	0.1904538879	0.2502358229	0.4452573447	6.19105762E - 06
	-0.1623604033	0.3089460561	0.1491746444	-9.01451453E - 06
	0.1864448178	0.2428992222	0.4320068557	-2.61979130E - 06
	-0.0449302706	0.1144916285	0.0733954587	-7.17750570E - 06
	0.1675935311	0.1774161896	0.3459538936	-1.27733934E - 05
	-0.0274959004	0.4539962587	0.4273210007	-5.20897654E - 06



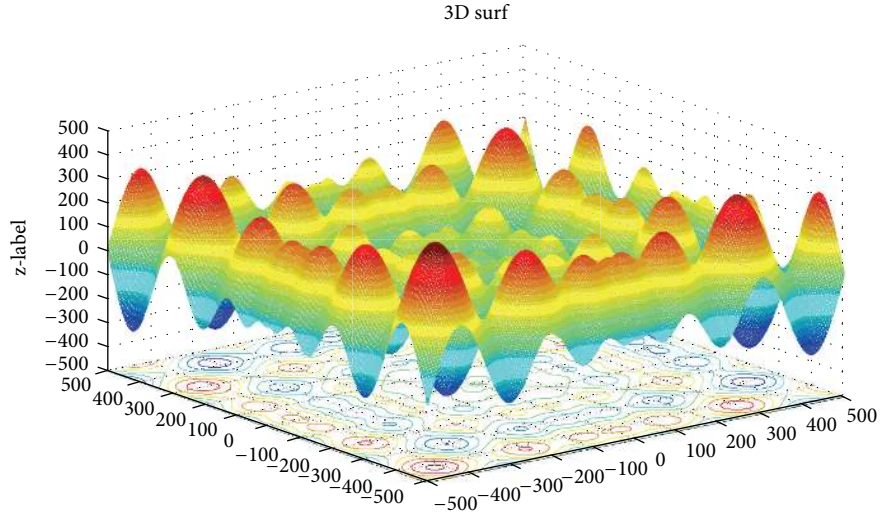


FIGURE 1: The landscape of Schwefel's function.

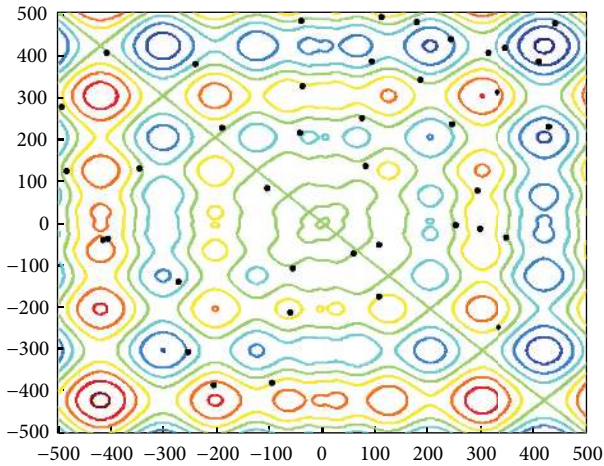


FIGURE 2: The locations of 40 bats in the initial phase.

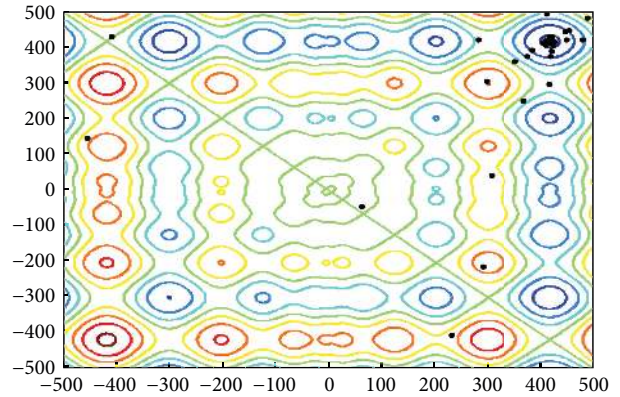


FIGURE 3: The locations of 40 bats after six iterations.

## 5. Conclusions

Aiming at the phenomenon of slow convergence rate and low accuracy of bat algorithm, we put forward an improved bat algorithm with differential operator and Lévy flights trajectory (DLBA) based on the basic framework of bat algorithm (BA), the purpose is to improve the convergence rate and precision of bat algorithm. In this paper, we define the frequency fluctuations up and down when the bat tracking prey, which influence the bats' location problem; it is more graphic to simulate the bat's behavior. Moreover, it can make the algorithm effectively jump out of the local optimum to add Lévy flights and differential operator, the main reason is because Lévy flight has prominent properties in the previously mentioned and the differential operator can guide bats find better solutions, sequentially, increase the convergent speed.

In addition, bats' position variation is influenced by the pulse emission rate and loudness as well. Firstly, pulse emission rate causes update of position, and more and more new position can be explored, consequently, increasing the diversity of population. Secondly, loudness is designed to strengthen local search and to guide bats find better solutions.

In this paper, we tested 14 typical benchmark functions and applied them to solve nonlinear equations, the simulation results not only showed that the proposed algorithm is feasible and effective, which is more robust, but also demonstrated the superior approximation capabilities in high-dimensional space. This is not surprising as the aim of developing the new algorithm was to try to use the advantages of existing algorithms and other interesting feature inspired by the fantastic behavior of echolocation of microbats. Numerically speaking, these can be translated into two crucial characteristics of the modern metaheuristics: intensification and diversification. Intensification intends to search around the current best solutions and select the best candidates or solutions, while diversification makes sure that the algorithm can explore the search space efficiently.

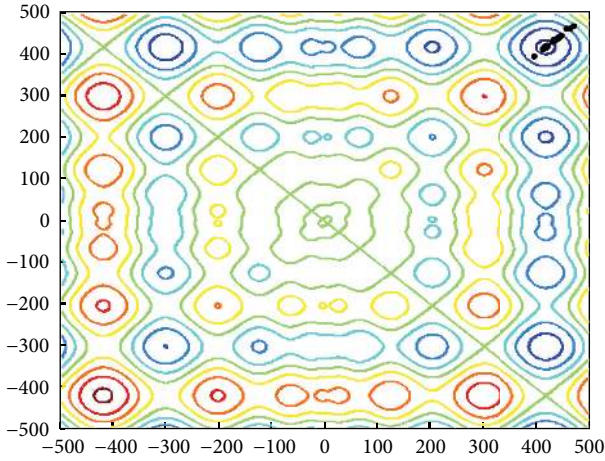


FIGURE 4: The locations of 40 bats after fifteen iterations.

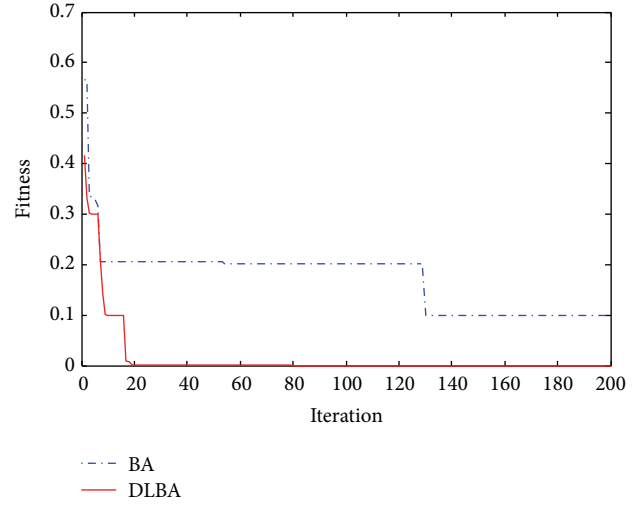


FIGURE 7: Convergence curves for the function  $f_7$ .

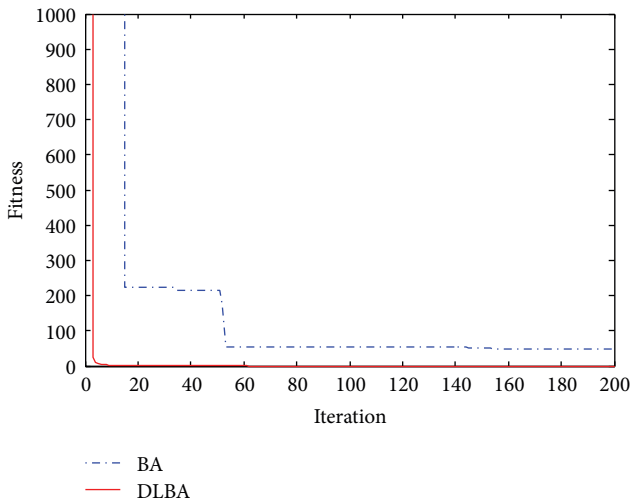


FIGURE 5: Convergence curves for the function  $f_2$ .

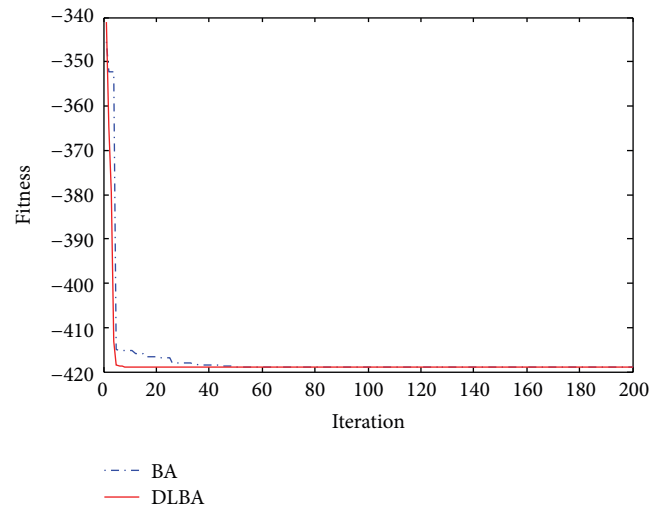


FIGURE 8: Convergence curves for the function  $f_{11}$ .

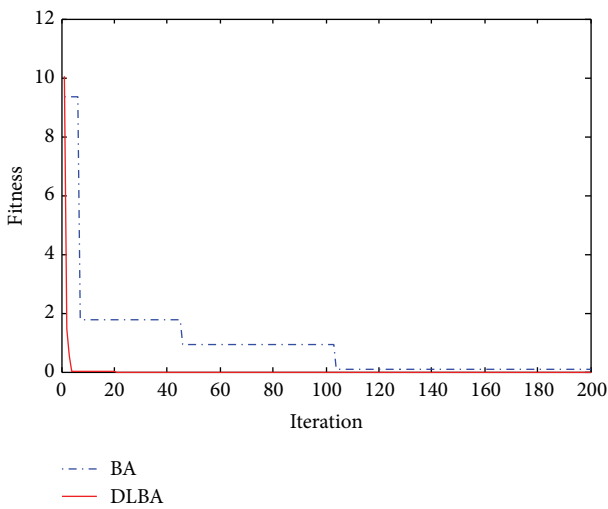


FIGURE 6: Convergence curves for the function  $f_4$ .

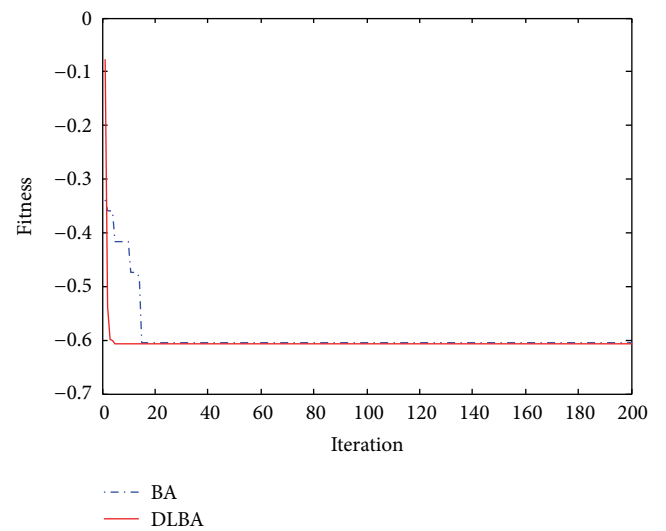


FIGURE 9: Convergence curves for the function  $f_{13}$ .

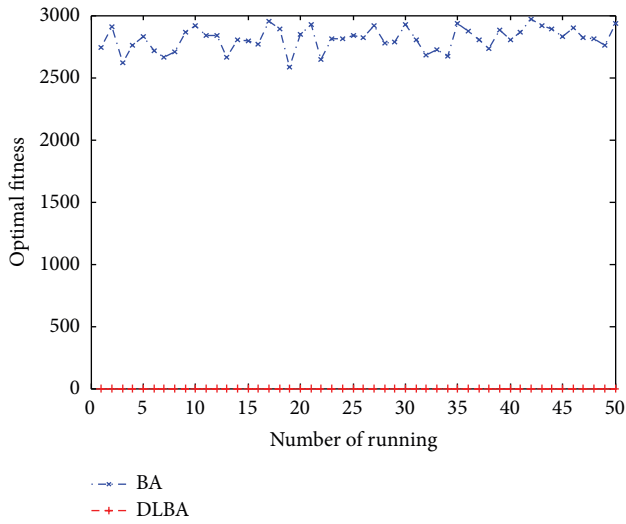


FIGURE 10: Distribution of optimal fitness for  $f_6$  ( $D = 128$ ).

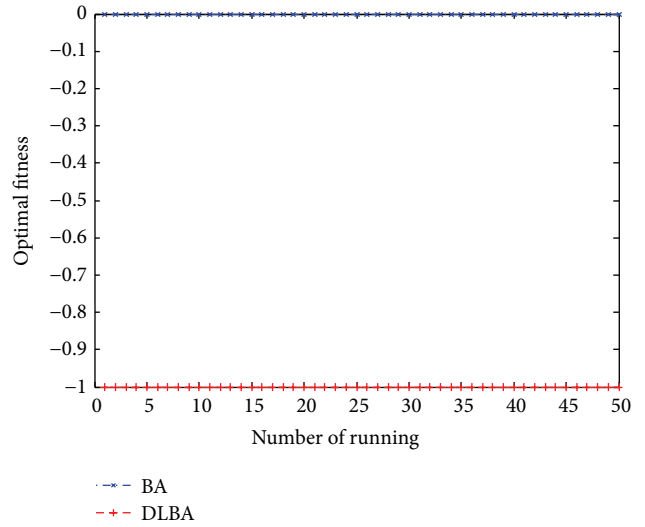


FIGURE 13: Distribution of optimal fitness for  $f_{14}$  ( $D = 512$ ).

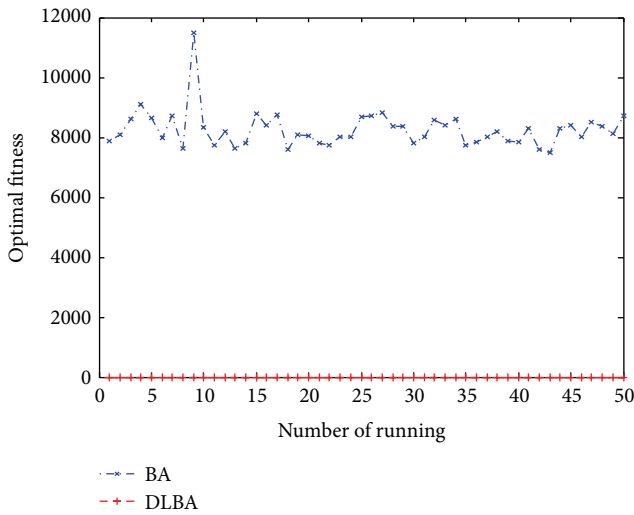


FIGURE 11: Distribution of optimal fitness for  $f_9$  ( $D = 256$ ).

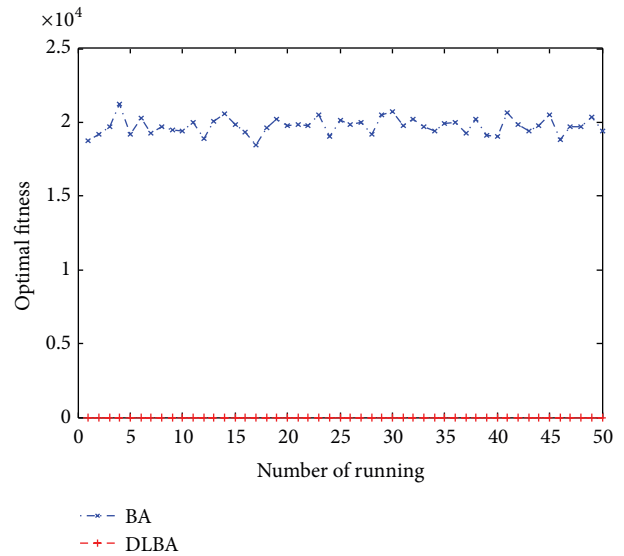


FIGURE 14: Distribution of optimal fitness for  $f_1$  ( $D = 1024$ ).

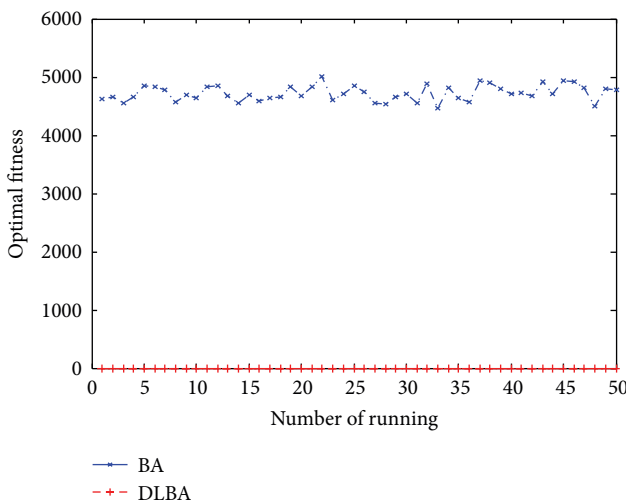


FIGURE 12: Distribution of optimal fitness for  $f_8$  ( $D = 320$ ).

This potentially powerful optimization strategy can easily be extended to study multiobjective optimization applications with various constraints, even to NP-hard problems. Further studies can focus on the sensitivity and parameter studies and their possible relationships with the convergence rate of the algorithm.

### Acknowledgments

This work is supported by the National Science Foundation of China under Grant no. 61165015, Key Project of Guangxi Science Foundation under Grant no. 2012GXNSFDA053028, Key Project of Guangxi High School Science Foundation under Grant no. 20121ZD008, and the Fund by Open Research Fund Program of Key Lab of Intelligent Perception

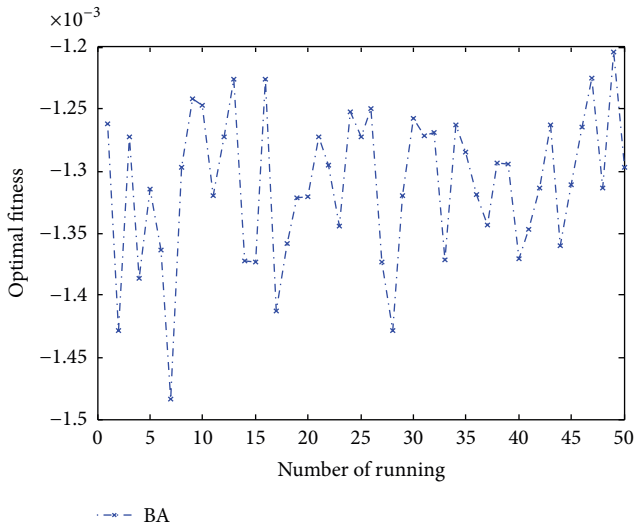


FIGURE 15: Details of BA line in Figure 12.

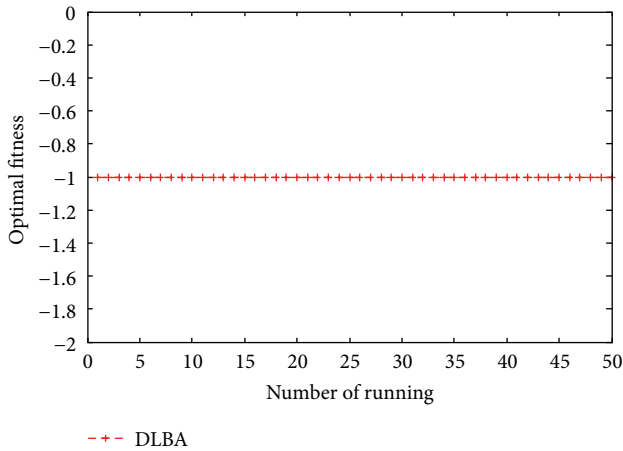


FIGURE 16: Details of DLBA line in Figure 12.

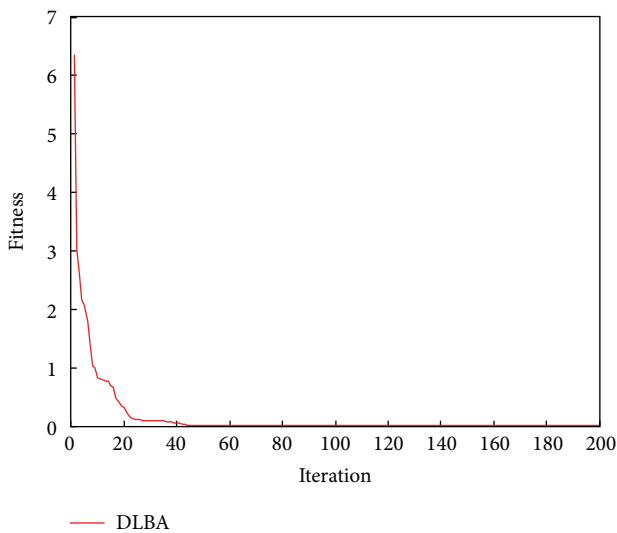


FIGURE 17: Convergence curves for DLBA.

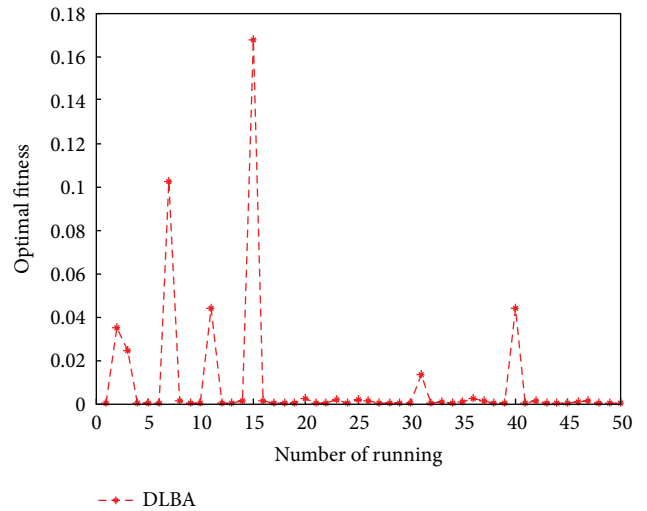


FIGURE 18: Distribution of optimal fitness for DLBA.

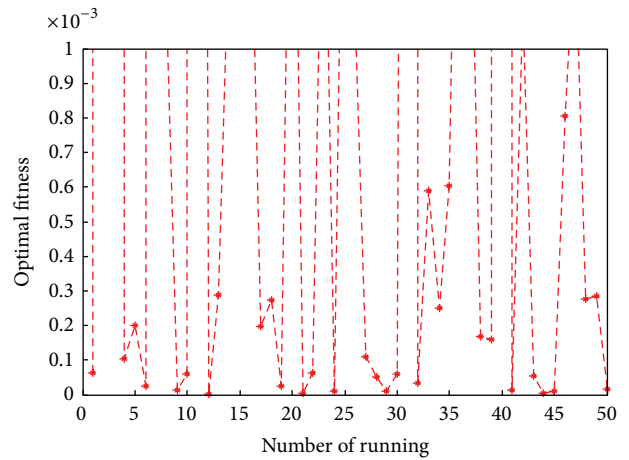


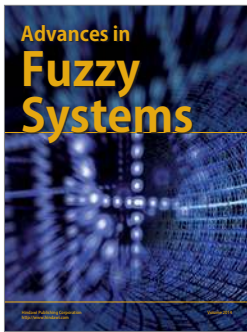
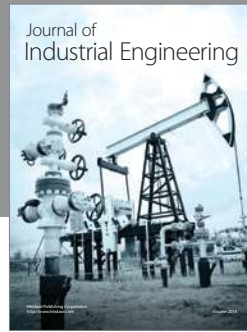
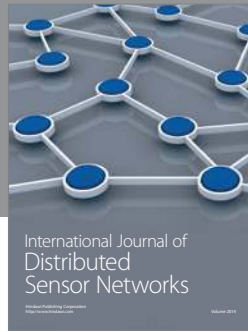
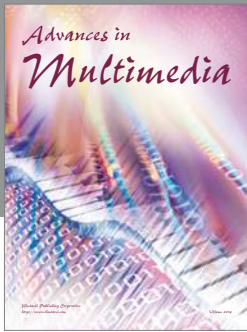
FIGURE 19: Distribution of optimal fitness less than 0.001 in Figure 18.

and Image Understanding of Ministry of Education of China under Grant no. IPIU01201100.

### References

- [1] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of IEEE International Conference on Neural Networks (ICNN '95)*, vol. 4, pp. 1942–1948, December 1995.
- [2] X.-S. Yang, *Nature-Inspired Metaheuristic Algorithms*, Luniver Press, Bristol, UK, 2nd edition, 2010.
- [3] X.-S. Yang, "Firefly algorithms for multimodal optimization," in *Proceedings of the 5th International Conference on Stochastic Algorithms: Foundations and Applications (SAGA '09)*, pp. 169–178, 2009.
- [4] B. Alatas, "ACROA: artificial chemical reaction optimization algorithm for global optimization," *Expert Systems with Applications*, vol. 38, no. 10, pp. 13170–13180, 2011.

- [5] K. N. Krishnanand and D. Ghose, "Glowworm warm optimization for searching higher dimensional spaces," *Studies in Computational Intelligence*, vol. 248, pp. 61–75, 2009.
- [6] A. R. Mehrabian and C. Lucas, "A novel numerical optimization algorithm inspired from weed colonization," *Ecological Informatics*, vol. 1, no. 4, pp. 355–366, 2006.
- [7] K. Price, R. Storn, and J. Lampinen, *Differential Evolution—A Practical Approach to Global Optimization*, Springer, Berlin, Germany, 2005.
- [8] <http://www.icsi.berkeley.edu/~storn/>.
- [9] R. Storn and K. Price, "Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.
- [10] X.-S. Yang, "A new metaheuristic Bat-inspired Algorithm," *Studies in Computational Intelligence*, vol. 284, pp. 65–74, 2010.
- [11] A. Mucherino and O. Seref, "Monkey search: a novel metaheuristic search for global optimization," in *Proceedings of the Conference on Data Mining, Systems Analysis, and Optimization in Biomedicine*, vol. 953, pp. 162–173, March 2007.
- [12] K. M. Passino, "Biomimicry of bacterial foraging for distributed optimization and control," *IEEE Control Systems Magazine*, vol. 22, no. 3, pp. 52–67, 2002.
- [13] G. R. Reynolds, "Cultural algorithms: theory and application," in *New Ideas in Optimization*, D. Corne, M. Dorigo, and F. Glover, Eds., pp. 367–378, McGraw-Hill, New York, NY, USA, 1999.
- [14] B. Alatas, "Chaotic harmony search algorithms," *Applied Mathematics and Computation*, vol. 216, no. 9, pp. 2687–2699, 2010.
- [15] R. Oftadeh, M. J. Mahjoob, and M. Shariatpanahi, "A novel meta-heuristic optimization algorithm inspired by group hunting of animals: hunting search," *Computers and Mathematics with Applications*, vol. 60, no. 7, pp. 2087–2098, 2010.
- [16] P. W. Tsai, J. S. Pan, B. Y. Liao, M. J. Tsai, and V. Istanda, "Bat algorithm inspired algorithm for solving numerical optimization problems," *Applied Mechanics and Materials*, vol. 148–149, pp. 134–137, 2011.
- [17] X.-S. Yang, "Bat algorithm for multi-objective optimization," *International Journal of Bio-Inspired Computation Bio-Inspired Computation (IJBIC)*, vol. 3, no. 5, pp. 267–274, 2011.
- [18] T. C. Bora, L. S. Coelho, and L. Lebensztajn, "Bat-Inspired optimization approach for the brushless DC wheel motor problem," *IEEE Transactions on Magnetics*, vol. 48, no. 2, pp. 947–950, 2012.
- [19] B. B. Mandelbrot, *The Fractal Geometry of Nature*, W.H. Freeman, New York, NY, USA, 1983.
- [20] V. A. Chechkin, R. Metzler, J. Klafter, and V. Y. Gonchar, "Introduction to the theory of Lévy flights," in *Anomalous Transport: Foundations and Applications*, pp. 129–162, Wiley-VCH, Cambridge, UK, 2008.
- [21] A. M. Reynolds and M. A. Frye, "Free-flight odor tracking in *Drosophila* is consistent with an optimal intermittent scale-free search," *PLoS ONE*, vol. 2, no. 4, p. e354, 2007.
- [22] C. Brown, L. S. Liebovitch, and R. Glendon, "Lévy flights in Dobe Ju/'hoansi foraging patterns," *Human Ecology*, vol. 35, no. 1, pp. 129–138, 2007.
- [23] P. Barthelemy, J. Bertolotti, and D. S. Wiersma, "A Lévy flight for light," *Nature*, vol. 453, no. 7194, pp. 495–498, 2008.
- [24] N. Mercadier, W. Guerin, M. Chevrollier, and R. Kaiser, "Lévy flights of photons in hot atomic vapours," *Nature Physics*, vol. 5, no. 8, pp. 602–605, 2009.
- [25] C. Grosan and A. Abraham, "A new approach for solving nonlinear equations systems," *IEEE Transactions on Systems, Man, and Cybernetics A*, vol. 38, no. 3, pp. 698–714, 2008.



# Hindawi

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

