Bat Algorithm Inspired Algorithm for Solving Numerical Optimization Problems

Article *in* Applied Mechanics and Materials · December 2011

DOI: 10.4028/www.scientific.net/AMM.148-149.134

Applied Mechanics and Materials Vols. 148-149 (2012) pp 134-137
Online available since 2011/Dec/22 at www.scientific.net

doi:10.4028/www.scientific.net/AMM.148-149.134

Bat Algorithm Inspired Algorithm for Solving Numerical Optimization Problems

Pei-Wei Tsai [a], Jeng-Shyang Pan [b], Bin-Yih Liao [c], Ming-Jer Tsai [d] and Vaci･Istanda [e]

Department of Electronic Engineering, National Kaohsiung University of Applied Sciences, Kaohsiung City, 80778, Taiwan (R.O.C.)

[a]pwtsai@bit.kuas.edu.tw, [b]jspan@cc.kuas.edu.tw, [c]byliao@cc.kuas.edu.tw, [d]zero61118@hotmail.com, [e]biungsu@yahoo.com.tw

Keywords: Evolved Bat Algorithm; Bat Algorithm; Swarm Intelligence; Optimization; Bio-inspired Computing.

Abstract. Inspired by Bat Algorithm, a novel algorithm, which is called Evolved Bat Algorithm (EBA), for solving the numerical optimization problem is proposed based on the framework of the original bat algorithm. By reanalyzing the behavior of bats and considering the general characteristics of whole species of bat, we redefine the corresponding operation to the bats' behaviors. EBA is a new method in the branch of swarm intelligence for solving numerical optimization problems. In order to analyze the improvement on the accuracy of finding the near best solution and the reduction in the computational cost, three well-known and commonly used test functions in the field of swarm intelligence for testing the accuracy and the performance of the algorithm, are used in the experiments. The experimental results indicate that our proposed method improves at least 99.42% on the accuracy of finding the near best solution and reduces 6.07% in average, simultaneously, on the computational time than the original bat algorithm.

Introduction

In more than three decades, swarm intelligence has been successfully used to solve optimization problems in the engineering, the financial, and the management fields. For example, particle swarm optimization (PSO) techniques have successfully been used to construct the portfolios of stock [1] and to construct parameters in neural network systems [4]; ant colony optimization (ACO) techniques have successfully been used to solve the traveling salesman problem (TSP) [3] and the routing problem of networks [5]; cat swarm optimization (CSO) [2] techniques have successfully been used to discover proper positions for information hiding [6-7].

In 2010, Yang proposed a new optimization algorithm, namely, Bat Algorithm (BA), based on swarm intelligence and the inspiration form observing the bats [8]. Although the original BA presents superior results in the experiments than PSO, we notice that the performance and the accuracy of the original BA still have the capacity to present better. In this paper, the behaviors and the characteristics of the bat are reanalyzed and redefined with the new formulae to present the Evolved Bat Algorithm (EBA). According to the experimental results, our proposed EBA presents more accurate result in finding near best solutions.

Bat Algorithm

Yang proposed BA [8] by observing the behaviors and the characteristics of the microbat in 2010. Three major characteristics of the microbat are employed to construct the basic structure of BA. The used approximate and the idealized rules in Yang's method are listed as follows:

1. Most of the species of the bat utilize the echolocation to detect their prey, but not all species of the bat do the same thing. However, the microbat is a famous example of extensively using the echolocation. Hence, the first characteristic is the echolocation behavior.

2. The second characteristic is the frequency that the microbat sends a fixed frequency f_{min} with a variable wavelength λ and the loudness A_0 to search for prey.
3. There are many ways to adjust the loudness. For simplicity, the loudness is assumed to be varied from a positive large A_0 to a minimum constant value, which is denoted by A_{min}.
   In Yang's method, the movement of the virtual bat is simulated by Eq. (1) – Eq. (3):

$$f_i = f_{min} + (f_{max} - f_{min}) \cdot \beta \tag{1}$$

$$v_i^t = v_i^{t-1} + (x_i^t - x_{best}) \cdot f_i \tag{2}$$

$$x_i^t = x_i^{t-1} + v_i^t \tag{3}$$

where f is the frequency used by the bat seeking for its prey, the suffixes, min and max, represent the minimum and maximum value, respectively. x_i denotes the location of the i^{th} bat in the solution space, v_i represents the velocity of the bat, t indicates the current iteration, β is a random vector, which is drawn from a uniform distribution, and $\beta \in [0, 1]$, and x_{best} indicates the global near best solution found so far over the whole population.

In addition, the rate of the pulse emission from the bat is also taken to be one of the roles in the process. The pulse emission rate is denoted by the symbol r_i, and $r_i \in [0, 1]$, where the suffix i indicates the i^{th} bat. In every iteration, a random number is generated and is compared with r_i. If the random number is greater than r_i, a local search strategy, namely, random walk, is detonated. A new solution for the bat is generated by Eq. (4):

$$x_{new} = x_{old} + \varepsilon A^t \tag{4}$$

where ε is a random number and $\varepsilon \in [-1, 1]$, and A^t represents the average loudness of all bats at the current time step. After updating the positions of the bats, the loudness A_i and the pulse emission rate r_i are also updated only when the global near best solution is updated and the random generated number is smaller than A_i. The update of A_i and r_i are operated by Eq. (5) and Eq. (6):

$$A_i^{t+1} = \alpha \cdot A_i^t \tag{5}$$

$$r_i^{t+1} = r_i^0[1 - e^{-\gamma t}] \tag{6}$$

where α and γ are constants. In Yang's experiments, $\alpha = \gamma = 0.9$ is used for the simplicity.

Our Proposed Method

Inspired by Yang's method, we propose the Evolved Bat Algorithm (EBA) based on utilizing the basic structure of BA and re-estimating the characters used in the original BA. In EBA, not only the movement of the bat is quite different from the original BA, but also the random walk process.

The sound speed spreads in the air is *340* meter per second. In the active sonar system, the distance between the sound wave source and the target, which bounds the wave back, is defined by Eq. (7):

$$D = \frac{v \cdot \Delta T}{2} \tag{7}$$

where D denotes the distance, v is the sound speed, and ΔT means the time difference between sending the sound wave and receiving the echo. We use the sound speed in the air to be the value of v. Furthermore, we replace the unit of v from meter per second to kilometer per second. Thus, Eq. (7) can be reformed into Eq. (8):

$$D = 170 \cdot \Delta T \left({}^{m}/_{second} \right) = 0.17 \cdot \Delta T \left({}^{km}/_{second} \right) \tag{8}$$

In our experiments, we use a random number in the range of $[-1\,,1]$ to denote ΔT. The negative part of ΔT comes from the moving direction in the coordinate. ΔT is given with a negative value when the transmission direction of the sound wave is opposite to the axis of the coordinate. The movement of the bat in EBA is defined by Eq. (9):

$$x_i^t = x_i^{t-1} + D \tag{9}$$

If a bat moves into the random walk process, its location will be updated by Eq. (10):

$$x_i^{t_R} = \beta \cdot (x_{best} - x_i^t) \tag{10}$$

where β is a random number in the range $\beta \in [0\,,1]$ and $x_i^{t_R}$ indicates the new location of the bat after the random walk process. The processes of EBA can be depicted as follows:

Step 1. Initialization: Randomly spread the bats into the solution space.

Step 2. Move the bats by Eq. (8) and Eq. (9). Generate a random number. If it is greater than the fixed pulse emission rate, move the bat by the random walk process, which is defined by Eq. (10).

Step 3. Evaluate the fitness of the bats and update the global near best solution.

Step 4. Check the termination condition to decide whether go back to step 2 or terminate the program and output the near best result.

Simulation and Experimental Results

To analyze the accuracy and the computational speed of our proposed method, three test functions, which are listed in Eq. (11) – Eq. (13), are used in the experiments. The results are compared with the original BA. The optimization goal for all test functions is to minimize the outcome.

$$f_1(x) = \sum_{i=1}^{M-1}[100 \cdot (x_{i+1} - x_i^2)^2 + (x_i - 1)^2] \tag{11}$$

$$f_2(x) = \frac{1}{4000}\sum_{i=1}^{M} x_i^2 - \prod_{i=1}^{M} cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \tag{12}$$

$$f_3(x) = \sum_{i=1}^{M}[x_i^2 - 10 \cdot cos(x\pi x_i) + 10] \tag{13}$$

The experiments are taken on the personal computer with an Intel Core-i7 950 3.0 GHz CPU, 12GB RAM, Windows 7 64bits OS, with Matlab Version 7.6.0.324 (R2008a). The total population size is set to *20* and the dimension of the solution space is set to *30* (M = *30*). Each test function contains the full iterations is repeated by *25* runs with different random seeds. The final result is obtained by taking the average of the outcomes from all runs. The initial range and the total iteration number for all test functions are listed in Table 1. The parameter setting for BA and our proposed method are listed in Table 2. The outcome and the computational time are listed in Table 3.

Table 1. The initial range and the total iteration number of the test functions.

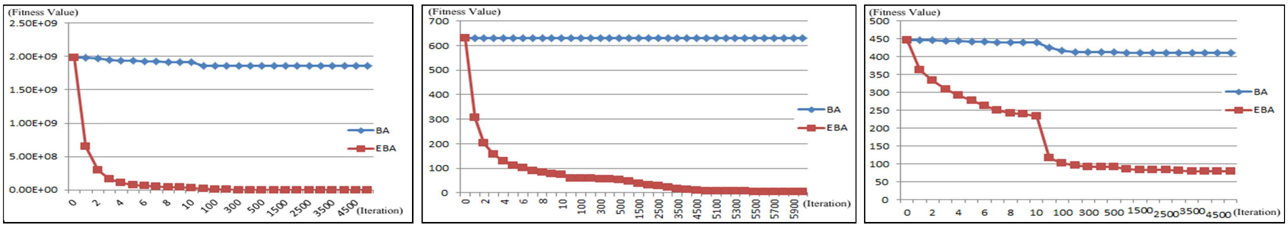
Function	Initial Range	Total Iteration
$f_1(x)$	$x_i \in [-50\,,50]$	5,000
$f_2(x)$	$x_i \in [-600\,,600]$	6,000
$f_3(x)$	$x_i \in [-5.12\,,5.12]$	5,000

Table 2. The parameter setting for BA and EBA.

Bat Algorithm [8]		Our Proposed Method	
Initial r_i^0	$[0\,,1]$	Fixed r	0.5
Initial A_i^0	$[1\,,2]$	ΔT	$[-1\,,1]$
$[f_{min}\,,f_{max}]$	$[0\,,100]$		
ε	$[-1\,,1]$		
α and γ	0.9		

Table 3. The average outcomes and the average computational time.

	Bat Algorithm [8]		Our Proposed Method	
	Average Outcome	Time (sec.)	Average Outcome	Time (sec.)
$f_1(x)$	1.864×10^9	2.632	2.266×10^2	2.589
$f_2(x)$	6.306×10^2	4.392	3.648×10^0	3.977
$f_3(x)$	4.104×10^2	2.976	7.949×10^1	2.764



(a) (b) (c)

Figure 1. Experimental result of the test functions: (a)-(c) represent test function 1 to 3, respectively.

According to the experimental results in Fig. 1 obtained from all test functions, EBA presents higher accuracy than the original BA on minimizing the outcome as the optimization goal. Moreover, the computational time of EBA is shorter than the original BA due to the framework and the processes are simpler than BA. The experimental results indicate that EBA improves the accuracy at least 99.42% and the computational time is reduced about 6.07% in average than the original BA.

Conclusions

In this paper, we propose a newly improved BA, which is called Evolved Bat Algorithm (EBA), by reanalyzing the characteristics of the bat and redefining the corresponding operations based on the basic framework of Bat Algorithm (BA). The experimental results indicate that our proposed EBA at least improves 99.42% and reduces the computational time about 6.07% in average. In the future work, more characters of bats will be involved in the EBA process to improve the accuracy.

References

[1] J.-F. Chang and S.-W. Hsu: The Construction of Stock's Portfolios by Using Particle Swarm Optimization. Proceedings of the 2nd International Conference on Innovative Computing, Information and Control, (2007).

[2] S.-C. Chu and P.-W. Tsai: Computational Intelligence Based on the Behavior of Cats, International Journal of Innovative Computing, Information and Control 3(1), 163 (2006).

[3] M. Dorigo and L. M. Gambardella: Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem, IEEE Transactions on Evolutionary Computation 1(1), 53 (1997).

[4] C. J. Lin, C. H. Chen, and C. T. Lin: A Hybrid of Cooperative Particle Swarm Optimization and Cultural Algorithm for Neural Fuzzy Networks and Its Prediction Applications, IEEE Transactions on Systems, Man, and Cybernetics-Part C: Applications and Reviews 39(1), 55 (2009).

[5] C. P. Pinto, A. Nägele, M. Dejori, T. A. Runkler, and J. M. C. Sousa: Using a Local Discovery Ant Algorithm for Bayesian Network Structure Learning, IEEE Transactions on Evolutionary Computation 13(4), 767 (2009).

[6] Z.-H. Wang, C.-C. Chang, and M.-C. Li: Optimizing Least-significant-bit Substitution Using Cat Swarm Optimization Strategy, Information Sciences, In Press (2010).

[7] Z.-H. Wang, C.-Y. Chang, and P.-W. Tsai: A Reversible Data Embedding Scheme Based on Hash Functions for VQ Index Tables. Proceedings of the 4th International Conference on Genetic and Evolutionary Computing, (2010).

[8] X.-S. Yang: A New Metaheuristic Bat-Inspired Algorithm, in Nature Inspired Cooperative Strategies for Optimization (NICSO 2010), Studies in Computational Intelligence 284, Edited J. R. González, D. A. Pelta, C. Cruz, G. Terrazas, and N. Krasnogor, Springer-Verlag, Berlin Heidelberg (2010), pp. 65-74.

Mechanical Engineering, Materials and Energy
10.4028/www.scientific.net/AMM.148-149

Bat Algorithm Inspired Algorithm for Solving Numerical Optimization Problems
10.4028/www.scientific.net/AMM.148-149.134