International Journal of
Current Multidisciplinary Studies
Available Online at http://www.journalijcms.com
Vol. 2, Issue,10, pp.479-482, October, 2016
IJCMS

RESEARCH ARTICLE

A STUDY OF BAT ALGORITHM AND ITS VARIANTS

RashmiAmardeep[1] and ThippeSwamy K[2]

[1]Sri Siddhartha Academy of Higher Education, Tumkur, India
[2]Visvesvaraya Technological University, PG Regional Center Mysore, India

ARTICLE INFO

Received 2nd, July, 2016,
Received in revised form 5 th,
August, 2016, Accepted 24th, September, 2016,
Published online 28th, October, 2016

Keywords:

Bat algorithm, Bio-inspired Algorithm, MOBA, Compact Bat Algorithm,

ABSTRACT

In solving global optimization problems Meta-heuristic algorithms are becoming very powerful. Over past few decades many Meta-heuristic algorithms have been developed. This paper intends to provide one of the nature inspired new metaheuristic optimization algorithm, called Bat Algorithm (BA) and its variants. The result of study shows that some variants of BAs can clearly outperform the standard BA.

INTRODUCTION

Modern optimization algorithms are often nature-inspired, typically based on swarm intelligence. The ways for inspiration are diverse and consequently algorithms can be many different types. However, all these algorithms tend to use some specific characteristics for formulating the key updating formulae [1].Meta heuristic algorithms has expanded significantly from past 3 years and this has lead to the development of many heuristic algorithms. Meta-heuristic techniques are well-known global optimization methods that have been successfully applied in many real-world and complex optimization problems [6, 7]

Bat Algorithm [4] is one of the heuristic optimization algorithms.BA was inspired by the echolocation behavior of microbats, with varying pulse rates of emission and loudness. The Bat algorithm was developed by Xin-She Yang in 2010. It uses the automatic zooming which try to balance exploration and exploitation during the search process by mimicking the variations of pulse emission rates and loudness of bats when searching for prey. BA has been found to be very efficient.

Most bats use short, frequency-modulated signals to sweep through about an octave, and each pulse lasts a few thousandths of a second (up to about 8 ms to 10 ms) in the frequency range of 25kHz to 150kHz.Typically, microbats can emit about 200 pulses per second, and the rate of pulse emission can be spread up to about 200 pulses per second when homing on their prey. Since the speed of sound in air is about v=340 m/s, the wavelength of the ultrasonic sound bursts with a constant frequency f is given by v/f which is in the range of 2 mm to 14mm for the typical frequency range from 25kHz to 150kHz.

Fundamental of Bat Algorithm

Some bats have evolved a highly sophisticated sense of hearing. They emit sounds that bounce off of objects in their path, sending echoes back to the bats. From these echoes, the bats can determine the size, distance, travelling speed and the texture of objects, all in a split second.

Various bat-inspired algorithms can be developed by using echolocation characteristics of microbats. Basic bat algorithm developed by Xin-She Yang in 2010 is as follows:

1. All bats use echolocation to sense distance of prey.
2. Bats fly randomly with velocity V_i at position X_i with a frequency f_{min}, varying wavelength 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 Є [0, 1], depending on the proximity of their target.
3. Although the loudness can vary in many ways, Yang[3]assume that the loudness varies from a large (positive) A_0 to a minimum constants value A_{min}.

The Variants of Bat Algorithm

Following are some of the bat algorithm variants:

Multi-Objective Bat Algorithm (MOBA)

*✉ ***Corresponding author:*** **RashmiAmardeep**
Sri Siddhartha Academy of Higher Education, Tumkur, India

Yang(2011)[11]extended BA to deal with multiobjective optimization, which demonstrates its effectiveness for solving few design benchmark's in structural engineering.

The Pareto front(PF) of a multiobjective can be defined as the set of non-dominated solutions so that
PF={s Є S l Ǝ s Є S; ś s}

Algorithm

Objective function $f_1(x), f_2(x), \ldots f_k(x)$, $x=(x_1,x_2,\ldots x_d)t$
Initialize the bat population x_i (i=1,2,..n) and V_i
For j=1 to N(points on Pareto fronts)
Generate K weights $w_k >= 0$ so that $\sum^K w_k = 1$
From a single objective $f = \sum^K w_k f_k$
While (t<max number of iterations)
Generate new solutions and update f and x of BA
if ($rand > r_i$)
Random walk around a selected best solution
endif
Generate a new solution by flying randomly
if ($rand < A_i$ & $f(x_i) < f(x_*)$)
Accept the new solutions, and increase r_i & reduce Ai
end if
Rank the bats and find the current best x_*
end while
Record x_* as a non-dominated solution
end
Postprocess results and visualization

Multiobjective design of a welded beam is a classical benchmark which has been solved by many Researchers. The simulations for these benchmarks and functions suggest that MOBA is a very efficient algorithm for multiobjective optimization. It can deal with highly nonlinear problems with complex constraints and diverse Pareto optimal sets. The proposed MOBA has been tested against a subset of well-chosen test functions, and then been applied to solve design optimization benchmarks in structural engineering. Results suggest that MOBA is an efficient multiobjective optimizer.

Bat Algorithm Embedded with FLANN (BAT -FLANN)

Sashikala Mishra[12] proposed a model for classification using bat algorithm to update the weights of a Functional Link Artificial Neural Network (FLANN) classifier.

The following are the changes made from standard BA:
The frequency f_k of bat B_k can be achieved by

$$\mathbf{F_k = c_1 \sum (D_{ki})/m)} \qquad \text{---(1)}$$

Where $c1$ is the pulse rate used to control the frequency fk of bat Bk, and when it reaches near or far from the object, the value of $c1$ is auto adjusted in each iteration by (5).

Distance S of the object z from bat B_k

$$\mathbf{S_{objectz} = f_k * D_k * wt} \qquad \text{---(2)}$$

Update position of bat

calculating the error by (3), the bat position can be changed by (4).

$$\mathbf{E_k = S_{objectz} - 1} \qquad \text{---(3)}$$

$$\mathbf{P_k = P_k + E_k} \qquad \text{----(4)}$$

When bat starts flying it assumes that the position is initialized to zero. Its position keeps on changing when it reaches nearer to the object. As closer it moves to the pray, error Ek and position Pk reduces to zero.

Update frequency f and weight wt after change of position of bat

$$\boldsymbol{C_1 = f_k + C_2 * E^2_k * P_k} \qquad \text{---(5)}$$

$$\boldsymbol{Wt = wt + 2 * \mu * E_k} \qquad \text{---(6)}$$

Algorithm

Read normalize data set *X of which*
matrix *D* of size *n *m* for training and rest 20% data set for testing.
Assign wt=random(m,T), μ=0.2, c_1=0.6 and c_2=0.0011
Where *m* is the number of column and *n* is the number of row in *X*.
T is the number of class labels.
No.Of Iteration is user defined
for *i*=1:NoOfIteration
delwt = *wt*
for *j*=1: *n*
read D_j
compute each *batj* frequency as f_j
$f_j = c_1 * \text{Mean}(D_j)$
compute class(object) distance from *batj* as $S_{object\ j}$
$S_{object\ j} = fj * D_j * delwt$
compute for each class
$E_j = S_{object\ j} - 1$
update $wt = wt - 2 * \mu * E_i$
compute the new position P_j and change pulse rate controller c1 of *Bat*j
if ($E_j < E_j - 1$)
$P_j = P_j + E_j$
$c_1 = fj + c_2 * P_j * E_j$

end
end
calculate the error and update *wt*
err(i)=Mean(E)
wt = wt- delwt
end
Postprocess the result and visualize.

The bat algorithm successfully formulated and is used to update the weight of the FLANN classifier. Wide knowledge of bats echolocation signals and their specific features results in a good accuracy in FLANN compared with other methods.

Directed Artificial Bat Algorithm (DABA)

Directed Artificial Bat Algorithm proposed by Amr Rekaby (2013) and choose TSP as the standard test case to evaluate the algorithms efficiency [5].DABA algorithm uses a generation of bats to find the approximate solution. Every bat has an initial position (represents a solution) and direction (search area scope). These initial values are defined in first generation initiation step. This position reflects a solution in the search space while the direction is the search scope. This scope is the area in the search space that the bat will search in it. The direction of the bat is a logical concept in DABA, it can be theoretically presented as 90' degree vision. Each bat is looking for a prey (better solution) in its direction scope. By referring to the local search, bat here is doing a local search by looking for the neighborhood solutions but with a directed search scope (does not cover all the surrounding neighbors).This direction (scope) is applied as a set of vectors of searching lines (will be described later). These vectors visit points in their direction. These points actually are available solutions in the search space. The total amount of visited points by each vector is simulating the wavelength in real bat echolocation.The count of vectors themselves is simulating the frequency in the natural echolocation system. As bats do in the real world, the artificial bats in DABA do. The frequency of the waves is increased if the bat finds a prey (good solution in the artificial world), but it is decreased on the other cases. Best solution is obtained by decreasing the wavelength and increasing the frequency.

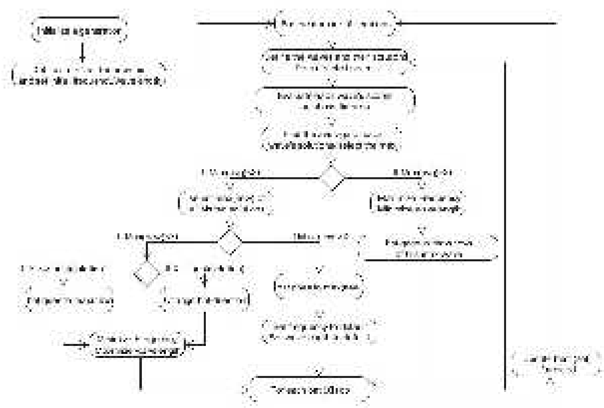

Fig 1 DABA Algorithm

From the results of study this proposed DABA achieves better results than Artificial Bee colony (ABC) with efficiency enhancements between 5% and 10%.

Binary Bat Algorithm (BBA)

Nakamura *et al.*(2012) developed a binary version of bat algorithm to solve classification and feature selection problems. BBA will have artificial bats navigating and hunting in binary search by changing their positions from “0”to “1”.Bat position is then represented by binary vectors using a sigmoid function[4].

$x_i^{\,j}(t)$ denotes the value of decision variable j for bat i at time step t is represented by :

$$x_i^{\,j}(t) = \begin{cases} 1 \text{ if rand} \geq S(v_i^{\,j}) \\ 0 \text{ otherwise} \end{cases}$$

Pseudocode

Initialize the bat position: X_i (i=1,2,…n) = rand(0 or 1) and V_i=0
Define Pulse frequency F_i
Initialize pulse rates r_i and the loudness A_i
While (t<Max number of iterations)
    Adjust frequency and updating velocities
    Calculate transfer function
    Update positions using equation(1)
    If(rand>r_i)
        Select a solution(Gbest)among the best solutions randomly
        Change some of the dimension of position vector with some of the dimensions of Gbest
    e**ndif**
Generate a new solution by flying randomly
    If((rand<A_i &f(x_i)<f(Gbest))
        Accept the new solutions
        Increase r_i and reduce A_i
    endif
Rank the bats and find the current Gbest
endwhile

Chaotic bat algorithm

Lin *et al*(2012)developed this algorithm using Levy flights and chaotic maps to carry out parameter estimation in dynamic biological systems. The methods using chaotic maps to replace random variables are called chaotic optimization [9] (CO). Amir H. Gandomi, Xin-She Yang, the author introduces chaos to the standard BA,and have developed a set of chaotic BA variants[8].

In this algorithms, different chaotic maps replaces the random variables in BA. Due to the ergodicity and mixing properties of chaos, algorithms can potentially carry out iterative search steps at higher speeds than standard stochastic search with standard probability distributions. To achieve such potential, the author use one-dimensional, non-invertible maps(Chebyshev map, Circle map, piecewise maps, etc.) to generate a set of chaotic bat algorithm variants.

The following parameters were modified to obtain variants in CBA:

1.Parameter β,loudness(A),pulse emission rate(r) were modified by chaotic maps(CM) to obtain variants
2. Frequency by :
$\mathbf{F_i = f_{min} + (f_{max} - f_{min})CM_i}$
3. Velocity equation is modified to:
$\mathbf{v^t_i = v^{t-1}{}_i + (x^t_i - x\ )CM_i\, f_i}$

Algorithm of CBA:
Initialization:
Objective function *f(x)*, *x* = *(x₁, ..., x d)T*
Generate initial population of bats xi (*i = 1, 2, ..., n*)
C_0 = a random number
k=1
Tuning of parameters using chaotic maps (β, λ, A or r= C_k)
Movement of Bats:
Generate new solutions by adjusting frequency and updating velocities and locations
if (rand > r)
    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(rand < A & f (xi) < f (x *))
    Accept the new solutions
end if
k = k+1
if(k≤ Maximum Generation)

Tune the parameters and go to movements of Bats
else
Post process the results
endif
Stop

The result of the work by author shows the improvement in reliability of the global optimality, and they also enhance the quality of the results. There is improvement in performance by changing various parameters in different chaotic maps.

Compact Bat Algorithm

CBA [10] is for solution numerical optimization problem which replace population with probability vector. CBA can solve optimization by using less memory usage and gives good performance.

In CBA less memory usage is obtained by simulate the behavior of population based algorithms by employing perturbation vector (PV) as probabilistic representation instead of population of solution. This reduces the number of parameters stored in the memory. Thus, a run of these algorithms requires much less capacious memory devices compared to their correspondent population based structure.

The distribution of the individual in the hypothetical swarms must be described by a probability density function (PDF) defined on the normalized interval from -1 to +1. The distribution of each bat of swarms could be assumed as Gaussian PDF with mean μ and standard deviation σ. A minimization problem is considered in an m-dimensional hyper rectangle in normalization of two truncated Gaussian cures. Without loss of generality the parameters assumed to be normalized so that each search interval is [-1, +1]. Therefore PV is vector of m*2 matrix specifying the two parameters of the PDF of each design variable being defined as

$PV^t = [\mu^t, \sigma^t]$

Where μ and σ are mean & standard deviation values of Gaussian (PDF) truncated within the interval [-1, +1] respectively.

J-S Pan *et al*. (eds) in his experiment results shows that the curves of CBA are faster in convergence. In comparison of CBA and BA shows that CBA has smaller number of memory variables than BA. Execution time in CBA is smaller than BA.

The implantation of CBA for optimization algorithms could have important significance for the development of embedded devices with small size, low price and being suitable for trend of ubiquitous computing today.

Reason For Bat Algorithm Popularity

BA has a capability of automatically zooming into a region where promising solutions have been found. Zooming facility provides automatic switch from explorative moves to local intensive exploitation. BA has a quick convergence rate, atleast at early stages of the iterations, compared with other algorithm. BA uses parameter control, which can vary the values of parameter (A and r) as the iterations proceed. This also provides a way to automatically switch from exploration to exploitation when the optimal solution approaching. Solve a wide range of problems and highly nonlinear problems efficiently.BA works well with complicated problems and gives promising optimal solutions.

CONCLUSION

The accuracy of finding the near best solution and the reduction in the computational cost, in the field of swarm intelligence for testing the accuracy and the performance of the algorithm many variants of BAT algorithm has emerged. Studying various literatures we conclude that bat algorithm is easy to implement and much superior algorithm in terms of accuracy and efficiency in comparison with and other meta-heuristic algorithm.

References

1. Bat algorithm: literature review and applications Xin -she yang *Int. J. Bio-Inspired Computation*, Vol. 5, No. 3, 2013
2. Eneko Osaba An Improved Discrete Bat Algorithm for Symmetric and Asymmetric Traveling Salesman Problems - Engineering Applications of Artificial Intelligence September 10, 2015
3. Huang,G.Q., Zhao W.J.,and Lu,Q. (2013)Bat algorithm with global convergence for solving large-scale optimization problem, Application Research of Computers ,Vol.30 no 3 ,1-10
4. Yang XS(2010)A new metaheuristic bat-inspired algorithm.In:Gonzalez JR *et al* (eds)Nature inspired cooperative strategies for optimization(NICSO),vol 284.Springer,Berlin , pp 65-74
5. Amr Rekaby Directed Artificial Bat Algorithm (DABA) A New Bio-Inspired Algorithm Egyptian Research and Scientific Innovation Lab (ERSIL) Cairo, Egypt *2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI) pp 1241-1246*
6. W. Annicchiarico, J. Periaux, M. Cerrolaza, G. Winter, Evolutionary Algorithmsand Intelligent Tools in Engineering Optimization, WIT Press, Portland, OR,2005.
7. A.H. Gandomi, X.S. Yang, S. Talatahari, A.H. Alavi, Metaheuristic Applications inStructures and Infrastructures, Elsevier, Waltham, MA, 2013.
8. L. Coelho, V.C. Mariani, Use of chaotic sequences in a biologically inspired algo-rithm for engineering design optimization, Expert Systems with Applications34 (2008) 1905–1913.
9. Amir H. Gandomia, Xin-She Yang, Chaotic bat algorithm, *Journal of Computational Science* 5 (2014) 224–232
10. J-s Pan *et al*. (eds) Intelligent Data Analysis and its applications, Volume 2, Advances in Intelligent system and computing 298 Springer (2014)
11. Yang, X. S., (2011), Bat Algorithm for Multiobjective Optimization, *Int. J. Bio-Inspired Computation*, Vol. 3, No. 5, pp.267-274.
12. Sashikala Mishra, Kailash Shaw, Debahuti Mishra A New Meta-heuristic Bat Inspired Classification Approach for Microarray Data Published by Elsevier Ltd(2011)