Bat Algorithm

Prof. Habiba Drias

Agenda

- Algorithm Origin
- Bat Echolocation
- Simulation of bat behavior
- Bat Algorithm (BA)
- BA as an extension of PSO
- Variants of BA
- Applications

The Algorithm Origin

- Authors: Xin-She Yang
- Title: A New Metaheuristic Bat-Inspired Algorithm,
- Publication: *Nature Inspired Cooperative Strategies for Optimization (NISCO **2010**)* (Eds. J. R. Gonzalez et al.), in Studies in Computational Intelligence, Springer Berlin, 284, Springer, 65-74 (2010).
- Affiliation: Department of Engineering, University of Cambridge, Trumpington Street, Cambridge CB2 1PZ, UK

How are bats fascinating?

- Bats are the only mammals that possess wings.
- There are around 1000 different species of bats, most of them are insectivore.
- They are of the order of Chiroptera.

- Two main Types of bats

Mega-bats

- ✓ Size: big
- ✓ Vision: well-developed
  - ➢ Visual cortices
  - ➢ Good visual acuity
- ✓ Smell sense
- ✓ No Echolocation

Micro-bats

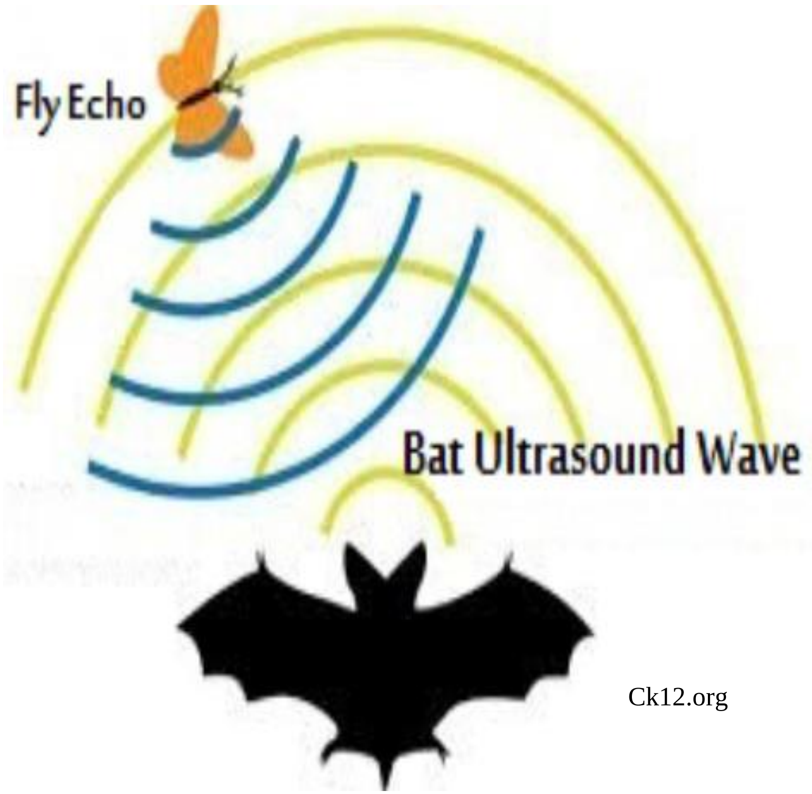
Use a sound called ***Echolocation*** to

- ✓ detect preys
- ✓ Avoid obstacles
- ✓ Locate the roosting crevices
- ✓ All these tasks are performed in complete darkness

Bats of intermediary size use a certain degree of echolocation.

Bat Echolocation

- Echolocation consists in producing a sonar composed of 2 steps:
    - emitting sound pulses
    - and then detecting surrounding objects from the reflected echo
- The sonar is also used under water by some kinds of fishes.
- It is also used by humans (Japanese) to attract and catch fishes.
- Micro-bats perceive their environment by:
    - Measuring the distance and orientation of the objects.
    - Detecting the type and the speed of the preys.



Bat Echolocation phenomenon

Simulation of Bats Behavior using Echolocation

- As for all the nature and biology-inspired approaches, bat-inspired approach simulates the movement of bats when searching for preys.

- The real bats are simulated by artificial bats as follows:

The natural bats

- Environment: the nature and the roosting crevices where the bats live.
- Swarm: bats from roosting crevices
- Target: preys (insects)
- Movement: natural based on echolocation and hunting strategies
- Bats evaluation (position, velocity …)

The artificial bats

- Search space: set of all possible solutions encapsulated in artificial bats
- Population of artificial bats or solutions
- Target: optimal solutions
- Movement: determined by mathematical equations based on simulated echolocation
- Fitness function

Approximations and Simplifications

- Bats use echolocation to sense distance and perceive in a magical way their surroundings.

- Bats fly randomly with:
  - velocity $\boldsymbol{v_i}$
  - at position $\boldsymbol{x_i}$
  - with a fixed frequency $\boldsymbol{f_{min}}$,
  - varying wavelength $\boldsymbol{\lambda}$ and loudness $\boldsymbol{A_0}$ to search for prey.

- They automatically adjust the wavelength of their emitted pulses and adjust the rate of pulse emission $\boldsymbol{r}$ from [0,1], depending on the proximity of their prey.

- The loudness varies from a positive large value $\boldsymbol{A_0}$ to a minimum value $\boldsymbol{A_{min}}$.

Bat Motion

- x' is the best solution of the current iteration.
- x* is the best solution of all the previous iterations.
- The movement of a bat is modelled as:

$$f_i = f_{min} + (f_{max} - f_{min})\beta$$
$$v_i^t = v_i^{t-1} + (x_i^t - x_*)f_i$$
$$x_i^t = x_i^{t-1} + v_i^t$$

where β is an empirical parameter

- The best solution of one iteration is

$$x_{new} = x' + \varepsilon * A_i$$

- The update of loudness and pulse rate

$$A_i^t = \alpha A_i^{t-1}$$

$$r_i^{t+1} = r_i^0(1 - e^{\gamma t})$$

Bat Algorithm

begin
generate at random a population of ***k*** bats (***k*** solutions);
for each bat i **do**
define its loudness A_i, its pulse frequency f_i and its velocity v_i;
set its pulse rate to r_i ;
select the best solution x^*;
while ((Max-Iter not reached) **do**
for each i = 1 to ***k*** **do**
compute a new solution (f_i, v_i, x_i) using these formulas

$$f_i = f_{min} + (f_{max} - f_{min})\beta$$
$$v_i^t = v_i^{t-1} + (x_i^t - x_*)f_i$$
$$x_i^t = x_i^{t-1} + v_i^t$$

if (rand > r_i) **then**
select a solution x' among the best solutions;
improve the solution using this formula ;

$$x_{new} = x' + \varepsilon * A$$

end if;
generate at random a new solution (f_i, v_i, x_i) ;
if (rand<A_i) **and** ($f(x_i) < f(x^*)$) **then**
accept the new solution;
increase r_i and reduce A_i using these formulas **end if**

$$A_i^t = \alpha A_i^{t-1}$$
$$r_i^{t+1} = r_i^0(1 - e^{\gamma t})$$

end for
Rank the bats and find the current best solution x* ;
end while
end

Data structures

- **The Bats Table.**
  - The bats table memorizes the pulse frequency, the velocity and the position of each bat at time ***t*** (each iteration).
  - Each entry corresponds to a bat for the current iteration.
  - ***x**** is the best solution found so far, by any bat in the swarm, from the beginning of the bat process. ***x'*** is the best solution found at one iteration.

Bats Table

	f_i	v_i^t	x_i^t
bat 1			
bat 2			
bat *i*			
bat *k*			

x* | ***x'***

Parameters setting

- The parameters for BA are :

- Maximum number of iterations.
- Number of bats.
- f_{min}
- f_{max}
- $\boldsymbol{\alpha}$
- $\boldsymbol{\gamma}$

Bat Algorithm: An extension of PSO

- BA can be seen as a hybridization of PSO and a local search.

BA
Balances between

PSO

Similar rules motion

- ✓ (f_i, v_i, x_i)

Local search

based on simulated annealing controlled by:

- ✓ Loudness
- ✓ Pulse rate

Important Features

- **BA is equivalent to PSO** when:
  - f_i is a random number,
  - $A_i = 0$
  - $r_i = 1$

- **BA is equivalent to Harmony Search** when:
  - $A_i = r_i$

- **BA is then more powerful than PSO and Harmony Search**

Some Variants of BA

- FLBA: Fuzzy Logic Bat Algorithm
  - Khan, K., Nikov, A., Sahai A., A Fuzzy Bat Clustering Method for Ergonomic Screening of Office Workplaces,S3T 2011, Advances in Intelligent and Soft Computing, 2011, Volume 101/2011, 59-66 (2011).

- MOBA: Multi-Objective Bat Algorithm
  - Yang, X. S., Bat algorithm for multi-objective optimization, Int. J. Bio-Inspired Computation, Vol. 3, No. 5, pp. 267–274, (2011).

- BBA: Binary Bat Algorithm
  - Nakamura, R. Y. M., Pereira, L. A. M., Costa, K. A., Rodrigues, D., Papa, J. P., Yang, X. S., (2012), BBA: A binary bat algorithm for feature selection, in: 25th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), 22-25 Aug. 2012, IEEE Publication, pp. 291-297.
  - S. Mirjalili, S. M. Mirjalili, X. Yang, Binary Bat Algorithm, Neural Computing and Applications, 2014, Springer.

Some Variants of BA

- EBA (Evolving Bat Algorithm)
  - P. W. Tsai, J. S. Pan, B. Y. Liao, M. J. Tsai, V. Istanda, Bat algorithm inspired algorithm for solving numerical optimization problems, Applied Mechanics and Materials, Vo.. 148-149, pp.134-137 (2012)
- IBA (Improved Bat Algorithm)
  - Y. Selim and U. K. Ecir, "Improved Bat Algorithm (IBA) on Continuous Optimization Problems," Lecture Notes on Software Engineering, vol. 1, no. 3, pp. 279-283, 2013.

Applications

- Continuous Optimization
  - Non-linear problems
  - Mathematic Continuous Functions
- Combinatorial Optimization
  - Scheduling problems
- Data Mining
  - Clustering problems
  - Classification problems
  - Association Rules Mining
- Computer Vision
  - Image matching

A Case Study: Max-SAT

Definition

Let $V = \{v_1, v_2, \dots, v_n\}$ be a set of Boolean variables

- a **literal** is a Boolean variable that appears with or without the negation operator.
- a **clause** is a Boolean disjunction of literals.
- the problem SAT is defined by the following (instance, question) pair:

Instance: **m** clauses over **n** Boolean variables (a CNF Boolean formula)
Question: is there an instanciation or interpretation of variables for which all the clauses are true simultaneously?

Example

$V = \{v_1, v_2, v_3, v_4\}$, $C = \{c_1, c_2, c_3\}$ such that:

$$\begin{cases} c_1 = v_1 + v_2 + \overline{v_4} \\ c_2 = v_2 + \overline{v_3} + v_4 \\ c_3 = \overline{v_1} + \overline{v_2} + v_3 \end{cases}$$

Where bar sign denotes the negation operator and '+' the disjunctive operator. One possible solution is the instanciation {T, T, T, T} respectively assigned to {v1, v2, v3, v4}.

Solving Max-SAT with BA

Solution coding

- a solution is a vector of n Boolean values, each one assigned to a variable.
- It is then a chain of n bits.

Search space

- the set of all potential instanciations for the instance.
- It is then the set of all Boolean vectors of length equal to n.
- its size is equal to 2^n.

Fitness function

- for Max-SAT
  - ✓ the number of satisfied clauses. The function is to be maximized or
  - ✓ the number of unsatisfied clauses. The function here is to be minimized.
- for Max-w-SAT
  - ✓ the sum of the weights of the satisfied clauses. The function is to be maximized or
  - ✓ the sum of the weights of the unsatisfied clauses. The function is to be minimized.

Distance between two solutions

- **Hamming Distance** between two strings x and y of same length equal to n:
  - it is equal to the number of entries with different symbols
    - $distance(x,y) = \sum_{i=1}^{i=n} x_i \oplus y_i$

      where $x_i \oplus y_i = \begin{cases} 0 & if \quad x_i = y_i \\ 1 & otherwise \end{cases}$
  - examples
    - ✓ x = ababcaabda, y = acabbaaada, then distance(x,y) = 3
    - ✓ x = 0010101110, y = 0100100000, then distance(x,y) = 5

- **Similarity** between two strings x and y of equal length
  - Similarity(x,y) = n- distance(x,y)

- **Dissimilarity** between two strings x and y of equal length
  - Dissimilarity(x,y) = distance(x,y)

- **The degree of diversity.**
  - The degree of diversity of a solution s relatively to a set S of other solutions is measured by the minimum of the distances between s and the elements in S.
  - It is calculated as: *diversity(s)* = Min { *distance(s,t), t* ∈ *S* }.

BA-SAT

begin
generate at random a population of *n* bats (*n* solutions);
for each bat *i* **do**
define its loudness A_i, its pulse frequency f_i and its velocity v_i;
set its pulse rate to r_i ;
select the best solution x^*;
while ((Max-Iter not reached) **do**
for each i = 1 to *n* **do**
compute a new solution (f_i, v_i, x_i) using these formulas
if (rand > r_i) **then**
select a solution x' among the best solutions;
improve the solution using the following formula ;
end if;
generate at random a new solution (f_i, v_i, x_i) ;
if (rand<A_i) **and** ($f(x_i) < f(x')$) **then**
accept the new solution; **end if**
increase r_i and reduce A_i using these formulas
end for
Rank the bats and find the current best x* ;
end while
end

In the initialization, consider a velocity as a distance, which is a non negative number.

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

$$v_i^t = \lfloor v_i^{t-1} + distance(x_i^t, x_*) f_i \rfloor$$

Invert v_i^t bits of x_i^{t-1} to obtain x_i^t

Perform a local search

$$A_i^t = \alpha\, A_i^{t-1}$$

$$r_i^{t+1} = r_i^0 (1 - e^{\gamma t})$$

Experiments

Table 1: Comparison of BA with GA, and PSO.

Functions/Algorithms	GA	PSO	BA
Multiple peaks	$52124 \pm 3277(98\%)$	$3719 \pm 205(97\%)$	$1152 \pm 245(100\%)$
Michalewicz's (d=16)	$89325 \pm 7914(95\%)$	$6922 \pm 537(98\%)$	$4752 \pm 753(100\%)$
Rosenbrock's (d=16)	$55723 \pm 8901(90\%)$	$32756 \pm 5325(98\%)$	$7923 \pm 3293(100\%)$
De Jong's (d=256)	$25412 \pm 1237(100\%)$	$17040 \pm 1123(100\%)$	$5273 \pm 490(100\%)$
Schwefel's (d=128)	$227329 \pm 7572(95\%)$	$14522 \pm 1275(97\%)$	$8929 \pm 729(99\%)$
Ackley's (d=128)	$32720 \pm 3327(90\%)$	$23407 \pm 4325(92\%)$	$6933 \pm 2317(100\%)$
Rastrigin's	$110523 \pm 5199(77\%)$	$79491 \pm 3715(90\%)$	$12573 \pm 3372(100\%)$
Easom's	$19239 \pm 3307(92\%)$	$17273 \pm 2929(90\%)$	$7532 \pm 1702(99\%)$
Griewangk's	$70925 \pm 7652(90\%)$	$55970 \pm 4223(92\%)$	$9792 \pm 4732(100\%)$
Shubert's (18 minima)	$54077 \pm 4997(89\%)$	$23992 \pm 3755(92\%)$	$11925 \pm 4049(100\%)$

Xin-She Yang, *Nature Inspired Cooperative Strategies for Optimization (NISCO **2010**)*, in Studies in Computational Intelligence, Springer Berlin, 284, Springer, 65-74 (2010)

Conclusions

Strength

- A promising algorithm for continuous optimization problems
  - ✓ Simulation of the original echolocation behavior of bats
    - ➢ A more sophisticated and structured process
  - ✓ Includes hybrid successful strategies of previous evolutionary algorithms such as:
    - ➢ Local search
    - ➢ Simulated annealing
    - ➢ PSO
  - ✓ Includes a random walk strategy
- Compared to other evolutionary algorithms, it has been shown to be more powerful than:
  - ✓ PSO
  - ✓ Genetic algorithm
  - ✓ Harmony search
- A good convergence
  - ✓ Thanks to the parameters α and γ that implement the convergence strategy inspired from the simulated annealing
- Numerous important variants:
  - ✓ MO-BAT, IBA, EBA, Fuzzy BA, …

Limitations and Perspectives

- **Limitations**
  - More analyses for algorithm convergence.
  - More comparisons with other functions and Problems with higher number of dimensions.

- **Perspectives**
  - More sophisticated modelling of pulse and loudness rates.
  - More variants.
  - More applications.

References

Bonabeau E., Dorigo M. and Theraulaz G.: *Swarm Intelligence: from Natural to Artificial Systems*, Oxford University Press, (1999).

Geem Z. W., Kim J. K. and Loganathan G. V.: *A new heuristic optimisation: Harmony search*, Simulation, Vol. 76(2), 60-68, (2001).

Heraguemi K. E. , Kamel N. and **Drias H.**: Association rule mining based on bat algorithm. Journal of Computational and Theoretical Nanoscience, 12(7):1195-1200, 2015.

Heraguemi K. E. , Kamel N. and **Drias H.**: : Multi-swarm bat algorithm for association rule mining using multiple cooperative strategies. Appl. Intell. 45(4): 1021-1033 (2016)

Khennak I. and **Drias H.**: Bat-Inspired Algorithm Based Query Expansion for Medical Web Information Retrieval. J. Medical Systems 41(2): 34:1-34:16 (2017)

Li C. , Nguyen T. T. , Yang M. , Yang S. and Zeng S.: Multi-population methods in unconstrained continuous dynamic environments: The challenges. Information Sciences, 296:95-118, 2015.

Li C. , Yang M. and Yang S.: An adaptive multi-swarm optimizer for dynamic optimization problems. Evolutionary computation, 22(4):559{594, 2014.

Parpinelli R. S. and Lopes H. S.: *New inspirations in swarm intelligence: a survey*, Int. J. Bio-Inspired Computation, Vol. 3, No. 1, 1–16, (2011).

References

Gandomi, A. H., Yang, X. S., Alavi, A. H., Talatahari , S. (2013). Bat algorithm for constrained optimization tasks, Neural Computing and Applications, http://link.springer.com/article/10.1007

Khan, K., and Sahai, A., (2012a). A comparison of BA, GA, PSO, BP and LM for training feed forward neural networks in e-learning context, Int. J. Intelligent Systems and Applications (IJISA), Vol. 4, No. 7, pp. 23–29.

Nakamura, R. Y. M., Pereira, L. A. M., Costa, K. A., Rodrigues, D., Papa, J. P., Yang, X. S., (2012). BBA: A binary bat algorithm for feature selection, in: 25th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), 22-25 Aug. 2012, IEEE Publication, pp. 291-297.

Richardson, P.: Bats. Natural History Museum, London, (2008)

Seyedali Mirjalili, Seyed Mohammad Mirjalili, Xin-She Yang, Binary Bat Algorithm, Neural Computing and Applications, 2014, Volume 25, Issue 3, pp 663-681.

Taha, A.M. and Mustapha A.: Multi-swarm bat algorithm. In Research Journal of Applied Sciences, Engineering and Technology, pages 1389-1395, 2015.

Tsai P. W. , Pan J. S., Liao B. Y., Tsai M. J., Istanda V.: Bat algorithm inspired algorithm for solving numerical optimization problems, Applied Mechanics and Materials, Vo.. 148-149, pp.134-137 (2012)

References

Wang G.-G. , Chang B. and Zhang Z.: A multi-swarm bat algorithm for global optimization. In Evolutionary Computation (CEC), 2015 IEEE Congress on, pages 480-485, May 2015.

Wolpert D. H. and Macready W. G.: *No free lunch theorems for optimisation*, IEEE Transaction on Evolutionary Computation, Vol. 1, 67-82, (1997).

Xin-She Yang, Bat algorithm: literature review and applications, Int. J. Bio-Inspired Computation, Vol. 5, No. 3, pp. 141–149 (2013). DOI: 10.1504/IJBIC.2013.055093

Yang, X. S., (2010). A New Metaheuristic Bat-Inspired Algorithm, in: Nature Inspired Cooperative Strategies for Optimization (NISCO 2010), Studies in Computational Intelligence Vol. 284, Springer Berlin, pp. 65–74.

Yang, X. S., (2011). Bat algorithm for multi-objective optimisation, Int. J. Bio-Inspired Computation, Vol. 3, No. 5, pp. 267–274.

Yang, X. S., (2012). Metaheuristic optimization with applications: Demonstration via bat algorithm, in: Proceedings of 5th Bioinspired Optimization Methods and Their Applications (BIOMA2012) (Eds. B.Filipic and J. Silc), 24-25 May 2012, Bohini, Slovenia, pp. 23-34.

Yang, X. S. and Gandomi, A. H., (2012). Bat algorithm: a novel approach for global engineering optimization, Engineering Computations, Vol. 29, No. 5, pp. 464–483.

References

Yang, X. S., (2013). Bat algorithm and cuckoo search: a tutorial, in: Artificial Intelligence, Evolutionary Computing and Metaheuristics (Eds. X. S. Yang), Studies in Computational Intelligence, Vol. 427, pp. 421–434.

Yang X. S.: *Nature-Inspired Metaheuristic Algorithms*, Luniver Press, (2008).

Yang X. S.: *Firefly algorithms for multimodal optimisation*, Proc. 5th Symposium on Stochastic Algorithms, Foundations and Applications, LNCS, Vol. 5792, 169-178, (2009).

Yang X. S.: *A new metaheuristic bat-inspired algorithm*, in: Nature Inspired Cooperative Strategies for Optimization (NICSO), Springer, SCI Vol. 284, 65-74, (2010).

Yang X. S., Cui Z. H., Xiao R. B., Gandomi A. H. and Karamanoglu M.: *Swarm Intelligence and Bio-Inspired Computation: Theory and Applications*, Elsevier, London, (2013).

Thank You !