Solving Second Order Linear Boundary Value Problem by Bat Algorithm.

Fatima Ouaar*
Laboratory of Applied Mathematics,
Mohamed Khider University, Biskra, Algeria

Abstract

This study aims to solve a second-order Boundary Value Problem (BVP) using Bat Algorithm (BA) created by Xin-She Yang, who took inspiration from the echolocation habits of microbats with varying emission and loudness pulse rates. By considering two numerical instances, the problem formulation is based on a finite difference approximation of the discretized derivative. When compared to the exact solution, the algorithm results, and the finite difference method, the experimental results demonstrate that the BA yields a satisfactorily precise approximation of the solutions, confirming the usefulness and effectiveness of the proposed method and demonstrating its applicability for solving BVP.

Keywords: Bat Algorithm (BA), numerical optimization, centre-difference formula, Two-point Boundary Value Problems (BVP).

2010 Mathematics Subject Classification: 65L05, 65L12, 65L10.

1 Introduction and Motivation

In mathematics, information science, and decision theory, optimization is the best solution within a specified domain that may minimize or maximize a function. Numerical optimization is an essential method in decision science, engineering, and physical systems research [18]. For optimization problems, we have to find [25] solutions that are either optimum or almost optimal with respect to certain goals. But the modern issues are so complicated that the traditional optimization methods don't work. Because of this, many scholars from many fields have utilized nature to develop a range of Metaheuristic Algorithms (MA), which are designed to tackle a large number of difficult optimization issues without necessitating a great deal of customisation for each one.

In MA, the following characteristics are nearly always present: First of all, they use ideas from physics, biology, or ethology, drawing inspiration from the natural world. Second, they make use of stochastic components, which include chance aspects. Thirdly, they don't use the Hessian matrix or gradient of the objective function. Fourth, they contain certain requirements that need to be modified to fit the particular problem at hand. By adopting different shapes dependent on the processes inspired by the systems, MA demonstrate their potential to address a broad range of optimization difficulties [26] by fostering a beautiful interaction between discovery and exploitation. Several examples are the Ant Colony Optimization (ACO) [5], Genetic Algorithms (GA) [8], Particle Swarm Optimization (PSO) [12], Artificial Bee Colony algorithm [10] and so forth. Numerous benefits of each of these algorithms are demonstrated by their wide range of applications.

Yang [22] initially presented the BA as a substitute technique for numerical optimization [23]. By generating strong echolocation, it imitates the way bats use echolocation to find prey. Nonetheless, in order to get over its shortcomings and capitalize on its strengths, BA was hybridized with other MA that were inspired by nature. Various adjustments have been suggested in this regard to enhance BA's performance. For instance, a BA that uses Lévy Flights and Differential Evolution (DE) operators throughout the optimization process is presented in [21]. A directional BA was published in 2017 [4], whereby directional echolocation is suggested as a means to enhance BA exploration. A noteworthy enhancement was suggested in [7], whereby the hybridization of BA and DE is employed. Additionally, a BA was hybridized with harmony search for global numerical optimization in [20]. In order to improve the conventional BA's search capabilities, chaotic maps have been used in place of normal distribution [2]. Komarasamy and Wahi (2012) developed the K-Means Bat Algorithm (KMBA) [13]. Nakamura et al.'s Binary Bat Algorithm (BBA) [16]. BA and its expansions have several uses, particularly in continuous optimization [24]. In engineering design, see [17]. Image matching [6], data mining, Microarray Data [15], and so on.

A BVP is a differential equation with boundary conditions in the subject of differential equations in mathematics [3]. According to [1], a solution to this equation needs to meet the specified boundary conditions. The BVPs may be found in many areas of engineering, control, optimization theory, and applied mathematics. Typically, the conventional analytical techniques [18], [19] cannot precisely solve BVPs since only a small number of them can be solved in closed forme using common mathematical procedures. As an alternative, computer algorithmic approximation approaches [3] have been utilized to get approximate solutions to a number of decimal points in the absence of solutions using numerical methods [9].

In this work, we employ a systematic approach based on the application of BA to numerically approximate a BVP solution [11], whereby the specified boundary conditions might be of concern. This paper is significant because it treats the second order linear BVP as an optimization issue. Since BA [22] is used to find optimal numerical solutions, the objective function is based on a finite difference approximation of the discretized derivative [14]. As a result, the obtained results are compared by the exact results and the results of the finite difference method.

This paper is organized as follows. In Section 2, we formulate the BVP as an optimization problem; section 3 provides basics on BA and its main steps for finding an approximate solution of BVP. The Section 4 exposes examples of second order linear BVP to show how BA can lead to satisfactory results. The comments and conclusion are made in section 5.

2 Bat Algorithm Overview

In 2010 [22], Yang gave a BA presentation. It imitates the echolocation behavior of microbats, as they are able to produce strong echolocation. One way to summarize this advantageuse algorithm is [23] Bats employ a loud sound beat to differentiate between barriers and prey in their local environment. This sound beat generates echoes from numerous bats nearby to ricochet back at a certain frequency. The bats are able to determine the distance between the obstruction and the prey as a result. They fly randomly, fluctuating in speed, frequency, and loudness, in search of food. Finding the prey nearest to us is the objective function's solution. The frequency and zooming parameters maintain the balance between exploration and exploitation operations. The algorithm kept running until the convergence requirements were met. The pseudo code of BA is represented in 2.1. The idealized rules of BA are:

1. To using echolocation to sense distance, all bats have the unexplained ability to discriminate between background barriers and food or prey.

2. Bats fly at point x_i with random velocity v_i, fixed frequency f_{min}, varying wavelength λ, and loudness A_0 in search of food. They may automatically adjust the wavelength (or frequency) and the rate of pulse emission $r \in [0, 1]$ in response to variations in target vicinity.

3. Although the loudness might vary in many ways, we'll assume that it does so as follows: from a large (positive) A_0 to a small constant number A_{min} .

```
Objective function f(x), x = (x1, ..., xd)^T
Initialize the bat population xi (i = 1, 2, ..., n) and vi
Define pulse frequency fi at xi
Initialize pulse rates ri and the loudness Ai
while (t < Max number of iterations)
Generate new solutions by adjusting frequency,
and updating velocities and locations/solutions
    if (rand > ri)
    Select a solution mong the best solutions
    Generate a local solution around the selected best solution
    end if
    Generate a new solution by flying randomly
    if (rand < Ai & f(xi) < f(x*))
    Accept the new solutions
    Increase ri and reduce A
    end if
Rank the bats and find the current best x*
end while
Postprocess results and visualization
```

Table 2.1: Pseudo code of BA

3 Problem formulation

We approximate the two-point second order BVP in this study. Given a function f : $R^3 \rightarrow$ R and certain integers α, β define the function. The issue

$$Y'' = f(x, y, y') \quad a \leq x \leq b, \tag{3.1}$$

according to the boundary conditions:

$$y(a) = \alpha, \quad y(b) = \beta, \tag{3.2}$$

referred to as two-point BVP. Under some circumstances, it has a unique solution that depends on how the function f and the boundary value work together to decide the result.

The necessary condition is given by the following theorem.

Theorem 3.1 *Assume that,* $X = (x_1, s_1, s_2) : x \in [a, b], s_1, s_2 \in \mathbb{R}$ *and* $f : R^3 \to \mathbb{R}$ *is a given function with the properties:*

- $f \in C(X)$,
- $\partial_1 f, \partial_2 f \in C(X)$,
- $\partial_2 f > 0$ *on* X,
- *There exists a nonnegative constant* M *such that* $|\partial_3 f| \leq M$ *on* X.

Mathematically speaking, optimization is minimization or maximization of a function subject to constraints on its variables [9]. In this section, second order BVP is first transformed into discrete version in order to formulate it as an optimization problem based on the minimization of the cumulative residual of all unknown interior nodes. The aim of this paper is to apply BA [22] for solving Two Point BVP for ordinary differential equations. More specifically, we consider the following system:

$$y''(x) + \frac{a_1(x)}{p_1(x)} G\big(x, y'(x)\big) + \frac{a_2(x)}{p_2(x)} G\big(x, y(x)\big) + \frac{a_3(x)}{p_3(x)} = 0 \tag{3.3}$$

subject to the boundary conditions in (3.2) where $x \in (a, b)$, α, β are real finite constants, and G is linear or nonlinear real-valued functions.

Remark 3.1 *the two functions* $p_i(x)$, $q_i(x)$ *may take the values* $p_i(0) = q_i(0) = 0$ *or* $p_i(1) = q_i(1) = 0$ *which make (3.3) to be singular at* $x=a$ *or* $x=b$, *while* $a_i(x)$ *are continuous real-valued functions on* $[a, b]$, *where* $i = 1, 2, 3$.

Remark 3.2 *throughout this work, me assume that 3.3 subject to boundary conditions (3.2) has a unique solution on* $[a, b]$.

Remark 3.3 *the term continuous is used to emphasize that the continuous nature of the optimization problem and the continuity of the resulting solution curves.*

The independent interval $[a, b]$ is divided into N subintervals of equal length k, which is given as $k = b - a/N$, for the first stage of formulation. The formula $x_i = ik, i = 0, 1, ..., N$ is used to determine the mesh points, or nodes. Consequently, the system to be approximated is provided as follows at the inner mesh points, x_i, $i = 1, 2, ..., N$-1:

$$y''(x_i) + \frac{a_1(x_i)}{p_1(x_i)} G\big(x_i, y'(x_i)\big) + \frac{a_2(x_i)}{p_2(x_i)} G\big(x_i, y(x_i)\big) + \frac{a_3(x_i)}{p_3(x_i)} = 0 \tag{3.4}$$

subject to the boundary conditions

$$y(x_0) = \alpha, y(x_N) = \beta,$$

Where $x_1 \leq x_i \leq x_{N-1}$, for $i = 1, 2, ..., N$-1.y (x_i) and $y(x_i)$ approximation formulas in Eq. (3.4) are substituted to complete the formulation. This yields a discretized form of Eqs. (3.3) and (3.2). A discrete function of x_i, $y(x_{i-(n-1)})$, $y(x_{i-(n-2)})$,..., and y($x_{i+(n-1)}$) will be the algebraic equations that arise. Once that is done, the discretized Eq. (3.4) has to be rewritten as follows:

$$F\left(x_i, y(x_{i-(n-1)}), y(x_{i-(n-2)}), \dots, y(x_{(i+(n-1))})\right) + \frac{a_2(x_i)}{p_2(x_i)} G_2\big(x_i, y(x_i)\big) + \frac{a_3(x_i)}{p_3(x_i)} \approx 0$$

Where F is given as:

$$F\left(x, y'(x), y''(x)\right) = y''(x) + \frac{a_1(x_i)}{p_1(x_i)} G\big(x_i, y'(x_i)\big),$$

Definition 3.1 *the residual of the general interior node, denoted by R, is defined as:*

$$R(i) = F\left(x_i, y(x_{i-(n-1)}), y(x_{i-(n-2)}), \dots, y(x_{(i+(n-1))})\right) + \frac{a_2(x_i)}{p_2(x_i)} G_2\big(x_i, y(x_i)\big) + \frac{a_3(x_i)}{p_3(x_i)} \tag{3.5}$$

Definition 3.2 *The objective function O, is a function of the residuals of all interior nodes. It may be stated as:*

$$O = \sqrt{\sum_{i=0}^{N} R^2(i).}$$

In the aim to simplify this study we take a simple form of Eq. (3.3)

$$y''(x) = \frac{a_3(x)}{p_3(x)}, \tag{3.6}$$

when the two functions $p_i(x) and\ q_i(x)$ take the values $p_i(0) = q_i(0) = 0\ \ or\ p_i(1) = q_i(1) = 0$ In (3.6) we put $a_3(x)/p_3(x)$=$y(x)$ consequently (3.6) be

$$y''(x) = y(x), \tag{3.7}$$

then to obtain the objective function we use the centre finite difference formula for the derivative we obtain:

$$y^{"}(x_i) \approx \frac{y_{i+1} - 2y_i + y_{i-1}}{2h} \approx F(y_{i+1}, y_{i-1}, y_i),$$

with the help of above equation the original differential equation is rewrite in discretized form as follows:

$$F(y_{i+1}, y_{i-1}, y_i) \approx y(x_i),$$

consequently, we have to consider the error formula (The residual of i^{th} node (r_i)) as:

$$R(i) = F(y_{i+1}, y_{i-1}, y_i) - y(x_i),$$

The objective function, associated to $Y = (y_1, y_2, \ldots, y_N)$ for N=10 will be:

$$O = \sqrt{\sum_{i=0}^{10} R^2(i)}.$$

In actuality, the number of mesh points used determines how accurate the answer is. Nonetheless, one can achieve a desired level of solution accuracy by increasing the number of mesh points. This method will be used in this study to approximate the solutions to the Eqs (3.3) and (3.2) employing BA in a numerical manner.

4 Numerical Results

To demonstrate and explain the computational efficacy of our technique, we look at a few example problems. We always employed the same step size, k. For each model problem, the absolute error and the comparison between the exact findings and the BA results are shown in Tables (4.3) and (4.4). The parameters used for the BA are shown in Tables (4.2). The Matlab environment version R2013a compiler was used to do all calculations on an Intel Duo Core 2.20 GHz PC running Microsoft Windows 2007 Professional. The iteration is continued until either 10^3 is reached in the number of iterations or the maximum difference between two following iterations is smaller than $10^{(-16)}$. The solutions are determined on N-1 nodes.

Parameter	Quantity
Dimension of the search variables (N)	10
Number of generations (d)	1000
Population size (n)	20
Loudness (constant or decreasing) (A)	0.5
Pulse rate (constant or decreasing) (r)	0.5

Table 4.2: Parameters adopted for the BA.

Let us look at a two model problems of second order linear BVP:

Problem 1. The first model problem is

$$y''=6x, 0\leq x\leq 2,$$
$$y(0)=0, y(2)=8.$$

where the analytic solution of this problem is the function $y(x)=x^3$.

Problem 2. The second model problem is

$$y'' = -K^2 y(x) + (K^2 - \pi^2)\sin(\pi x),$$
$$y(0)=0, y(1)=0, x\in [0,1].$$

The analytical solution is found to be $y(x)=\sin(\upsilon x)$.

We use the uniform mesh with $N=9$;

$$\overline{\omega_h} = \left\{x_1 = ih, i = 0,1,2,3,4,5,6,7,8,9,10, h = \frac{2-0}{9+1} = 0.2\right\}$$

Hence, the mesh points for problem 1 are

$x_0=0, x_1=0.2, x_2=0.4, x_3=0.6, x_4=0.8, x_5=1.0, x_6=1.2, x_7=1.4, x_8=1.6, x_9=1.8, x_{10}=2.0.$

The mesh points for problem 2 are

$x_0=0, x_1=0.1, x_2=0.2, x_3=0.3, x_4=0.4, x_5=0.5, x_6=0.6, x_7=0.7, x_8=0.8, x_9=0.9, x_{10}=1.0.$

The obtained results, the comparison of the method to the exact solution and the incurred error in Bat algorithm's method are shown in tables (4.3) and (4.4) are visualized via figures (4.1) and (4.2) respectively. These pictures show the solutions. The continuous line is the correct solution; the dotted lines are the approximation solutions. We can see the approximations converge to the correct solutions.

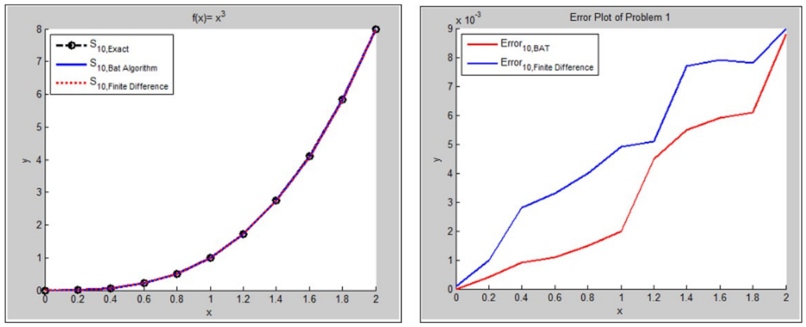

Figure 4.1: Results of problem 1.

Node	x_i	Exact results	Finite Difference	Absolute Error	BA	Absolute Error
0	0	0.0000	0.0001	0.0001	0.0000	0.0000
1	0.2	0.0080	0.0070	0.0010	0.0076	0.0004
2	0.4	0.0640	0.0612	0.0028	0.0631	0.0009
3	0.9	0.2160	0.2127	0.0033	0.2149	0.0011
4	0.8	0.5120	0.5080	0.0040	0.5105	0.0015
5	1.0	1.0000	0.9951	0.0049	0.9980	0.0020
6	1.2	1.7280	1.7229	0.0051	1.7235	0.0045
7	1.4	2.7440	2.7363	0.0077	2.7385	0.0055
8	1.6	4.0960	4.0881	0.0079	4.0901	0.0059
9	1.8	5.8320	5.8242	0.0078	5.8259	0.0061
10	2.0	8.0000	7.9910	0.0090	7.9912	0.0088

Table 4.3: Numerical results of studied methods vs Exact results for Problem 1 for d=10

.

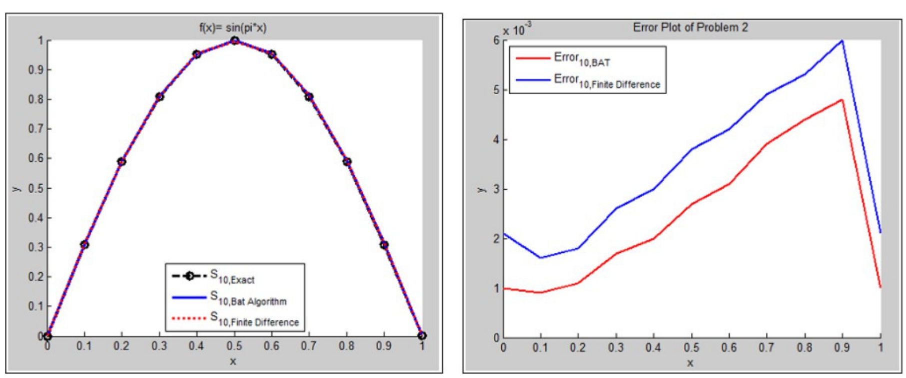

Figure 4.2: Results of problem 2.

Node	x_i	Exact results	Finite Difference	Absolute Error	BA	Absolute Error
0	0	00000	0.0021	0.0021	00000	0.001
1	0.1	0.3090	0.3106	0.0016	0.3090	0.0009
2	0.2	0.5878	0.5896	0.0018	0.5878	0.0011
3	0.3	0.8090	0.8116	0.0026	0.8090	0.0017
4	0.4	0.9511	0.9541	0.0030	0.9511	0.0020
5	0.5	1.0000	0.9962	0.0038	1.0000	0.0027
6	0.6	0.9511	0.9553	0.0042	0.9511	0.0031
7	0.7	0.8090	0.8139	0.0049	0.8090	0.0039
8	0.8	0.5878	0.5931	0.0053	0.5878	0.0044
9	0.9	0.3090	0.3150	0.0060	0.3090	0.0048
10	1.0	0.0000	0.0021	0.0021	0.0000	0.001

Table 4.4: Numerical results of studied methods vs Exact results for Problem 2 for d=10

5 Conclusion

This study discusses the usage of conventional BA to solve BVP when it's employed as a tool for numerical optimization using selected cases. Upon comparing the precise solutions, algorithmic results, and finite difference technique outcomes, it was shown that BA provided exponentially superior solutions with the least amount of error (keep in mind that the error function's ideal value is zero).

References

[1] Baxley, J. V., (1981), Nonlinear Two Point Boundary Value Problems in Ordinary and Partial Differential Equations (Everrit, W.N. and Sleeman, B.D. Eds.), Springer- Verlag., New York, pp. 46-54.

[2] Chaotic bat algorithm, (2014), Computational Science, 5(2): 224-32.

[3] Collatz, L ., (1966), Numerical Treatment of Differential Equations (3/ed.), Springer Verlag., Berlin.

[4] Chakri, A., Khelif, R., Benouaret, M., Yang, X.S., (2017), New directional bat algo- rithm for continuous optimization problems. Expert Systems with Applications.

[5] Dorigo, M., Maniezzo, V., Colorni, A., (1996), The ant system: optimization by a colony of cooperating agents. IEEE Trans. Syst. Man Cybern. B 26, pp- 29-41.

[6] Du, Z. Y., and Liu B., (2012), Image matching using a bat algorithm with mutation, Applied Mechanics and Materials, Vol. 203, No. 1, pp. 88–93. numerical optimization problems, Applied Mechanics and Materials., Vol 148-149, pp.134-137.

[7] Fister, I.J., Fister, D., and Yang, X.S., (2013), A Hybrid Bat Algorithm, Elek- trotehniški vestnik., (in press).

[8] Goldberg, D.E., (1989), Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley: Reading, MA.

[9] Jain, M. K., Iyenger, S. R. K. and Jain, R. K., (1987), Numerical Methods for Scientific and Engineering Computation (2/ed.), Wiley Eastern Ltd,. New Delhi.

[10] Karaboga, D., and Basturk, B., (2007), A powerful and efficient algorithm for numer- ical function optimization: artificial bee colony (ABC) algorithm, Journal of Global Optimization, 39, pp- 459–471.

[11] Keller, H. B., (1968), Numerical Methods for Two Point Boundary Value Problems, Waltham, Mass, Blaisdell Publ, Co.

[12] Kennedy, J., and Eberhart, R. C., (1995), Particle swarm optimization, In: Proceed- ings of IEEE International Conference on Neural Networks. No. IV. 27 Nov–1 Dec, pp. 1942–1948, Perth Australia.

[13] Komarasamy, G., and Wahi, A., (2012), An optimized K-means clustering technique using bat algorithm, European J. Scientific Research, Vol. 84, No. 2, pp. 263-273.

[14] Li, J., (2005), General explicit difference formulas for numerical differentiation, Jour- nal of Computational and Applied Mathematics., 183, pp. 29-52.

[15] Mishra, S., Shaw, K., and Mishra, D., (2012), A New Metaheuristic Bat Inspired Classification Approach for Microarray Data, Procedia Technology., vol.4, pp. 802 – 806.

[16] Nakamura, R. Y. M., Pereira, L. A. M., Costa, K. A., Rodrigues, D., Papa, J. P., and Yang, X. S., (2012), BBA: A Binary Bat Algorithm for Feature Selection, Graphics, Patterns and Images (SIBGRAPI), pp. 291-297

[17] Selim, Y., Ecir, U. K., (2015), A new modification approach on bat algorithm for solving optimization problems, Applied Soft Computing., Volume 28, pp. 259–275.

[18] Stoer, J. and Bulirsch , R., (1991), Introduction to Numerical Analysis (2/e), Springer-Verlag., Berlin Heidelberg.

[19] Troesch, B.A., (1976), A simple approach to a sensitive two-point boundary value problem. J. Comput. Phys. 21, pp. 279-290.

[20] Wang, G. G., Guo, L. H., (2013), A novel hybrid bat algorithm with harmony search for global numerical optimization, Journal of Applied Mathematics, (in press).

[21] Xie, J., A Zhou, Y., A Chen, H., (2013), A novel bat algorithm based on differential operator and lévy flights trajectory. Computational Intelligence and Neuroscience.

[22] Yang, X. S., (2010), A New Metaheuristic Bat-Inspired Algorithm, NICSO 2010, SCI 284, pp. 65–74.

[23] Yang, X. S., and Gandomi, A. H., (2012), Bat Algorithm: A Novel Approach for Global Engineering Optimization, Engineering Computations, Vol. 29, Issue 5, pp. 464–483.

[24] Yang, X. S., (2013), Bat algorithm: literature review and applications, Int. J. Bio- Inspired Computation, Vol. 5, No. 3, pp. 141–149.

[25] Yang , X. S., (2014), Book Nature Inspired Optimization Algorithm, Elsevier.

[26] Yang, X. S., (2008), Nature-Inspired Metaheuristic Algorithms, Luniver Press.