Black Hole Algorithm: Solving Optimization Problems in Python

Izrulfizal Saufihamizal Ibrahim[1], Rosziati Ibrahim[2], and Jahari Abdul Wahab[3]

[1]Phd Candidate, Department of Software Engineering, Universiti Tun Hussein Onn Malaysia, Johor, Malaysia
[2]Supervisor, Department of Software Engineering, Universiti Tun Hussein Onn Malaysia, Johor, Malaysia
[3]Industry Collaborator, Engineering R&D Department, Sena Traffic Systems Sdn. Bhd., Kuala Lumpur, Malaysia

Corresponding author: Rosziati Ibrahim (rosziati@uthm.edu.my).

ABSTRACT Optimization problems can be found in many fields including engineering, medical, education, and even science. These problems can be solve using traditional mathematics but nowadays algorithms have been used to solve these problems more efficiently with less execution time. In this paper, an optimization algorithm called Black Hole Algorithm (BHA) will be explained and applied to three continuous test functions which are Akley, Beale, and Booth functions with the difference in sample size as the parameter. Based on the benchmarking results of the three functions, because of BHA's random sample generation, functions like Beale took more iterations than Ackley and Booth which both have iterations less 20 while Beale took over 20.

INDEX TERMS Black Hole, Metaheuristics, Minimization methods, Optimization.

I. INTRODUCTION

Optimization challenges are unavoidable in today's society, as different systems are deployed practically everywhere, including the trade market, transportation, scheduling, and engineering. To overcome these challenges, a variety of solutions have been devised, including the typical conventional methods. Optimization problems can now be tackled utilizing a metaheuristics technique that produces a more efficient result. As a result, this condition encourages academics to improve existing metaheuristics or devise a new algorithm to tackle the problem more effectively.

In this paper, the optimization problem that is focused on is the test functions or the artificial landscape. The test functions chosen are the single-objective optimization which is the Ackley, Beale, and Booth function.

BHA is then being applied to the test functions to get the minimum solutions. Since BHA utilizes a sample of candidates or better known as stars, then it is being chosen for the manipulated parameter. We found that BHA handles Beale function the slowest compared to the other test functions. The number of stars used for the research are 300, 600, and 900 with the same number of iterations of 150.

II. LITERATURE REVIEW

The difference between the two data models can be start with how the data are being structured. The structuring of the data is crucial as it defines how the data will be regulated and organized. Data model structuring can be defined as how the data is stored in the database and in what configurations.

A. OPTIMIZATION PROBLEM

In giving an answer for an issue, the likelihood of experiencing an issue that will require a various arrangement is unavoidable (Farnad, B., Jafarian, A., and Baleanu, D, 2018). These are the sorts of issues wherein a best arrangement should be picked for the issue to be addressed accurately. The fundamental objective in a goal work is to either boost or limit a goal work f(x,y) where y is the answer for x being a component of a bunch of occurrences. An illustration of an advancement issue is when in a circumstance where there are numerous assignments with various targets to be fulfilled with various time constraint. To tackle this, a planning framework can be carried out (Chen, W, 2015). Be that as it may, in doing as such, there will be answers for a more limited time frame of execution which is the most ideal arrangement. This is the place where the goal capacity will be executed and addressed.

With an issue of various conduct to such an extent that is have no reach and can take any worth often genuine number, consistent streamlining contrasts from discrete advancement

by its reformulation in obliged issues (Li, J., Sun, Y., and Hou, S, 2021). A couple of models potentially look good assuming the variables take on characteristics from a discrete set, oftentimes a subset of numbers, while various models contain factors that can take on any certifiable worth. Models with discrete variables are discrete upgrade issues; models with unending elements are persevering smoothing out issues. Constant smoothing out issues will regularly be clearer to deal with than discrete headway issues; the flawlessness of the limits infers that the veritable limit and impediment work regards at a point x can be used to finish up information about concentrations in a neighborhood of x.

It is normal that the data for the given issue is known exactly. Regardless, for a few authentic issues, the data can't be known unequivocally for an arrangement of reasons. The chief explanation is a result of fundamental assessment screw up. The second and more significant clarification is that a couple of data address information about the future and can't be known with conviction (Chakri, A., Khelif, R., Benouaret, M., and Yang, X. S, 2017). In upgrade under weakness, or stochastic improvement, the weakness is joined into the model. Strong improvement methodology can be used when the limits are known exceptionally inside explicit limits; the goal is to observe a response that is attainable for all data and ideal in some sense (Chakri et al, 2017). Stochastic improvement models exploit the way that probability courses directing the data are known or can be evaluated; the goal is to discover some game plan that is possible for all potential data models and advances the ordinary show of the model.

B. METAHEURISTICS

Metaheuristics are approaches that uses normal peculiarities to accomplish a heuristic arrangement or to intricate, a superior answer for conventional issues with results that are quicker and more precise. In metaheuristic computations, meta-implies 'past' or 'more raised level'. They overall perform better contrasted with essential heuristics. All metaheuristic computations use some compromises of neighborhood search and overall examination. The variety of plans is regularly recognized through randomization. Regardless of the reputation of metaheuristics, there is no agreed significance of heuristics and metaheuristics in the composition (Dokeroglu, T., Sevinc, E., Kucukyilmaz, T., and Cosar, A, 2019). A couple of researchers use 'heuristics' and 'metaheuristics' alternately. Regardless, the new example will in everyday name all stochastic estimations with randomization and overall examination as metaheuristic. Randomization gives a fair approach to moving away from adjacent chase to the pursuit on the overall scale. In this manner, essentially all metaheuristic estimations are normally fitting for nonlinear exhibiting and overall improvement (Morales-Castañeda, B., Zaldivar, D., Cuevas, E., Fausto, F., and Rodríguez, A, 2020).

For as long as anyone can remember, especially at the early seasons of humankind's arrangement of encounters, the essential method for managing decisive reasoning has reliably been heuristic or metaheuristic - by trial and error. Various critical exposures were done by 'thinking about novel thoughts', and routinely incidentally; that is heuristics. Archimedes' Eureka second was a heuristic triumph. For sure, our step-by-step opportunity for growth is predominantly heuristic. The unmistakable quality and achievement of metaheuristics can be credited to many reasons, and one of the essential reasons is that these computations have been made by copying the best cycles in nature, including natural structures, and physical and substance processes. For most estimations, we know their essential parts, yet the way that exactly these parts convey to achieve usefulness really remains commonly confidential, which rouses more powerful examinations. Get together examination of a few computations shows some information, but in regular mathematical assessment of metaheuristic estimations really has many open requests despite everything a persistent powerful investigation point.

C. TEST FUNCTIONS

Test functions are a type of quality assurance that test the performance of an algorithm and ensuring that it is up to the task. It is crucial to first comprehend the sets of inputs, and finding the outputs for the function in order to make sure the algorithm that is being tested is behaving correctly. Failure to do so will result in the algorithm itself being not reliable to solve such problems let alone an optimization problem.

In this research, the problem that is being tested on is the single-objective optimization problem in finding a specific minimum solution. The first function is the Ackley function which is a non-convex function. The minimum solution of the function is at point 0 which is f(0,0) with a search boundary of $-5 \leq x,y \leq 5$. This function can be defined on a 2-dimensional domain as:

$$f(x,y) = -20\exp\left[-0.2\sqrt{0.5(x^2 + y^2)}\right] - \exp[0.5(cos2\pi x + cos2\pi y)] + e + 20 \qquad (1)$$

The next function is the Beale function which is a multimodal function. The minimum solution of the function is at point 0 which is f(3,0.5) with a search boundary of -4.5 $\leq x,y \leq 4.5$. This function can be defined on a 2-dimensional domain as:

$$f(x,y) = (1.5 - x + xy)^2 + (2.25 - x + xy^2)^2 + (2.625 - x + xy^3)^2 \qquad (2)$$

The next function is the Booth function. The minimum solution of the function is at point 0 which is f(1,3) with a search boundary of $-10 \leq x,y \leq 10$. This function can be defined on a 2-dimensional domain as:

$$f(x,y) = (x + y - y^2) + (2x + y - 5)^2 \qquad (3)$$

D. BLACK HOLE PHENOMENA

A black hole is an object that is being left when a star of a monstrous size finally collapsed. The pulling force of gravity inside the black hole is extensive that not even visible light can abscond from the black hole. Any object that traverses through the edge of the center will be gulped by it and evaporate and disintegrate. The circle edge of the center in space is called the occasion skyline. The range of the occasion skyline is named as the Schwarzschild span (Kumar, S., Datta, D., & Singh, S. K, 2015). At this range, the departure speed is equivalent to the light speed, and when light goes through, even it can't get away. Not a single object can escape from inside the occasion skyline since nothing can go quicker than light. In calculating the radius of the black hole, a specific formula can be used which is the Schwarzschild radius and calculated as follows (Kumar et al, 2015):

$$R = \frac{2GM}{c^2}$$

In which G is the gravity constant,

M is the Black Hole's mass,

and c is the speed of light in space

FIGURE 1. Schwarzschild calculation.

E. BLACK HOLE ALGORITHM

The Black Hole Algorithm (BHA) is a population-based strategy that has some normal highlights with other population-based strategies. Likewise with other population-based calculations, a populace of up-and-comer answers for a given issue is created and dispersed arbitrarily in the pursuit space. The population-based calculations develop the made populace towards the ideal arrangement through specific systems. For instance, in genetic algorithm, the development is finished by change and hybrid tasks. In particle swarm optimization, this is finished by moving the competitor arrangements around in the pursuit space utilizing the best tracked down areas, which are refreshed as better areas are found by the up-and-comers.

Like other population-based calculations, in the Black Hole Algorithm an arbitrarily produced populace of up-and-comer arrangements which is the stars are put in the pursuit space of some issue or capacity. After instatement, the wellness upsides of the populace are assessed and the best applicant in the populace, which has the best wellness esteem, is chosen to be the dark opening and the rest structure the typical stars. The dark opening can assimilate

the stars that encompass it. The pseudocode for the algorithm is shown in Figure 2 (Kumar et al, 2015).

FIGURE 2. BHA Pseudocode.

III. RELATED WORKS

Several past researches are also done and can be viewed in which they have different parameters and frameworks but still on the same page of finding the solution for optimization problem. The author Davut Izci [8] wrote a paper about solving benchmark functions as such it is similar to those of Ackley which is the Sphere, Rosenbrock and Ackley itself but using different approach. The approach that was being used were the hybridization of Artificial Electric Field Algorithm and the traditional Nelder-Mead Simplex Method and better know together as the AEF-NM Algorithm. From the research, results are shown that the hybridization of the two methods work wonderfully in demonstrating better in optimization. Other than that, an author uses Particle Swarm Optimization (PSO) to tackle the Ackley function also produce promising results. The results were shown to only have 0.05 error value compared to the theoretical optimization solution. (Cai, W., Yang, L., & Yu, Y., 2020).

ChangJun Wen, Bo Xia, and Xin Liu in 2017 [10] did research in hybridization of Simulated Annealing Algorithm (SA) and Particle Swarm Optimization (PSO) which is called SAPSO. The hybrid algorithm is then being used to solve the original Ackley function as well as the second order Ackley function. The result shows that the hybridization process is a great success as the results shows significant change in optimization regarding the best fitness value and the first iteration value. A study had also been conducted by Ganesh Kakandikar and Omkar Kulkarni in 2017 [11] regarding the Grasshopper Optimization Algorithm (GOA) in solving several benchmark functions namely Beale function, Matya function, and two Rosenbrock functions which is constrained to cubic and to a disk. The results shows that GOA produce solution that is close to the theoretical solution which shows that GOA can be utilized in both constrained and unconstrained optimization problem. GOA also helps in preventing the solution from being trap in

a local optima by using the attractive and repulsive force of the grasshoppers or the candidates.

IV. APPLYING BHA

In applying BHA in testing the functions, it is being utilized by manipulating the parameter of the sample size or better known as the stars. The number of sample size differs from three sets of experiment which is 300, 600, and 900. However, the number of iterations remain the same which is 150. This is to ensure the ambiguity of the result and to get a more accurate result with no external factors influencing the result. The setup of the experiment is shown below:

TABLE I
SETUP OF EXPERIMENTS

Test function	Sample size	Iteration	Expected solution
Ackley	300	150	[0,0]
Beale	600	150	[3,0.5]
Booth	900	150	[1,3]

V. RESULT AND ANALYSIS

The result gained from the experiment is shown as follows:

TABLE 2
RESULT FOR 300 SAMPLE SIZE

Test function	Actual solution	Optimum iteration	Execution time
Ackley	[0.0,0.0]	15	2.0s
Beale	[3.0,0.5]	25	1.3s
Booth	[1.0,3.0]	9	1.2s

For a sample size of 300, the optimum iteration needed for Booth peaks those from Ackley and Beale along with the execution time being the shortest among the three test functions.

TABLE 3
RESULT FOR 600 SAMPLE SIZE

Test function	Actual solution	Optimum iteration	Execution time
Ackley	[0.0,0.0]	9	4.8s
Beale	[3.0,0.5]	35	3.5s
Booth	[1.0,3.0]	13	3.3s

For a sample size of 600, the optimum iteration for Ackley is the best compared to Beale and Booth while the execution time for Ackley is the slowest among the three.

TABLE 4
RESULT FOR 900 SAMPLE SIZE

Test function	Actual solution	Optimum iteration	Execution time
Ackley	[0.0,0.0]	15	8.4s
Beale	[3.0,0.5]	25	6.4s
Booth	[1.0,3.0]	9	6.2s

For a sample size of 900, Booth function requires less iteration compared to the other two test functions. As for the execution time, it is the same as the iteration which is led by Booth.
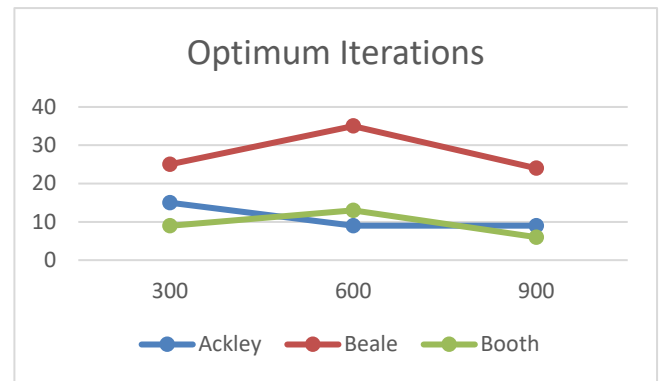

FIGURE 3. Optimum Iteration Graph.

As for the analysis, it can be seen from Figure 3, which is the graph for the optimum iteration, BHA took the greatest number of iterations to find the solution for Beale function for the three sample sizes. Then followed by a cross of parameter between Ackley and Booth.

As for the graph of execution time, it is clear from Figure 4 that Ackley function took more time to be executed until 150 iterations for the three sample sizes which is significantly higher compared to Beale and Booth.

FIGURE 4. **Execution Time Graph.**

VI. CONCLUSION

As a conclusion, this paper presented the implementation of an optimization algorithm which is the Black Hole Algorithm (BHA) in solving several minimization problems. The three functions namely Ackley function, Beale function, and Booth function are used to find the minimum solution for each function.

This paper also demonstrated that BHA runs the three functions with different results according to the execution time and the optimum iteration. Booth function were being solved the fastest with less iterations needed. This is due to the nature of the function that does not possess any complex mathematical element. The source code was written in Python so that it can be executed and combined with existing source codes.

REFERENCES

[1] Farnad, B., Jafarian, A., & Baleanu, D. (2018). A new hybrid algorithm for continuous optimization problem. Applied Mathematical Modelling, 55, 652-673.

[2] Chen, W. (2015). Artificial bee colony algorithm for constrained possibilistic portfolio optimization problem. Physica A: Statistical Mechanics and its Applications, 429, 125-139.

[3] Juneja, M., & Nagar, S. K. (2016, October). Particle swarm optimization algorithm and its parameters: A review. In 2016 International Conference on Control, Computing, Communication and Materials (ICCCCM) (pp. 1-5). IEEE.

[4] Hussien, A. G., Hassanien, A. E., Houssein, E. H., Amin, M., & Azar, A. T. (2020). New binary whale optimization algorithm for discrete optimization problems. Engineering Optimization, 52(6), 945-959.

[5] Shao, W., Pi, D., & Shao, Z. (2016). A hybrid discrete optimization algorithm based on teaching–probabilistic learning mechanism for no-wait flow shop scheduling. Knowledge-Based Systems, 107, 219-234.

[6] Li, J., Sun, Y., & Hou, S. (2021). Particle swarm optimization algorithm with multiple phases for solving continuous optimization problems. Discrete Dynamics in Nature and Society, 2021.

[7] Chakri, A., Khelif, R., Benouaret, M., & Yang, X. S. (2017). New directional bat algorithm for continuous optimization problems. Expert Systems with Applications, 69, 159-175.

[8] Dokeroglu, T., Sevinc, E., Kucukyilmaz, T., & Cosar, A. (2019). A survey on new generation metaheuristic algorithms. Computers & Industrial Engineering, 137, 106040.

[9] Morales-Castañeda, B., Zaldivar, D., Cuevas, E., Fausto, F., & Rodríguez, A. (2020). A better balance in metaheuristic algorithms: Does it exist? Swarm and Evolutionary Computation, 54, 100671.

[10] Ezugwu, A. E., Adeleke, O. J., Akinyelu, A. A., & Viriri, S. (2020). A conceptual comparison of several metaheuristic algorithms on continuous optimisation problems. Neural Computing and Applications, 32(10), 6207-6251.

[11] Kumar, S., Datta, D., & Singh, S. K. (2015). Black hole algorithm and its applications. In Computational intelligence applications in modelling and control (pp. 147-170). Springer, Cham.

[12] Guzdial, M., & Ericson, B. (2016). Introduction to computing and programming in python. New York, NY: Pearson.

[13] Srinath, K. R. (2017). Python–the fastest growing programming language. International Research Journal of Engineering and Technology (IRJET), 4(12), 354-357.

[14] Beni, G. (2020). Swarm intelligence. Complex Social and Behavioural Systems: Game Theory and Agent-Based Models, 791-818.

[15] Khan, R., Amjad, M., & Srivastava, A. K. (2016, February). Optimization of automatic generated test cases for path testing using genetic algorithm. In 2016 Second International Conference on Computational Intelligence & Communication Technology (CICT) (pp. 32-36). IEEE.

[16] Mohi-Aldeen, S. M., Mohamad, R., & Deris, S. (2016). Application of Negative Selection Algorithm (NSA) for test data generation of path testing. Applied Soft Computing, 49, 1118-1128.

[17] Bao, X., Xiong, Z., Zhang, N., Qian, J., Wu, B., & Zhang, W. (2017). Path-oriented test cases generation based adaptive genetic algorithm. PloS one, 12(11), e0187471.

[18] Khari, M., Sinha, A., Verdu, E., & Crespo, R. G. (2020). Performance analysis of six meta-heuristic algorithms over automated test suite generation for path coverage-based optimization. Soft Computing, 24(12), 9143-9160.

[19] Madhavi, D. (2016). A White Box Testing Technique in Software Testing: Basis Path Testing. Journal for Research| Volume, 2(04).

[20] Cai, G., Su, Q., & Hu, Z. (2021). Automated test case generation for path coverage by using grey prediction evolution algorithm with improved scatter search strategy. Engineering Applications of Artificial Intelligence, 106, 104454.

[21] Cai, W., Yang, L., & Yu, Y. (2020, August). Solution of ackley function based on particle swarm optimization algorithm. In 2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA) (pp. 563-566). IEEE.

[22] Joung, I., Kim, J. Y., Gross, S. P., Joo, K., & Lee, J. (2018). Conformational Space Annealing explained: A general optimization algorithm, with diverse applications. Computer Physics Communications, 223, 28-33.

[23] Cai, W., Yang, L., & Yu, Y. (2020, August). Solution of ackley function based on particle swarm optimization algorithm. In 2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA) (pp. 563-566). IEEE.

[24] Wen, C., Xia, B., & Liu, X. (2017, August). Solution of second order ackley function based on SAPSO algorithm. In 2017 3rd IEEE International Conference on Control Science and Systems Engineering (ICCSSE) (pp. 624-627). IEEE.

[25] Izci, D., Ekinci, S., Orenc, S., & Demirören, A. (2020, October). Improved artificial electric field algorithm using nelder-mead simplex method for optimization problems. In 2020 4th International Symposium on

Multidisciplinary Studies and Innovative Technologies (ISMSIT) (pp. 1-5). IEEE.

[26] Neve, A. G., Kakandikar, G. M., & Kulkarni, O. (2017). Application of grasshopper optimization algorithm for constrained and unconstrained test functions. International Journal of Swarm Intelligence and Evolutionary Computation, 6(3), 1-7.