Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

Black Hole Algorithm in Solving Optimization Problems using Test Functions

Izrulfizal Saufihamizal Ibrahim1, Rosziati Ibrahim2, and Jahari Abdul Wahab3

1Phd Candidate, Department of Software Engineering, Universiti Tun Hussein Onn Malaysia, Johor, Malaysia

2Supervisor, Department of Software Engineering, Universiti Tun Hussein Onn Malaysia, Johor, Malaysia

3Industry 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. Optimization problems can be solved using traditional mathematics. In recent years, many researchers used metaheuristics algorithms such as honeybee algorithm to solve optimization problems more efficiently with less execution time. In this paper, an optimization algorithm called Black Hole Algorithm (BHA) will be used and applied to three continuous test functions which are Akley, Beale, and Booth. The test functions are used with difference sample size to get the optimum results for optimization problems. Based on the comparative analysis on the three functions, Beale function takes more iterations as compared with Ackley and Booth, and Booth function has less iterations needed with fastest execution time. Finally, the results based on the case studies show that BHA is capable of solving real-life problems involving optimization with great accuracy and short time execution.

INDEX TERMS Black Hole Algorithm, Metaheuristics, Test Functions, Optimization.

1. INTRODUCTION

Many real-world problems can be represented as difficult optimization problems, which is usually the case. Several optimization techniques have been developed to find near-optimal solutions, such as the meshless computational methods, which solve complex engineering problems without building the usually complex mesh points. [1] Optimization challenges are unavoidable, 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. Many scientific and engineering problems can be represented as optimization problems. One way to address these challenges is to develop customized solutions for specific problems. There are various ways to develop algorithms, but there may be a more efficient way to do it if the algorithms are generally applicable. Such algorithms are easy to apply, usually robust and reduce development time [2]. Optimization problems can now be tackled by utilizing a metaheuristics algorithm that produces a more efficient result. Half of the cost of completing a software project is spent on testing it, because it takes a lot of time and effort to do it properly. Test automation has reduced the time and cost of the manual testing process [3]. As a result, this condition encourages academics to improve existing metaheuristics or devise a new algorithm to tackle the problem of optimization more effectively. Solving combinatorial optimization problems often requires finding one or more optimal solutions within a specified solution space. Among other things, different approaches, including purely computational and artificial intelligence-based approaches, are adopted in the existing T-Way strategies

approaches nature-inspired algorithms such as Particle Swarm Optimization (PSO), Harmony Search Algorithm (HSA), and Bat Algorithm (BA) [4].

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 are the Ackley, Beale, and Booth functions. Black hole algorithm (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. The number of stars used for the study are 300, 600, and 900 with the same number of iterations of 150.

The rest of this paper is organized as follows. Section II looks at literature reviews and Section III discusses the related works. The results are discussed in Section IV and Section V covers the conclusion and the future work of the research.

1. LITERATURE REVIEWS

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.

1. OPTIMIZATION PROBLEM

In giving an answer for an issue, the likelihood of experiencing an issue that will require a various arrangement is unavoidable [5]. 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. The popularity of these tools acknowledges their versatility as design tools and their impeccable ability to find optimal solutions in complex multimodal search spaces. [6] To tackle this, a planning framework can be carried out [7]. 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. As we all know, application-oriented algorithms should be designed use as much problem-specific knowledge as possible to conduct algorithmic searches [8].

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 [9]. 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.

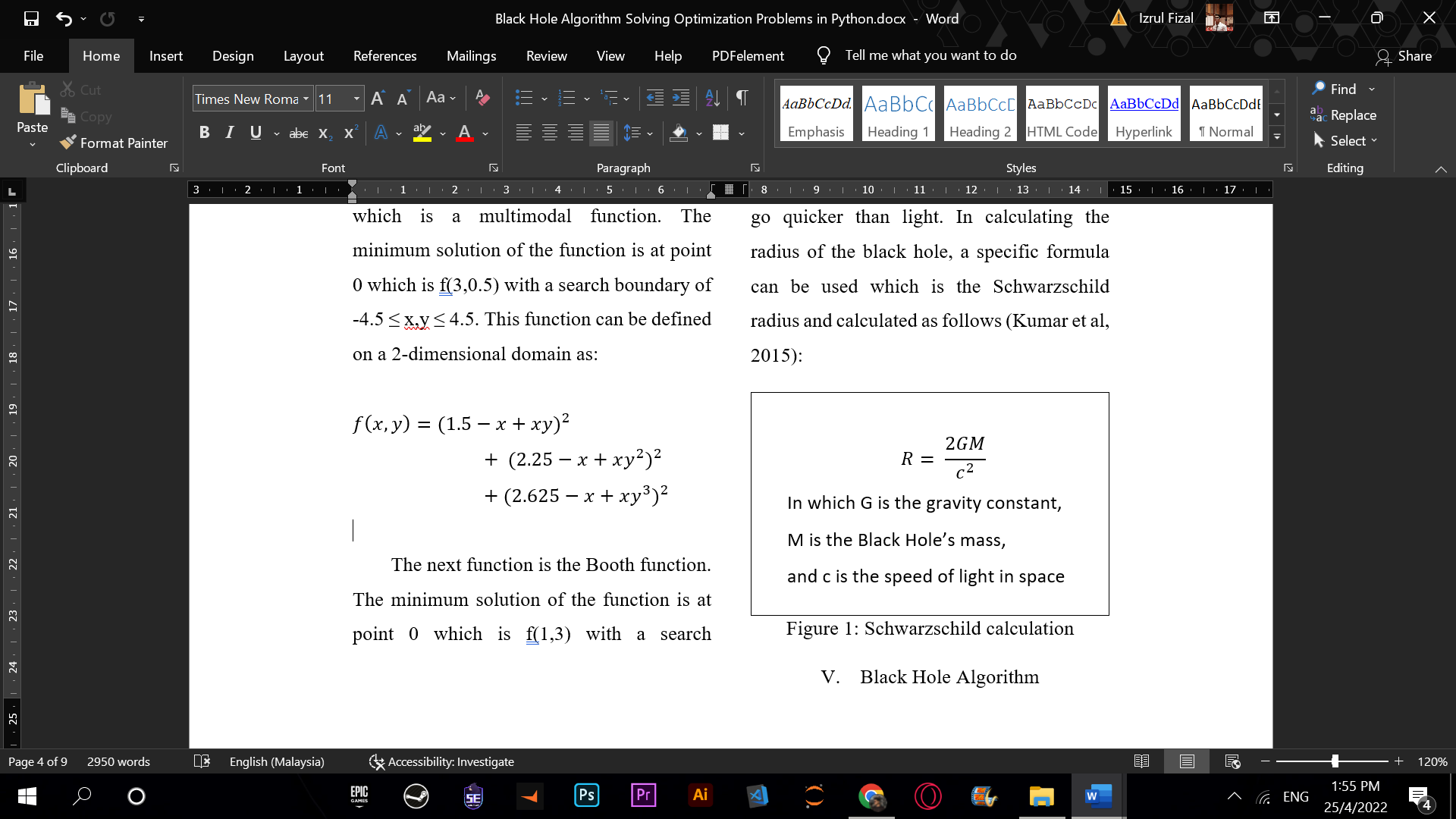
It is normal that the data for the given issue is known exactly. Regardless, for a few authentic issues, the data can't be known unequivocally for an arrangement of reasons. The chief explanation is a result of fundamental assessment screw up. The second and more significant clarification is that a couple of data address information about the future and can't be known with conviction [10]. 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 [11]. 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.

1. 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. Several binary metaheuristic algorithms have been developed that can be used to solve continuous problems while conserving the concepts of the search process. [11] 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 [12]. 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 [13].

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. In Path testing process of software testing, path testing is 50% effective at catching bugs during unit testing. Unstructured code is more effective than structured code when it comes to achieving the intended goal. All software organizations use this type of software release testing to get their software products into the market or to their customers [14]. 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.

1. 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. In software testing, it is applied by the challenge of generating test cases for software under test is to find a set of data that leads to the highest coverage when used as input [15]. 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 the early days of automated software testing, most test data generators used symbolic or dynamic techniques. Static symbolic test data generators assign symbolic values to variables in the program, rather than actual values. The dynamic test data generator, on the other hand, needs to actually run the program with some selected inputs. Even if there are desirable test requirements that have not been met, you can still make decisions using the data collected during the execution process. The closest data to meet the test requirements [16]. A research were done by using test functions in 2020 by Patcharin Kamsing, Peerapong, and Soemsak in testing the Filter-based Gradient Descent (PF-GD) optimizer [17]. The test functions are Rosenbrock, Booth, Beale, and Matyas functions.

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 ≤ x,y ≤ 5. This function can be defined on a 2-dimensional domain as:

(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 ≤ x,y ≤ 4.5. This function can be defined on a 2-dimensional domain as:

(2)

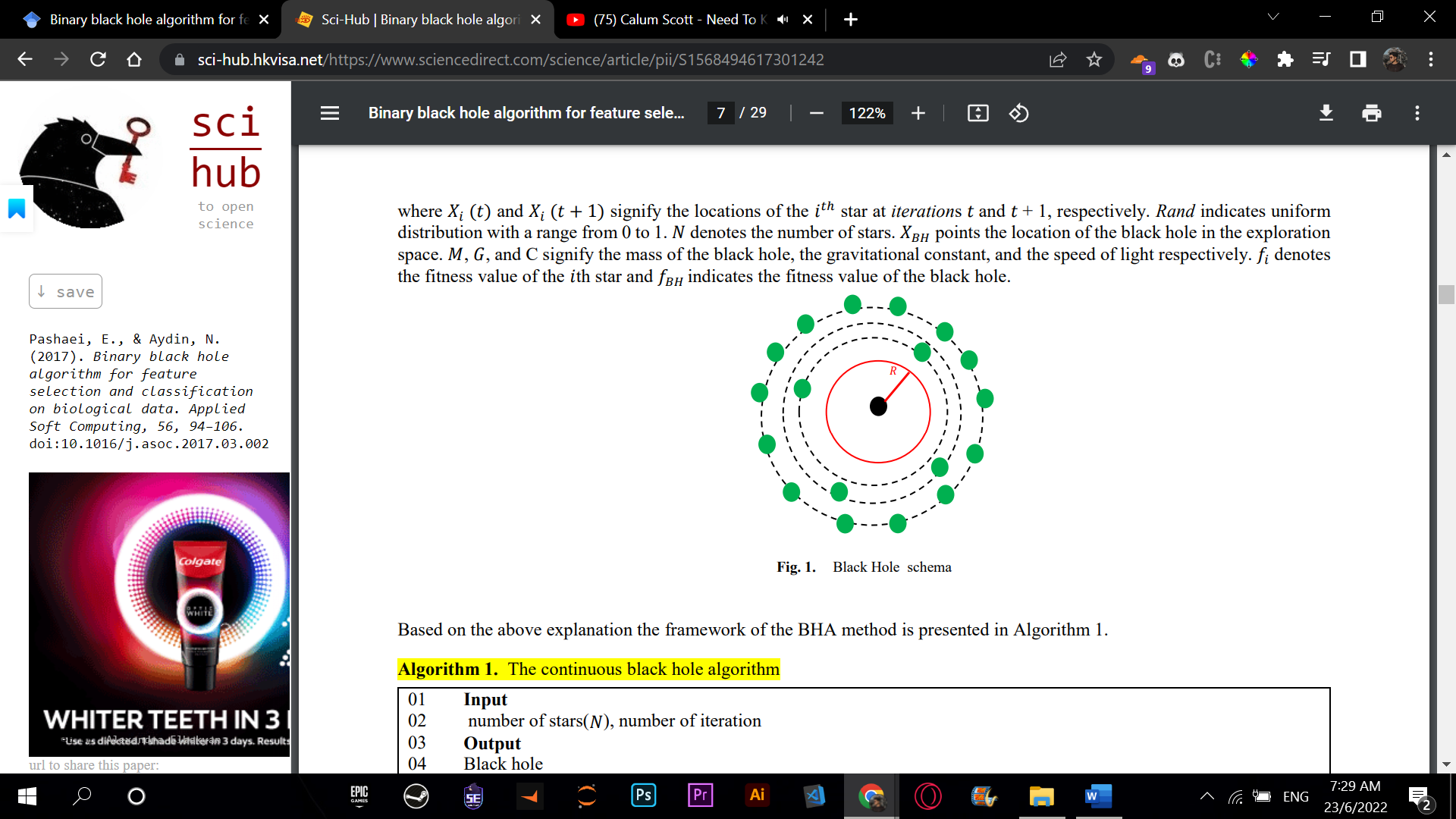
The next function is the Booth function. The minimum solution of the function is at point 0 which is f(1,3) with a search boundary of -10 ≤ x,y ≤ 10. This function can be defined on a 2-dimensional domain as:

(3)

1. BLACK HOLE PHENOMENA

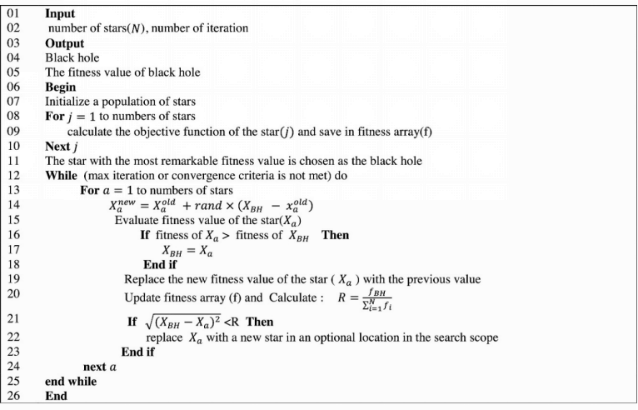
A black hole is an object that is being left when a star of a monstrous size finally collapsed. The pulling force of gravity inside the black hole is extensive that not even visible light can abscond from the black hole. Any object that traverses through the edge of the center will be gulped by it and 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 [18]. 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 [18]:

1. Schwarzschild calculation.



1. Black Hole Schema [19].
2. BLACK HOLE ALGORITHM

The Black Hole Algorithm (BHA) is a population-based strategy that has some normal highlights with other population-based strategies. BHA has been found to be very effective in a variety of applications [20]. 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 3.

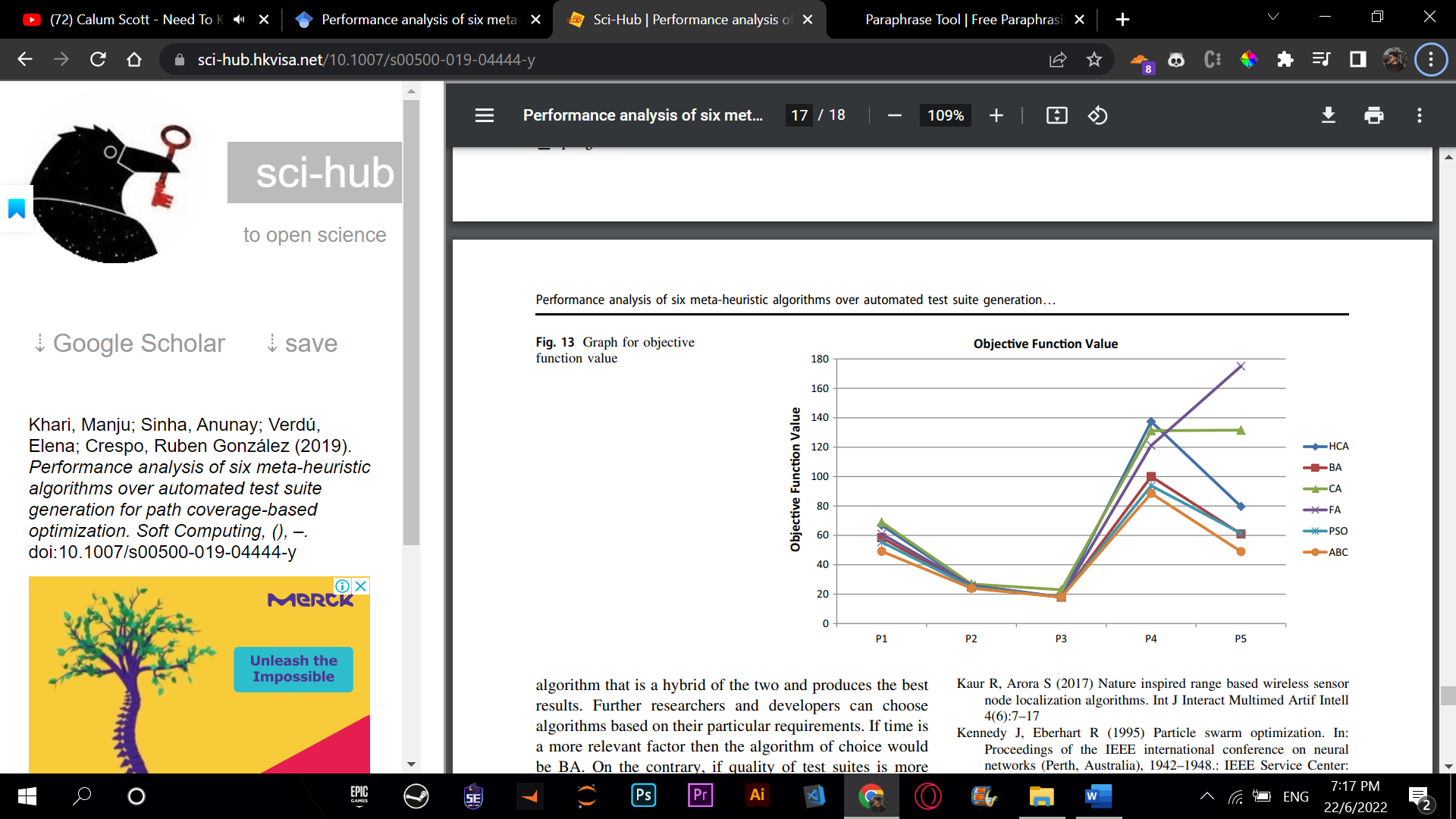
1. BHA Pseudocode [18].

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 [21] 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 [22].

ChangJun Wen, Bo Xia, and Xin Liu in 2017 [23] 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 [24] 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. On top of that, a study also made in 2020 by author Hamsa Naji and Dayang Jawawi on optimization involving benchmark functions using Multiple Black Hole Algorithm. The idea and concept of it is using multiple Black Holes which is NBH instead of using only one. The benchmark functions that were being used are the Parabolic, Ackley, Rastrigin, and Alpine [25].

In 2019, the authors Manju Khari, Anunay Sinha, Elena Verdu, and Ruben Gonza´lez Crespo did research on multiple metaheuristics algorithm to generate test suite for path coverage-based optimization [26]. The six metaheuristics methods that were used are Hill Climbing Algorithm (HCA), Particle Swarm Optimization (PSO), Firefly Algorithm (FA), Cuckoo Search Algorithm (CS), Bat Algorithm (BA), and Artificial Bee Colony (ABC). The summary of the results is shown in Figure 4.

COMPATIVE ANALYSIS ON TEST FUNCTIONS

1. Graph for objective function value [26].

In 2018, a study was undergone in comparing the effectiveness between Artificial Human Optimization Algorithm and Differential Evolution. The study also uses benchmark functions as the parameter for the comparison. The benchmark that were being used are Ackley, Bohachevsky, Booth, Three-Hump Camel and Beale benchmark functions. The result from the study shows that Particle Swarm Optimization works better in finding the solution than the improved algorithm HSPSO [27].

COMPARATIVE ANALYSIS ON TEST FUNCTIONS

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 sizes | Iteration | Expected solution |
| Ackley | 300  600  900 | 150 | [0,0] |
| Beale | 300  600  900 | 150 | [3,0.5] |
| Booth | 300  600  900 | 150 | [1,3] |

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.

1. Optimum Iteration Graph.

As for the analysis, it can be seen from Figure 5, 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 6 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.

1. Execution Time Graph.

REAL-LIFE PROBLEMS AS CASE STUDIES

After the benchmark functions tests have been done, and the results have been noted, it is now crucial to test BHA on real-life problems involving optimization. There are everyday problems that can be solve using optimization techniques whether it is involving finding the most optimum value which is the maximum or the minimum values. Three case studies have been identified and will be solve using BHA.

The first case study is profit maximization. The problem is that supposed that a seller is selling an item for a price, how can the seller set the price for an item to get the maximum number of profits for the items sold. In this case study, the number of items to be sell is denoted with n, and P(n) is the profit supposed that the seller has a fixed modal of RM4,000, and the production cost, y, for each item is RM1.00. Now a maximization problem has presented itself.

The next case study is the capacity maximization of a production facility. Supposed that a company is operating for 5 months straight non-stop in producing the products, how can the maximum rate of operation be determined? For this case study, we have to take into consideration the input for this case which is the time as t and the amount of time needed to produce one product, s. The maximization problem that will be solve is the value for f[0,150] which is the 150 days max number of days.

The third case study involved in finding the largest possible size of an area of a rectangular shape for a fixed length of fence. We denote length of the fence that is currently available as l is 200m, and the area of the fenced property will be A(x,y) which is xy which is the area of a rectangle. How can we determined the maximum area of the property covered by the fence using BHA? The three case studies will be applied to BHA to be solved and will also be consider the execution time for each problems to be added to the overall analysis.

RESULTS AND DISCUSSION

The results from the case studies applied to BHA is shown as a summary in Table 5.

TABLE 4

Result For Case Studies

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Case Study | Input | Expected Outcome | Actual Outcome | Time |
| Profit Maximization | P[5000,1] | 3000 | 3000 | 1.1s |
| Capacity Maximization | F[0,150] | 98.33 | 98.33 | 1.2s |
| Maximum Area Determination | A[200,50] | 5000 | 5000 | 1.1s |

From the result, it is shown that BHA is capable of solving real life optimization problems with great accuracy. Furthermore, the results also include the execution time for the whole process which is all of them possessed a great result of less than 2 second’s time. The only main difference between the three is the Capacity Maximization problem which takes a little more time probably due to its complexity. However, it is not that significant as the difference between the other two problems differs only 0.1s.

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. Python is also known to have an abundance of libraries that assist with data analysis and scientific computing. Python is also known to have a wealth of libraries, supports data analysis and scientific calculations [28].

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.

REFERENCES

1. 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
2. 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.
3. 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.
4. Nsaif, H. N., & Jawawi, D. N. A. (2020, May). Binary Black Hole-Based Optimization For T-Way Testing. In IOP Conference Series: Materials Science and Engineering (Vol. 864, No. 1, p. 012073). IOP Publishing.
5. Farnad, B., Jafarian, A., & Baleanu, D. (2018). A new hybrid algorithm for continuous optimization problem. Applied Mathematical Modelling, 55, 652-673.
6. 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.
7. Chen, W. (2015). Artificial bee colony algorithm for constrained possibilistic portfolio optimization problem. Physica A: Statistical Mechanics and its Applications, 429, 125-139.
8. 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.
9. 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.
10. 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.
11. 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.
12. Dokeroglu, T., Sevinc, E., Kucukyilmaz, T., & Cosar, A. (2019). A survey on new generation metaheuristic algorithms. Computers & Industrial Engineering, 137, 106040.
13. 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.
14. Madhavi, D. (2016). A White Box Testing Technique in Software Testing: Basis Path Testing. Journal for Research| Volume, 2(04).
15. 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.
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. Kamsing, P., Torteeka, P., & Yooyen, S. (2020). An enhanced learning algorithm with a particle filter-based gradient descent optimizer method. Neural Computing and Applications, 32(16), 12789-12800.
18. 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.
19. Pashaei, E., & Aydin, N. (2017). Binary black hole algorithm for feature selection and classification on biological data. Applied Soft Computing, 56, 94-106.
20. Abualigah, L., Elaziz, M. A., Sumari, P., Khasawneh, A. M., Alshinwan, M., Mirjalili, S., ... & Gandomi, A. H. (2022). Black hole algorithm: A comprehensive survey. Applied Intelligence, 1-24.
21. 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.
22. 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.
23. 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.
24. 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.
25. Al-Sammarraie, H. N. N., & Jawawi, D. N. (2020). Multiple black hole inspired meta-heuristic searching optimization for combinatorial testing. IEEE Access, 8, 33406-33418.
26. 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.
27. Gajawada, S., & Mustafa, H. M. (2018). Testing Multiple Strategy Human Optimization based Artificial Human Optimization Algorithms. Computer Reviews Journal, 1(2).
28. Srinath, K. R. (2017). Python–the fastest growing programming language. International Research Journal of Engineering and Technology (IRJET), 4(12), 354-357.
29. Hatamlou, A. (2018). Solving travelling salesman problem using black hole algorithm. Soft Computing, 22(24), 8167-8175.
30. Xie, W., Wang, J. S., & Tao, Y. (2019). Improved black hole algorithm based on golden sine operator and levy flight operator. IEEE Access, 7, 161459-161486.
31. Kumar, D., Gandhi, B. G., & Bhattacharjya, R. K. (2020). Firefly algorithm and its applications in engineering optimization. In Nature-Inspired Methods for Metaheuristics Optimization (pp. 93-103). Springer, Cham.
32. Munoz, R., Olivares, R., Taramasco, C., Villarroel, R., Soto, R., Barcelos, T. S., ... & Alonso-Sánchez, M. F. (2018). Using black hole algorithm to improve eeg-based emotion recognition. Computational intelligence and neuroscience, 2018.
33. García, J., Crawford, B., Soto, R., & García, P. (2017, February). A multi dynamic binary black hole algorithm applied to set covering problem. In International Conference on Harmony Search Algorithm (pp. 42-51). Springer, Singapore.