

## Sensitivity of Nature Inspired Meta-Heuristic Algorithms to Parameter Settings and Development of Algorithms Look-Up Table

Raihanatu Mohammed Hamid<sup>1</sup>, Hama Sabo<sup>1</sup>, Fatsuma Jauro<sup>2</sup>, Ali Muhammad Usman<sup>3</sup>, Haruna Chiroma<sup>4</sup>

<sup>1</sup>Department of Mathematical Sciences Abubakar Tafawa Balewa University Bauchi, Nigeria.

<sup>2</sup>Department of Computer Science Ahmadu Bello University Zaria, Nigeria

<sup>3</sup>Computer Science Department Federal College of Education (Technical) Gombe, Nigeria

<sup>4</sup>College of Computer Science and Engineering, University of Hafr Al-Batin, Saudi Arabia

### ABSTRACT

Numerous studies have been conducted on nature-inspired metaheuristics, with a particular focus on hybridization, performance evaluation, and their application to solving optimization problems. However, one persistent challenge is the laborious process of selecting optimal parameter settings for these algorithms, lacking a systematic approach for easily determining suitable values. In this research, we explore the sensitivity of nature-inspired metaheuristic algorithms to changes in their parameter settings. We aim to develop a convenient look-up table containing recommended parameter combinations for ten popular algorithms in this field. These algorithms include Artificial Bee Colony (ABC), Bat Algorithm, Cuckoo Search Optimization (CSO), Firefly Algorithm (FFA), Flower Pollination Algorithm (FPA), Genetic Algorithm (GA), Hybrid Bat Algorithm, Mothflame Optimization (MFO), Particle Swarm Optimization (PSO), and Simulated Annealing (SA). The look-up table presents various parameter combinations along with corresponding solution quality and convergence speed, specifically in relation to global solutions. Among these algorithms, FPA demonstrates superior convergence by achieving the global solution in seven benchmark problems. FFA, ABC, CSO, and PSO also produce highly favorable results, closely approaching the global solution. In terms of runtime efficiency, CSO outperforms the other algorithms. By developing this comprehensive look-up table for nature-inspired metaheuristic algorithms, we contribute a valuable resource to the existing literature. This table offers an additional suggested approach for researchers to easily select optimal combinations of parameter settings. We believe that our work will greatly benefit the research community by simplifying the process of choosing appropriate parameter values for these algorithms.

### ARTICLE INFO

#### Article History

Received: March, 2023

Received in revised form: April, 2023

Accepted: June, 2023

Published online: June, 2023

### KEYWORDS

Nature-Inspired Metaheuristic, Parameter Combination, Parameter Settings, Lookup Table.

### INTRODUCTION

Optimization is applicable in our everyday aspect of life, from engineering to business, household to routine of internet (Agarwal and Mehta 2014). In almost all aspects of life endeavours we are trying to maximize profit or consider optimum solution, quality or in some instance time (Yang and Papa 2016)S. As valuable resources such as time,

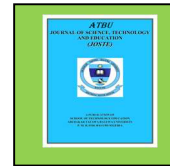
resources and money are limited in catering for our daily life applications, we need to come up with ways to use these resources under various constraints. These days' optimizations problems are indispensable to solve using computer simulations with efficient search algorithms (Yang 2010). Search algorithm can be deterministic or stochastic. Deterministic algorithms called gradients based

Corresponding author: Raihanatu Mohammed Hamid

✉ [raihanatumohdhamid@gmail.com](mailto:raihanatumohdhamid@gmail.com)

Department of Mathematical Sciences, Faculty of Science, Abubakar Tafawa Balewa University, Bauchi.

© 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved



because they use gradient information and are perfect for unimodal problems. They perform poorly if there is little discontinuity in the objective functions. In situation like this a non-free gradient is used which does not use derivatives but use only function values. However, stochastic algorithm is of two types namely; the heuristics and the metaheuristic, which have, slight difference. The term heuristics mean “to discover by trial and error” (Yang 2010). In heuristics, algorithm there is no assurance that optimum solution will be determined, thus is time consuming and good only in instances where best solutions are not necessary required.

The higher level of stochastic algorithm is the metaheuristic algorithm, which performs better than heuristics. Metaheuristics comprises of two major components which are; exploration (diversification) and exploitation (intensification) (Yang and He 2016). Diversification is to creates so many solutions so as to tour the search space on a global space, on the other hand intensification means focusing within the local region so that the best solution is exploited within the given region (Yang 2010) . These two components of metaheuristics algorithms are combined to ensure that optimum solution is achieved (Salgotra, Singh et al. 2018).

Nature inspired metaheuristic algorithms are used purposely to get best, feasible solutions within the shortest possible time (Yang and Papa 2016). While choosing an algorithm, there is no guarantee as to whether the algorithm you choose will yield the desired results or not. How it works, there is no certainty that the best solution can be derived, so the need to practical apply them is required to actually ascertain the algorithm that will produce an efficient good quality results, and others that will produce nearly optimum and even those that

## PREVIOUS WORKS

This section reviews papers that compare nature inspired metaheuristics algorithms, such as particle swarm optimization and genetic algorithms, as well as papers that deals with parameter control. In the end, we mention papers that use similar benchmark functions; (Janah and Fujimoto 2018) compared the performance of firefly algorithm (FFA) and particle swarm optimization algorithm (PSO) in solving simultaneous localization and

cannot be applied for a particular problem domain (Yang 2014).

Many studies on metaheuristics algorithms have been carried out on analysing the nature inspired metaheuristics algorithms (Ezugwu, Shukla et al. 2021).

The performance of the meta-heuristic algorithms heavily relies on the parameter settings. The standard approach to realize the best parameters settings remain an open research problem to the research community (Yang 2014, Chiroma, Herawan et al. 2017, Ezugwu, Shukla et al. 2021). In metaheuristics algorithms, setting of parameter has a vital role in defining their behaviour, protecting the search and biasing the effectiveness of the results. Appropriate parameter setting may be altered during optimization process making the task tougher (Parpinelli, Plichoski et al. 2019).

The approaches already discussed in the literature such as trial and error approach through initial experimentation and adopting the parameters have the following limitations: The trial and error approach is time consuming, laborious and prompt to errors. Adopting the parameters from the literature cannot work for different domains, therefore, making balancing exploitation and exploration to be difficult. However, study on the sensitivity of the algorithms across multiple meta-heuristic algorithms on different benchmark functions to develop lookup table for the parameters is scares in the literature.

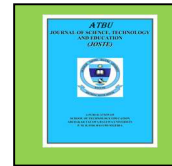
To avoid the limitations of the trial-and-error approach through initial experimentation, challenges of the adoption approach and ease the work of researchers in selecting the optimum parameter settings for the algorithms, this research intent to investigate the sensitivity of the meta-heuristic algorithms and develop a loop-up table. mapping (SLAM) problems. The FFA and PSO were applied to solve in SLAM problems. The result was found that, PSO is better than FFA. However, FFA was slow. (Rathasamuth and Nootyaskool 2016) compare particle swarm optimization (PSO), genetic algorithm (GA) and flower pollination optimization (FPA) in solving discrete space problem. The main objective is to examine the performance of the algorithms on discrete space and calculate velocity of PSO as well as Lévy flight movement of FPA when using discrete parameter. Experiment result showed

Corresponding author: Raihanatu Mohammed Hamid

✉ [raihanatumohdhamid@gmail.com](mailto:raihanatumohdhamid@gmail.com)

Department of Mathematical Sciences, Faculty of Science, Abubakar Tafawa Balewa University, Bauchi.

© 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved



that FPA could perform better than PSO and GA. (Korkmaz and Akgüngör 2018) compare the performance of artificial bee colony (ABC) and flower pollination algorithm (FPA) in finding optimal approach to vehicle delay models at signalized intersections. ABC and FPA are approaches used in developing different delay models. The results show that FPA approach out performs ABC approach in finding optimum solution. (Kaur and Agrawal 2017) comparative study of Bat and Cuckoo Search Algorithm for Regression Test Case Selection. The performance of BA and CS where evaluated for selecting test cases. Results indicates that the Cuckoo Search Algorithms perform a little better than Bat Algorithm. However, CSA is slow in terms of running time when compared to BA. (Ma, Bi et al. 2016) compare the performance on photovoltaic model parameter identification using six bio-inspired optimization algorithms, i.e. genetic algorithm, differential evolution, particle swarm optimization, bacteria foraging algorithm, artificial bee colony, and cuckoo search. GA, DE, PSO, BFA, ABC and CS are compared statistically by testing over single-diode models to evaluate their performance in terms of accuracy and stability under uniform solar irradiance and various environmental conditions. Results indicate that cuckoo search algorithm is more robust and precise among these bio-inspired optimization algorithms, the paper shows that bio-inspected algorithms are capable of improving the existing PV models by using optimized parameters. (Oral and Turgut 2018) compare the performances of Flower Pollination Algorithm (FPA), Forest Optimization Algorithm (FOA) and Artificial Algae Algorithm (AAA). FPA, FOA and AAA where compared under 10 well known benchmark to solve optimization problems. The results revealed that the optimization performance of FPA is superior to FOA and AAA. (Sapra, Sharma et al. 2017) compare three metaheuristic algorithms applied to solve Knapsack Problem. Tabu Search algorithm (TSA), Scatter Search algorithm (SSA) and Local Search algorithm (LSA) are applied to the knapsack problem. The results show that SSA have least execution time than TSA and LSA. (Hasnan, Elewe et al. 2017) compare the performance of two metaheuristics algorithm in solving Radio frequency identification network planning optimization problem. Firefly algorithm (FA)

and Multi-Colony Global Particle Swarm Optimization (MC-GPSO) are used in solving large-scale RFID Network Planning optimization problems. The results show that FFA has better optimization quality of solving NP hard problems than PSO. (Oral and Turgut 2018) compare the performance of three nature-inspired algorithm under hard restriction. Particle Swarm Optimization (PSO), Artificial Bee Colony (ABC) and Grey Wolf Optimizer (GWO) are analysed under severe hard restriction to solve complex optimization problems. The test results revealed that ABC is the best in all scenarios compared to PSO and GWO. (Agarwal and Mehta 2018) compare the performance evaluation of five metaheuristics algorithm to solve real parameter optimization problems. It accesses the efficiency of bat algorithm (BA), artificial bee colony algorithm (ABC), cuckoo search algorithm (CS), firefly algorithm (FA) and flower pollination algorithm (FPA) benchmark functions. The study reveals that ABC algorithm perform best for majority of the problems on high dimension, CS is best on small dimension, followed by FPA, then BA and FA for all kinds of benchmarks function. (Singh and Walia 2017) compare the performance evaluation of four popular nature-inspired optimization algorithms: particle swarm optimization (PSO), fire fly algorithm (FA), cuckoo search (CS) and bacterial foraging algorithm for the design of bored pile foundation by artificial neural networks. Statistical analysis of PSO, FA, CS and BFA are used in solving the three dimensional optimization problem. The outcome of the performance evaluations shows that PSO is the best for solving the problems. (Luthra, Chaturvedi et al. 2017) The study aimed at comparing the performance evaluation of four metaheuristics algorithms: Bat algorithm (BA), genetic algorithm (GA), artificial bee colony algorithm (ABC) and ant colony optimization (ACO) algorithms. BA, GA, ABC and ACO are evaluated to know which one is the best in terms of efficiency, accuracy, performance, reliability and computation time to solve optimization problem. The results show that ABC optimizations outperforms all the other three algorithms.

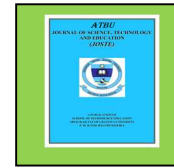
Bertolini, Mezzogori et al. (2019) compare the performance of new metaheuristics to solve integrated job-maintenance scheduling problem. Harmony Search for Integrated Job Maintenance

Corresponding author: Raihanatu Mohammed Hamid

✉ [raihanatumohdhamid@gmail.com](mailto:raihanatumohdhamid@gmail.com)

Department of Mathematical Sciences, Faculty of Science, Abubakar Tafawa Balewa University, Bauchi.

© 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved



Scheduling (HS/4/IJMS), cuckoo search with levy flight (CS-LF), simulated annealing (SA), cuckoo search with migration nest memory (CS-MNM) algorithms where compared. Results show that CS-MNM proves to be more superior to the modified harmony search, SA and CS-LF in terms of short computational time. (Chassiakos and Rempis 2019) compare the performance evaluation of five evolutionary-based metaheuristics algorithms with their enhanced formulations in solving the time-cost trade-off problem in project time cost optimization. Hill climbing (HC), custom version genetic algorithm (CGA), saw tooth genetic algorithm (STGA), micro genetic algorithm ( $\mu$ GA) standard particle swarm optimization (SPSO), enhanced particle swarm optimization (EPSO), differential evolution (DE), and artificial bee colony (ABC) algorithms. The results indicate a slight outperformance of EPSO followed by DE and  $\mu$ GA to be the better than other algorithms in converging faster to the solution quality as well as computational time.

(Akbari and Henteh 2019) compare the performance evaluation of two metaheuristics algorithms to solve discrete and continuous size optimization problem of 2D truss structure. Genetic algorithm (GA) and particle swarm optimization (PSO) are used for solving truss optimization. The results indicate that GA is generally the best economically for the discrete size cases and PSO is economically the best for continuous size cases. (Saeidian, Mesgari et al. 2016) compare the performance evaluation of two metaheuristics optimization in algorithms in locating the location and allocation earthquake relief centres. Genetic algorithm (GA) and bee algorithm (BA) are applied in finding optimum location and allocating parcels to reliefs centre. The outcome of the research shows that GA is faster than BA. (Sachan, Gupta et al. 2018) compare the performance evaluation of seven metaheuristics algorithms to solve optimization problem. Particle swarm optimization (PSO), differential evolution (DE), shuffled frog leaping algorithm (SFLA), artificial bee colony (ABC), firefly algorithm (FA), teacher learning based optimization (TLBO) and jaya Algorithm to solve optimization problem. Results of the study shows that TLBO performs better than others six algorithms in terms of mean result, standard deviation and best result and

have faster convergence speed. (Mohseni, Brent et al. 2020) compare the performance evaluation of six metaheuristics algorithms for the optimal capacity planning of an isolated, battery less hydrogen-based micro-grid. Mothflame optimization(MFO), dragon fly algorithm(DA), salp swarm algorithm(SS), ant lion optimizer (ALO), grey wolf optimizer (GWO) and grasshopper optimization algorithm(GO) are applied in finding the optimal capacity of hydrogen-based micro-grid. The output of the evaluation reveals that Mothflame algorithm is better than five algorithms in respect of the application area. (Ebrahimi, Jeon et al. 2020)compare the performance evaluation of four new hybrid optimization algorithms to solve a scheduling layout problem in a flexible shop job system. Particle swarm optimization-dynamic learning strategy (PSO-DLS), particle swarm optimization-simulated annealing (PSO-SA), ant colony optimization-local search strategy (ACO-LSS) and ant colony optimization-simulated annealing (ACO-SA) algorithms are applied in solving the scheduling problem. Results of the study indicates that hybrid ant colony optimization and simulated annealing algorithm performs better than the three new hybrid algorithm in respect of the problem domain.

(Moya, Chica et al. 2019) compare the performance of selected metaheuristics in solving new robust car sequencing problem under uncertain partial demand. Greedy randomized adaptive search procedure (GRASP), ant colony optimization (ACO), variable neighborhood sear (VNS) and the memetic ant colony optimization (ACO-LS). Results indicate that ACO-LS is the best because it performs well regardless of the difficulty of the problem. (Kaveh, Izadifard et al. 2020) compare the performance of three metaheuristic algorithms to optimally minimize carbon dioxide ( $\text{CO}_2$ ) emission and construction cost in design of reinforced concrete frames. Enhanced colliding bodies optimization (ECBO), enhanced vibrating particles system (EVPS) and particle swarm optimization (PSO) are applied in optimizing the construction cost and  $\text{CO}_2$  emission. The outcome of the comparison shows that ECBO has higher tendency of obtaining best solution. (Leal-Naranjo, Soria-Alcaraz et al. 2019) compare the performance evaluation of three metaheuristic algorithms in optimizing the dimensional synthesis of a spherical

Corresponding author: Raihanatu Mohammed Hamid

✉ [raihanatumohdhamid@gmail.com](mailto:raihanatumohdhamid@gmail.com)

Department of Mathematical Sciences, Faculty of Science, Abubakar Tafawa Balewa University, Bauchi.

© 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved

parallel manipulator. Non-dominated sorting genetic algorithm(NSGA-II) multi objective particle swarm optimization algorithm(MOPSO), multi objective evolutionary algorithm based on decomposition(MOEA/D) are applied solving dimensional synthesis problem of a spherical parallel manipulator. Outcome of the research shows that MOEA/D algorithm outperforms NSGA-II, MOPSO.

In this paper, ten nature inspired metaheuristics algorithms are subjected to parameters settings to investigate their sensitivity across twenty (20) benchmark functions, and the study developed a lookup table for nature inspired meta-heuristic algorithms parameter control settings for easing the work of the research community.

The paper is organized as follows. Related work is described in Section 2. The concept of the selected NIMA is briefly presented in Section 3. Important basic concept of parameter settings and sensitivity are highlighted in section 4. The methodology of the paper is described in Section 5 including the benchmark functions. In Section 6, experiments results and the discussion of the research is described in details. Conclusion and remark are finally given in Section 7.

### SELECTION OF NATURE INSPIRED METAHEURISTICS ALGORITHMS

#### Artificial Bee Colony Algorithm

Artificial Bee Colony is a swarm based artificial algorithm with three groups of bees; namely employed bees, onlookers' bee and scouts bee. The scouts are responsible for randomly searching or discovering of food source, the onlooker bee is responsible for waiting to make a decision on the food source to choose and the employed bee is responsible for going to the food source. Each search cycle comprises of three steps; First sending the employed bees to the food sources and determining its nectar amounts, second selecting the food sources by the onlooker bee after communicating the information including the nectar amount of the foods by the employed bee and lastly sending them to possible food source. Position of food source is represented by the solution of the optimization problem and amount of food source nectar equals the quality (fitness) of the related solution. Either the number of employed or the onlooker bees is

equivalent to the number of solutions in the population. Randomly distributed initial population  $P(G = 0)$  of SN solution is generated, where SN is the population size. Each solution  $x_i = (i = 1, 2, \dots, SN)$  is a  $D$ -dimensional vector, where  $D$ =number of optimization parameter. After initialization, the population of the positions can be repeated in cycles,  $C = 1, 2, \dots, C_{max}$  of the search procedure of employed, the onlooker and scout bees. Depending on the probability value in relation to the food source selected by the onlooker bee,

$$P_i = \frac{fit_i}{\sum_{i=1}^{SN} fit_n}, \quad (1)$$

Where  $fit_i$  is the fitness value of the solution,  $i$  calculated by its employed bee proportional to the nectar amount of the position of the food source and SN is the population of source and employed bee. To produce a candidate food position from the old one, ABC uses the below expression

$$V_{ij} = X_{ij} + \varphi_{ij}(X_{ij} - X_{kj}), \quad (2)$$

Where  $k \in \{1, 2, \dots, BN\}$  and  $j \in$  are indexes chosen at random, and  $k$  is determined randomly,  $\varphi_{ij}$  is the random number between  $[-1, 1]$

(3.2) shows that as the difference between the parameters of the  $x_{ij}$  and  $x_{kj}$  decreases, the perturbation on the position  $X_{ij}$  decreases. In ABC algorithm, if the position of food source can't be improved via an expected number of cycles called *limit* then the food source is anticipated to be abandoned (Karaboga and Basturk 2007)

#### Bat Algorithm

Bat algorithm was developed by Xin-She Yang (Yang 2011), which imitates the behavior of microbats on the basis of echolocation. This echolocation is necessary for frequency tuning. A loud sound pulse is emitted by this bats with an echo that bounces back from the objects that surrounds the sound. With regards to bat algorithm, bat flies in search of a prey at random velocity  $v_i$  at position  $x_i$  have a fixed frequency rate  $[f_{min}, f_{max}]$ , with varying emission rate  $r \in (0, 1)$  and loudness  $A_0$  to hunt for a prey depending on the closeness of the prey. The updated equations are as follows:

$$f_i = f_{min} + (f_{max} - f_{min})\epsilon, \quad (3)$$

$$v_i^{t+1} = v_i^t + (x_i^t - x^*)f_i \quad (4)$$

Corresponding author: Raihanatu Mohammed Hamid

✉ [raihanatumohdhamid@gmail.com](mailto:raihanatumohdhamid@gmail.com)

Department of Mathematical Sciences, Faculty of Science, Abubakar Tafawa Balewa University, Bauchi.

© 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved

Where  $\varepsilon$  = random number drawn from a uniform distribution

$x^*$  = current best solution found during iterations.

The loudness and pulse rate can differ with iteration  $t$  in this approach

$$A_i^{t+1} = \alpha A_i^t, \quad r_i^t = r_i^0 [1 - \exp(-\beta t)] \quad (5)$$

At this time,  $\alpha$  and  $\beta$  are constants, we can simply use  $\alpha = \beta = 0.9$  for most simulations (Yang 2011)

### Cuckoo Search Optimization Algorithm

Cuckoo search algorithm (CS) was created by (Yang and Deb 2009) and was improved by Levy flight (Pavlyukevich 2007) based on the brood parasitism of some cuckoo species instead of simple isotropic random walks. Cuckoos are fascinating birds, which can make beautiful sound, and because of their aggressive reproduction strategy, some species such as the ani and Guira cuckoos lay their eggs in communal nests, though they may remove others' eggs to increase the hatching probability of their own eggs. Quite a number of species engage the obligate brood parasitism by laying their eggs in the nests of other host birds

CSA algorithm deploy a balanced combination of local random walk and the global explorative random walk that is regulated by a switching parameter  $P_a$ . Local random walk is evaluated using:

$$X_i^{t+1} = X_i^t + \alpha s H(P_a - \varepsilon)(X_j^t - X_k^t) \quad (6)$$

Here,  $X_j^t$  and  $X_k^t$  are two solutions selected differently and randomly by random permutation,

$H(u)$  is Heaviside function  $\varepsilon$  = random number derived from uniform distribution and step sizes are extracted from levy distribution. However, global random walk is conducted using levy flights:

$$X_i^{t+1} = X_i^t + \alpha L(s, \lambda), \quad (7)$$

Where

$$L(s, \lambda) = \frac{\lambda \Gamma(\lambda) \sin(\pi\lambda/2)}{\pi} \frac{1}{s^{1+\lambda}}, \quad (s \gg s_0 > 0). \quad (8)$$

$\alpha > 0$  factor used in scaling which control scale of the step sizes and  $S_0$  fixed step size (small).

To improve the performance of an algorithm, quite a fraction of the new solutions is expected to be generated by far-field randomization, and should be far away from the current best solution so that local optimum will not trap the system. CSA can be best applied in solving engineering optimization problems (Yang and Deb 2009).

### Firefly Algorithm

Firefly Algorithm (FFA) was developed based on mimicking the flashing pattern and behavior of fireflies. Xin She Yang originates it in 2007. Movement of firefly  $i$ , which is attracted to another brighter firefly,  $j$  is evaluated by the nonlinear equation updated below:

$$X_i^{t+1} = X_i^t + \beta_0 e^{-\gamma r^{2ij}} (X_j^t - X_i^t) + \alpha \varepsilon_i^t, \quad (9)$$

Where,  $\beta_0$  = attractiveness at distance  $r = 0$ .

The attractiveness of a firefly differs because of distance from other fireflies. The difference in attractiveness  $\beta$  with respect to the distance  $r$  can be evaluated by

$$\beta = \beta_0 e^{-\gamma r^2} \quad (10)$$

(Yang 2010)

### Flower Pollination Algorithm

Flower Pollination Algorithm (FPA) imitates pollination features of flowering plants. Xin – She Yang in 2014 originated it. The most applied algorithm equation is applied based on life cross-pollination, flower constancy abiotic self-pollination. Thus, the equation can be expressed as follows:

$$X_i^{t+1} = X_i^t + \gamma L(s, \lambda)(g_* - X_i^t), \quad (11)$$

Where,  $X_i^t$  = pollen  $i$  or solution vector  $X_i$  at iteration  $t$ ,  $g_*$  is the current best solution gotten all solution.  $\gamma$  is a scale factor that control step size,  $L(s, \lambda)$  is the randomization derived from the equation. The equation above is a global search step; it can be complimented by an intensive local search below:

$$X_i^{t+1} = X_i^t + \epsilon (X_j^t - X_k^t), \quad (12)$$

where,  $X_i^t$ ,  $X_j^t$ ,  $X_k^t$  are the three solution vectors at iteration  $t$ , each vector solution represent different flowers belonging to the same plant species,  $\epsilon$  is a random number which is uniformly distributed, between two search equations, there exists proximity probability  $P$  that regulates the search equation to be taken. Ideally,  $p = 0.5$  can be used, but  $p = 0.8$  looks more factual and important in reality. Yang reiterates that FPA is good in solving multi objectives optimization problems (Yang 2014).

### Genetic Algorithm

Genetic algorithm is based on the Darwin's evolution of biological systems initiated by J. Holland

and his team members in 1960s and 1970s (Holland 1992). GA uses the genetic operators of mutation, crossover and selection. Every solution is encoded as a string (mostly decimal and binary) called a chromosome. When two parent strings are crossed, they produce an offspring (new solution) by transferring part or genes of the chromosomes. A typical crossover has a higher probability of 0.8 - 0.95. Crossover provides an opportunity of producing new solutions where the main characteristics of the new solution will be passed on to newer solutions when the parents solution are mixed, thus providing good convergence under the right condition. In addition, mutation takes place through tossing some digits if a string, there by generating new solutions. It probability is usually low which ranges from 0.0001 to 0.05. Mutation has an advantage of over crossover through generating new solution outside the solution subspace thus escaping the local optima. Selection provides an avenue for a good solution to be selected, solution fittest is the criteria for solution to be selected. In some occasion, to maintain the best solution in the population, those solutions are passed onto the next generation without being altered. This is termed elitism. GA had been applied in verse areas of optimization, design and application (Goldberg and Holland 1988).

### Mothflame Algorithm

Mothflame optimization algorithm is inspired by the movement of moths fly in the night using a mechanism called transverse orientation for movement, where the insects uphold a permanent position with deference to the moon which can be used as an effective tool when travelling through a long distance in a straight track targeting a light that is a far. The important factor to consider in the algorithm is the moths and the position of the moths in the space. These moths can fly 1-D, 2-D or even hyper dimensional space with varying position vectors. The moths and the flame serves as a solution. The moths are the agents of search and goes around the search space and the flames are the notable signs or position path define by the moths during search space. As a results every moth search for a flame and upgrade it when searching for a better solution to avoid missing it best solution. The mechanism of the moths can be updated using

logarithmic spiral considering the following conditions:

1. Spiral's original point should begin from the moth
2. Spiral's ending point should be the location of the flame
3. Variation of the series of spiral should not exceed from the search space

In view of these points, logarithmic spiral for the MFO algorithm can be defined as follows:

$$S(M_i F_j) = D_i \cdot e^{bt} \cdot \cos(2\pi) + F_j \quad (13)$$

Where  $D_i$  indicates the distance of the  $i - th$  moth for the  $j - th$  flame,  $b$  is a constant for defining the shape of the logarithmic spiral, and  $t$  is a random number in  $(-1, 1)$ .  $D$  is computed as follows:

$$D_i = |F_j - M_i| \quad (14)$$

where  $M_i$  indicate the  $i - th$  moth,  $F_j$  indicates the  $j - th$  flame, and  $D_i$  indicates the distance of the  $i - th$  moth for the  $jth$  flame. (Mirjalili 2015)

### Particle Swarm Optimization

Particle Swarm Optimization Algorithm (PSO) was developed based on swarm behavior of fish and bird schooling in nature by their velocity  $V_i$  and position  $X_i$ . A flying swarm of birds searching for food inspires the general idea of PSO. The updates of  $V_i$  and  $X_i$  is determined by the formula below:

$$V_i^{t+1} = V_i^t + \alpha \varepsilon_1 [\varphi^* - X_i^t] + \beta \varepsilon_2 [X_i^* - X_i^t], \quad (15)$$

Where

$\varphi^*$  = recent best solution,  $X_i^*$  = individual best solution of particle  $i$ ,  $\varepsilon_1$  and  $\varepsilon_2$  = Random variable derived by the uniform distribution in  $[0, 1]$ ,  $\alpha$  &  $\beta$  = learning parameters

Thus, the position expressed as

$$X_i^{t+1} = X_i^t + V_i^{t+1} \quad (16)$$

PSO standard algorithm has been extended using many variant and the improvement is adopting inertia function. (Kennedy, Eberhart et al. 2005)

### Simulated Annealing Algorithm

Simulated Annealing (SA) is a good example of nature-inspired algorithms (Kirkpatrick, Gelatt et al. 1983) that is trajectory based. In some cases, SA is termed as the extension of metropolis-

hastings algorithm but used in another form. SA uses a stochastic approach in generating new moves and determining the acceptance. Boltzmann-type probability determined how a new move is been accepted.

$$P = \exp\left[-\frac{\Delta E}{K_B T}\right] \quad (17)$$

Where  $K_B$  represents the Boltzmann's constant,  $T$  represents the temperature for controlling the annealing process. Change in energy  $\Delta E$  should relate with the objective function  $f(x)$ . Hence, the preceding probability turn out to be:

$$P(\Delta f, T) = e^{-\Delta f/T} \quad (18)$$

Regardless of the changes, a random number ( $r$ ) is mostly use as threshold. Thus, move is accepted if  $p > r$ . it is important to note the special case

An algorithm is a set of interacting markov chain, we can generally express an algorithm as:

$$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}^{t+1} = A[x_1, \dots, x_n, \epsilon_1, \dots, \epsilon_m, p_1(t), \dots, p_k(t)] \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}^t \quad (19)$$

The equation generate a set of new solution  $(x_1 \dots x_2)^{t+1}$  derived from the population of  $n$  solutions. The value of a given matrix  $A$  determine the behaviour of an algorithm which is regulated by the parameters  $p_k(t)$  and randomness vector  $= (1, \dots, m)$ . According to markovian theory, the first number is making the convergence rate of the algorithm regulated by next value  $\lambda_2$  of  $A$ . However, it is very difficult to determine the general value. As a result of these the parameter tuning become very challenging work (Yang 2014). Therefore, parameter tuning is a relevant topic under active study. The purpose of parameter tuning is to get the best parameter settings, which will make an algorithm to perform well for a wide range of problems. Presently, parameter tuning done mostly using extensive, comprehensive parametric studies and no perfect method in general. Broadly, parameter tuning needs higher level optimization methods to deal with as it is an optimization problem itself (Yang 2014) .

when  $T \rightarrow 0$  corresponds to the classical hill-climbing because better results are accepted, it is important to note if the system is climbing up or decreasing along a hill (Yang, Chien et al. 2015)

## PARAMETER SETTING CONCEPTS

### Parameter Control and Tuning

A major challenge when dealing with metaheuristic algorithms is how to regulate exploitation and exploration effectively, which is subject to further research. It is necessary to have a relationship that can mathematically prove how parameters can extensively influence the performance of an algorithm, which still have not captured the attention of the researchers. Only unrealistic assumptions that lacks theoretical proofs or results.

### Sensitivity Analysis of Nature Inspired Algorithm

Sensitivity analysis (SA) is target towards determining which parameter of an algorithm have influence on the output of that algorithm. Sensitivity analysis (SA) is the study of how the uncertainty in the output of a model (numerical or otherwise) can be apportioned to different sources of uncertainty in the model input.it is termed as a prerequisite for model building in any setting be it prognostic or diagnostic.(Saltelli, Tarantola et al. 2004). The aim of carrying out sensitivity analysis is to study how various input parameters influence the general output of a model. SA is prevalent in modelling, simulation and in verse application area such as traffic simulation, macro-economic modelling etc. Saltelli, Tarantola, Campolongo, & Ratto list out several approaches to sensitivity analysis; as local SA, screening and global SA.

In the screening methods the focus on effects of inputs parameter on the model. Here explorarion of multiple dimensional space is allowed. The parameter influence is based on linear or non linear interaction and effects within the parameters.

Corresponding author: Raihanatu Mohammed Hamid

✉ [raihanatumohdhamid@gmail.com](mailto:raihanatumohdhamid@gmail.com)

Department of Mathematical Sciences, Faculty of Science, Abubakar Tafawa Balewa University, Bauchi.

© 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved



Considering several evaluations of functions  $f$ , SA permits to identify the inputs parameter  $x_i$ , in an input  $X = (x_1, \dots, x_D)$ , has a significant impact on  $f$ . For input  $X$ , the basic effect of the  $i$ th input parameter is evaluated using the equation where  $\Delta$  is an offset:

$$EE_i(X) = \frac{f(x_1, \dots, x_i + \Delta, \dots, x_D) - f(X)}{\Delta} \quad (20)$$

The methods develop is to come up with trajectories for sample inputs. A trajectory contains one input parameter at a time and a successful offset (Saltelli, Tarantola et al. 2004).

### METHODOLOGY

The paper ascertained the level of sensitivity of the ten NIMA using quality solution and run time as performance metrics. The NIMA were run using NiaPy framework involving PyScripter. Ink a python IDE to generate the results. Therefore, in order to compare the performance evaluation of nature-inspired some functions are applied to achieve optimum solution. To determine the best and worst case performance scenario of metaheuristic algorithms like convergence speed, robustness, fault

recovering metric, being a multimodal problem solver among others benchmark functions are used. No benchmarks function is better than another in terms of achieving robustness (Korkmaz and Akgüngör 2018)

Yang, in a survey paper about benchmark functions, suggested the use a good set of benchmark functions for performance validation and comparison with other algorithms. However, there is no any agreed good list of functions among metaheuristic researchers. However, Yang elaborates the good set should contain functions with diverse characteristics and should be unbiased. Nevertheless, researchers choose a set of functions on which the target algorithm performs better than other selected counterparts do. This raises an important concern over biasness towards algorithm's suitable set of functions (Hussain, Salleh et al. 2017). Twenty benchmarks functions were used as the test suites, ten unimodal and ten multimodal with their test functions and global minimum solution as shown in table 1

**Table 1:** Benchmarks functions (Name and global solution)

S/N	Name of test Function	Global solution
1.	Rastrigin	$f(x^*) = 0, \quad \text{at } x^* = (0, \dots, 0)$
2.	Rosenbrock	$f(x^*) = 0, \quad \text{at } x^* = (0, \dots, 0)$
3.	Griewank	$f(x^*) = 0, \quad \text{at } x^* = (0, \dots, 0)$
4.	Ackley	$f(x^*) = 0, \quad \text{at } x^* = (0, \dots, 0)$
5.	Schwefel	$f(x^*) = 0, \text{at } x^* = 420968, \dots, 420.96874$
6.	Schwefel221	$f(x^*) = 0, \quad \text{at } x^* = (0, \dots, 0)$
7.	Schwefel222	$f(x^*) = 0, \quad \text{at } x^* = (0, \dots, 0)$
8.	ModifiedSchwefel	$f(x^*) = 0, \text{at } x^* = 420.968746, \dots, 420.968746$
9.	Alpine1	$f(x^*) = 0, \quad \text{at } x^* = (0, \dots, 0)$
10.	Alpine2	$f(x^*) = 2.808^D, \quad \text{at } x^* = (7.917, \dots, 7.917)$
11.	HappyCat	$f(x^*) = 0, \quad \text{at } x^* = (-1, \dots, -1)$
12.	Ridge	$f(x^*) = 0, \quad \text{at } x^* = (0, \dots, 0)$
13.	Qing	$f(x^*) = 0, \quad \text{at } x^*(\pm i)$
14.	Quintic	$f(x^*) = 0, \quad \text{at } x^* = f(-1 \text{ or } 2)$

Corresponding author: Raihanatu Mohammed Hamid

✉ [raihanatumohdhamid@gmail.com](mailto:raihanatumohdhamid@gmail.com)

Department of Mathematical Sciences, Faculty of Science, Abubakar Tafawa Balewa University, Bauchi.

© 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved

15.	Salomon	$f(x^*) = 0, \quad \text{at } x^* = (0,0)$
16.	SumSquares	$f(x^*) = 0, \quad \text{at } x^* = (0, \dots, 0)$
17.	StyblinskiTang	$f(x^*) = -78.332 \text{ at } x^*$ $= (-2.903534, \dots, -2.903534)$
18.	BentCigar	$f(x^*) = 0,$ $\text{at } x^* = (420.968746, \dots, 420.968746)$
19.	Zakharov	$f(x^*) = 0, \quad \text{at } x^* = (0, \dots, 0)$
20.	Powell	$f(x^*) = 0, \text{ at } x^* = 420968, \dots, 420.96874$

Figure 1 shows the flow of activities undergone during the research experiment, where 10 nature-inspired metaheuristics algorithms were selected from the review. The algorithms were run using the NiaPy simulator using the simulators' benchmark functions. Sensitivity analysis is then carried out while different parameters are set and altered for several times to determine the best- and worst-case scenario of each algorithm. The results obtained were compared and determine the best case. A look up table is designed using the evaluated results.

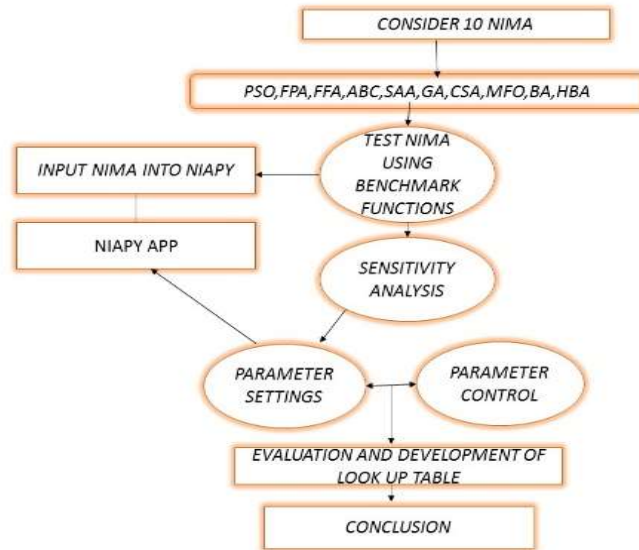


Figure 1: Data flow of the proposed work

## RESULTS AND DISCUSSION

From the beginning, each algorithm was run severally and 10 was selected for each individual algorithm. All algorithms were tested on 20 different benchmark functions. Different parameters were fined tuned for the experiment so as to ascertain the level of performance with respect to the parameter limits. For each results obtained from a single combination, the best, average and worse case values were selected from the results. The best case

is defined by the value that is closest to the global solution of a benchmark function that indicated the highest solution quality of that specific algorithm. The average case is the mean value of the results and the worst case is the results that is far from the global solution.

### Parameter Tuning

To demonstrate how NIMA becomes sensitive to parameter change, we run the

Corresponding author: Raihanatu Mohammed Hamid

✉ [raihanatumohdhamid@gmail.com](mailto:raihanatumohdhamid@gmail.com)

Department of Mathematical Sciences, Faculty of Science, Abubakar Tafawa Balewa University, Bauchi.

© 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved

experiment for the 10 algorithms with each algorithm being run 10 times with different parameter combinations.

**Table 2:** Different parameter tuning for ten metaheuristic algorithms for ten runs.

No of Runs	ABC	BAT	CSO	FFA	FPA	GA	HBAT	MFO	PSO	SA
	Parameters	Parameters	Parameters	Parameters	Parameters	Parameters	Parameters	Parameters	Parameters	Parameters
1	NP=10, Limit= 100	NP=10, A=0.01, r=0.01, Qmin=-1, Qmax=1	NP=10, pa=0.01, alpha=0.01	NP=10, alpha=1, betamin=1, gamma=1	NP=10, p=0.01, beta=0.01	NP=10, Ts=5, Mr=1, Cr=1	F=0.01, CR=0.01	NP=10	NP=10, C1=0.01, C2=0.01, w=1.0, vMin=-1.0, vMax=1.0	delta=0.01, T=1000, deltaT=0.01
2	NP=20, Limit= 200	NP=20, A=0.03, r=0.03, Qmin=-2, Qmax=2	NP=20, pa=0.03, alpha=0.03	NP=20, alpha=2, betamin=2, gamma=2	NP=20, p=0.03, beta=0.03	NP=20, Ts=10, Mr=2, Cr=2	F=0.03, CR=0.03	NP=20	NP=20, C1=0.03, C2=0.03, w=2.0, vMin=-2.0, vMax=2.0	delta=0.03, T=2000, deltaT=0.03
3	NP=30, Limit= 300	NP=30, A=0.05, r=0.05, Qmin=-3, Qmax=3	NP=30, pa=0.05, alpha=0.05	NP=30, alpha=3, betamin=3, gamma=3	NP=30, p=0.05, beta=0.05	NP=30, Ts=15, Mr=3, Cr=3	F=0.05, CR=0.05	NP=30	NP=30, C1=0.05, C2=0.05, w=3.0, vMin=-3.0, vMax=3.0	delta=0.05, T=3000, deltaT=0.05
4	NP=40, Limit= 400	NP=40, A=0.1, r=0.1, Qmin=-4, Qmax=4	NP=40, pa=0.1, alpha=0.1	NP=40, alpha=4, betamin=4, gamma=4	NP=40, p=0.1, beta=0.1	NP=40, Ts=20, Mr=4, Cr=4	F=0.1, CR=0.1	NP=40	NP=40, C1=0.1, C2=0.1, w=4.0, vMin=-4.0, vMax=4.0	delta=0.1, T=4000, deltaT=0.1
5	NP=50, Limit= 500	NP=50, A=0.15, r=0.15, Qmin=-5, Qmax=5	NP=50, pa=0.15, alpha=0.15	NP=50, alpha=5, betamin=5, gamma=5	NP=50, p=0.15, beta=0.15	NP=50, Ts=25, Mr=5, Cr=5	F=0.15, CR=0.15	NP=50	NP=50, C1=0.15, C2=0.15, w=5.0, vMin=-5.0, vMax=5.0	delta=0.15, T=5000, deltaT=0.15
6	NP=60, Limit= 600	NP=60, A=0.2, r=0.2, Qmin=-6, Qmax=6	NP=60, pa=0.2, alpha=0.2	NP=60, alpha=6, betamin=6, gamma=6	NP=60, p=0.2, beta=0.2	NP=60, Ts=30, Mr=6, Cr=6	F=0.2, CR=0.2	NP=60	NP=60, C1=0.2, C2=0.2, w=6.0, vMin=-6.0, vMax=6.0	delta=0.2, T=6000, deltaT=0.2
7	NP=70, Limit= 700	NP=70, A=0.4, r=0.4, Qmin=-7, Qmax=7	NP=70, pa=0.4, alpha=0.4	NP=70, alpha=7, betamin=7, gamma=7	NP=70, p=0.4, beta=0.4	NP=70, Ts=35, Mr=7, Cr=7	F=0.4, CR=0.4	NP=70	NP=70, C1=0.4, C2=0.4, w=7.0, vMin=-7.0, vMax=7.0	delta=0.4, T=7000, deltaT=0.4
8	NP=80, Limit= 800	NP=80, A=0.6, r=0.6, Qmin=-8, Qmax=8	NP=80, pa=0.6, alpha=0.6	NP=80, alpha=8, betamin=8, gamma=8	NP=80, p=0.6, beta=0.6	NP=80, Ts=40, Mr=8, Cr=8	F=0.6, CR=0.6	NP=80	NP=80, C1=0.6, C2=0.6, w=8.0, vMin=-8.0, vMax=8.0	delta=0.6, T=8000, deltaT=0.6
9	NP=90, Limit= 900	NP=90, A=0.8, r=0.8, Qmin=-9, Qmax=9	NP=90, pa=0.8, alpha=0.8	NP=90, alpha=9, betamin=9, gamma=9	NP=90, p=0.8, beta=0.8	NP=90, Ts=45, Mr=9, Cr=9	F=0.8, CR=0.8	NP=90	NP=90, C1=0.8, C2=0.8, w=9.0, vMin=-9.0, vMax=9.0	delta=0.8, T=9000, deltaT=0.8
10	NP=100, Limit= 1000	NP=100, A=1.0, r=1.0, Qmin=-10, Qmax=10	NP=100, pa=1.0, alpha=1.0	NP=100, alpha=10, betamin=10, gamma=10	NP=100, p=1.0, beta=1.0	NP=100, Ts=50, Mr=10, Cr=10	F=1.0, CR=1.0	NP=100	NP=100, C1=0.01, C2=0.01, w=1.0, vMin=-1.0, vMax=1.0	delta=1.0, T=10000, deltaT=1.0

Table 2 shows ten possible parameter combinations for the ten NIMA that were used to

obtain results that were compared with the global solution.

**Look Up Table for 10 Popular Metaheuristic Algorithms**

Comparing the NIMA algorithm results with the benchmark function global solutions, the best value of each algorithm is selected among the 10 possible combination of each algorithm to ascertain the algorithm that converges to the global solution. Among the values considered, the best case for each of the algorithm with respect to the benchmark function is selected to stand for the algorithm.

A look-up table has been developed purposely to the research community to choose the appropriate parameters for running any of the 10 popular NIMA. It contains the algorithms, parameter combination, solution quality and convergence speed. Section A, showing the look up table for NIMA with regards to solution quality.

**Section A: Look Up Table for NIMA with Regards to Solution Quality**

**Table 3:** Look up table for popular nature inspired metaheuristic algorithms (solution quality)

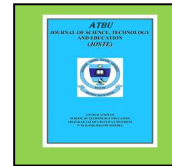
ABC	BENCHMARK	PARAMETER COMBINATION	BEST CASE VALUE
	Ackley	NP=90, Limit= 900	-0.01707
	Alpine 1	NP=100, Limit= 1000	0.00708
	Alpine 2	NP=10, Limit= 100	2.84182
	BentCigar	NP=90, Limit= 900	-0.00736
	Griewank	NP=10, Limit= 100	0.00351
	Happycat	NP=60, Limit= 600	0.01372
	Modified Schwefel	NP=60, Limit= 600	0.00664

Corresponding author: Raihanatu Mohammed Hamid

[raihanatumohdhamid@gmail.com](mailto:raihanatumohdhamid@gmail.com)

Department of Mathematical Sciences, Faculty of Science, Abubakar Tafawa Balewa University, Bauchi.

© 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved



Powell	NP=10, Limit= 100	0.00434
Qing	NP=20, Limit= 200	0.06833
Quintic	NP=30, Limit= 300	-0.01626
Ridge	NP=40, Limit= 400	-0.05213
Rosenbrock	NP=40, Limit= 400	-0.01755
Solomon	NP=90, Limit= 900	-0.00038
Schwefel	NP=10, Limit= 100	-0.35330
Schwefel221	NP=30, Limit= 300	-0.00247
Schwefel222	NP=100, Limit= 1000	0.01897
StyblinkisTang	NP=10, Limit= 100	-4.95247
Sumsquare	NP=40, Limit= 400	0.00779
Zakharov	NP=40, Limit= 400	0.01772
Rastrigin	NP=80, Limit= 800	-0.01123

**BAT**

BENCHMARK	PARAMETER COMBINATION	BEST CASE RESULTS
Ackley	NP=10, A=0.01, r=0.01, Qmin= -1, Qmax=1	-0.05079
Alpine 1	NP=20, A=0.03, r=0.03, Qmin= -2, Qmax=2	0.00448
Alpine 2	NP=80, A=0.6, r=0.6, Qmin= -8, Qmax=8	2.81233
BentCigar	NP=50, A=0.15, r=0.15, Qmin= -5, Qmax=5	0.08122
Griewank	NP=20, A=0.03, r=0.03, Qmin= -2, Qmax=2	0.00965
Happycat	NP=100, A=1.0, r=1.0, Qmin= -10, Qmax=10	0.08394
Modified Schwefel	NP=50, A=0.15, r=0.15, Qmin= -5, Qmax=5	0.22060
Powell	NP=100, A=1.0, r=1.0, Qmin= -10, Qmax=10	-0.00111
Qing	NP=50, A=0.15, r=0.15, Qmin= -5, Qmax=5	0.06228
Quintic	NP=10, A=0.01, r=0.01, Qmin= -1, Qmax=1	0.03029
Ridge	NP=100, A=1.0, r=1.0, Qmin= -10, Qmax=10	0.17272
Rosenbrock	NP=90, A=0.8, r=0.8, Qmin= -9, Qmax=9	-0.01349
Solomon	NP=10, A=0.01, r=0.01, Qmin= -1, Qmax=1	-0.04864
Schwefel	NP=100, A=1.0, r=1.0, Qmin= -10, Qmax=10	0.69604
Schwefel221	NP=60, A=0.2, r=0.2, Qmin= -6, Qmax=6	0.00064
Schwefel222	NP=40, A=0.1, r=0.1, Qmin= -4, Qmax=4	-0.00007
StyblinskiTang	NP=10, A=0.01, r=0.01, Qmin= -1, Qmax=1	-4.91466
Sumsquare	NP=70, A=0.6, r=0.6, Qmin= -7, Qmax=7	-0.00043
Zakharov	NP=10, A=0.01, r=0.01, Qmin= -1, Qmax=1	0.02134
Rastrigin	NP=30, A=0.01, r=0.01, Qmin= -1, Qmax=1	-0.00130

**CSO**

BENCHMARK	PARAMETER COMBINATION	BEST CASE VALUE
Ackley	NP=30,pa=0.05 and alpha=0.05	-0.00176
Alpine 1	NP=50,pa=0.15 and alpha=0.15	-0.00893
Alpine 2	NP=50,pa=0.15 and alpha=0.15	2.81333
BentCigar	NP=100,pa=1.0 and alpha=1.0	-0.05926
Griewank	NP=80,pa=0.6 and alpha=0.6	0.09206

Corresponding author: Raihanatu Mohammed Hamid

✉ [raihanatumohdhamid@gmail.com](mailto:raihanatumohdhamid@gmail.com)

Department of Mathematical Sciences, Faculty of Science, Abubakar Tafawa Balewa University, Bauchi.

© 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved



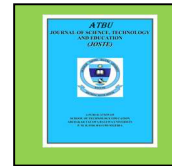
Happycat	NP=100,pa=1.0 and alpha=1.0	-0.06937
Modified Schwefel	NP=10,pa=0.01 and alpha=0.01	0.01202
Powell	NP=70,pa=0.4 and alpha=0.4	0.00304
Qing	NP=10,pa=0.01 and alpha=0.01	0.49773
Quintic	NP=100,pa=1.0 and alpha=1.0	0.00200
Ridge	NP=40,pa=0.1 and alpha=0.1	-0.08472
Rosenbrock	NP=100,pa=1.0 and alpha=1.0	0.06402
Solomon	NP=100,pa=1.0 and alpha=1.0	0.07164
Schwefel	NP=70,pa=0.4 and alpha=0.4	-0.09797
Schwefel221	NP=70,pa=0.4 and alpha=0.4	0.09640
Schwefel222	NP=50,pa=0.15 and alpha=0.15	-0.03795
StyblinskiTang	NP=70,pa=0.4 and alpha=0.4	-4.94499
Sumsquare	NP=90,pa=0.8 and alpha=0.8	0.01445
Zakharov	NP=50,pa=0.15 and alpha=0.15	-0.03570
Rastrigin	NP=80,pa=0.6 and alpha=0.6	-0.00363
<b>FFA</b>		
<b>BENCHMARK</b>	<b>PARAMETER COMBINATION</b>	<b>BEST CASE VALUE</b>
Ackley	NP=80,alpha=8, betamin=8, gamma=8	0.00706
Alpine 1	NP=70,alpha=7, betamin=7, gamma=7	-0.00820
Alpine 2	NP=40,alpha=4, betamin=4, gamma=4	2.80954
BentCigar	NP=90,alpha=9, betamin=9, gamma=9	0.06310
Griewank	NP=90,alpha=9, betamin=9, gamma=9	-0.18679
Happycat	NP=10,alpha=1, betamin=1, gamma=1	0.00196
Modified Schwefel	NP=40,alpha=4, betamin=4, gamma=4	0.01358
Powell	NP=40,alpha=4, betamin=4, gamma=4	-0.00434
Qing	NP=90,alpha=9, betamin=9, gamma=9	-0.51804
Quintic	NP=40,alpha=4, betamin=4, gamma=4	0.01344
Ridge	NP=30,alpha=3, betamin=3, gamma=3	-0.00152
Rosenbrock	NP=20,alpha=2, betamin=2, gamma=2	0.00223
Solomon	NP=90,alpha=9, betamin=9, gamma=9	0.20258
Schwefel	NP=40,alpha=4, betamin=4, gamma=4	0.00894
Schwefel221	NP=40,alpha=4, betamin=4, gamma=4	0.02777
Schwefel222	NP=30,alpha=3, betamin=3, gamma=3	0.00141
StyblinskiTang	NP=100,alpha=10, betamin=10, gamma=10	-4.99003
Sumsquare	NP=60,alpha=6, betamin=6, gamma=6	-0.01074
Zakharov	NP=60,alpha=6, betamin=6, gamma=6	-0.00438
Rastrigin	NP=20,alpha=2, betamin=2, gamma=2	-0.00388
<b>FPA</b>		
<b>BENCHMARK</b>	<b>PARAMETER COMBINATION</b>	<b>BEST CASE VALUE</b>
Ackley	NP=40, p=0.1, beta=0.1	-0.00002
Alpine 1	NP=50, p=0.15, beta=0.15	0.00000
Alpine 2	NP=90, p=0.8, beta=0.8	2.85009

Corresponding author: Raihanatu Mohammed Hamid

✉ [raihanatumohdhamid@gmail.com](mailto:raihanatumohdhamid@gmail.com)

Department of Mathematical Sciences, Faculty of Science, Abubakar Tafawa Balewa University, Bauchi.

© 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved



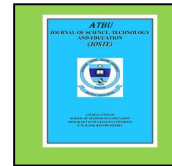
BentCigar	NP=40, p=0.1, beta=0.1	0.00000
Griewank	NP=40, p=0.1, beta=0.1	-0.00012
Happycat	NP=60, p=0.2, beta=0.2	-0.00012
Modified Schwefel	NP=60, p=0.2, beta=0.2	0.00000
Powell	NP=10, p=0.01, beta=0.01, NP=100, p=1.0, beta=1.0, NP=70, p=0.4, beta=0.4	0.00000
Qing	NP=90, p=0.8, beta=0.8	-0.00233
Quintic	NP=60, p=0.2, beta=0.2	-0.00001
Ridge	NP=100, p=1.0, beta=1.0	0.00003
Rosenbrock	NP=50, p=0.15, beta=0.15	0.00007
Solomon	NP=70, p=0.4, beta=0.4	0.00005
Schwefel	NP=100, p=1.0, beta=1.0	0.94365
Schwefel221	NP=20, p=0.03, beta=0.03	-0.00009
Schwefel222	NP=30, p=0.05, beta=0.05	0.00020
StyblinskiTang	NP=40, p=0.1, beta=0.1	-4.59590
Sumsquare	NP=20, p=0.03, beta=0.03	0.00000
Zakharov	NP=100, p=1.0, beta=1.0	0.00000
Rastrigin	NP=50, p=0.15, beta=0.15	0.00000
<b>GA</b>		
<b>BENCHMARK</b>	<b>PARAMETER COMBINATION</b>	<b>BEST CASE VALUE</b>
Ackley	NP=90, Ts=45, Mr=9, Cr=9	-0.00558
Alpine 1	NP=80, Ts=40, Mr=8, Cr=8	-0.03665
Alpine 2	NP=40, Ts=20, Mr=4, Cr=4	2.80480
BentCigar	NP=20, Ts=10, Mr=2, Cr=2	-0.08475
Griewank	NP=80, Ts=40, Mr=8, Cr=8	0.09206
Happycat	NP=100, Ts=50, Mr=10, Cr=10	-0.09824
Modified Schwefel	NP=100, Ts=50, Mr=10, Cr=10	-0.28541
Powell	NP=70, Ts=35, Mr=7, Cr=7	0.00304
Qing	NP=50, Ts=25, Mr=5, Cr=5	-0.05497
Quintic	NP=70, Ts=35, Mr=7, Cr=7	0.00723
Ridge	NP=40, Ts=20, Mr=4, Cr=4	-0.17230
Rosenbrock	NP=30, Ts=15, Mr=3, Cr=3	-0.02198
Solomon	NP=20, Ts=10, Mr=2, Cr=2	0.08675
Schwefel	NP=70, Ts=35, Mr=7, Cr=7	-0.09797
Schwefel221	NP=90, Ts=45, Mr=9, Cr=9	-0.02872
Schwefel222	NP=90, Ts=45, Mr=9, Cr=9	-0.01522
StyblinskiTang	NP=50, Ts=25, Mr=5, Cr=5	-4.97510
Sumsquare	NP=10, Ts=5, Mr=1, Cr=1	0.00457
Zakharov	NP=30, Ts=15, Mr=3, Cr=3	0.02011
Rastrigin	NP=20, Ts=10, Mr=2, Cr=2	-0.00466
<b>Hbat</b>		
<b>BENCHMARK</b>	<b>PARAMETER COMBINATION</b>	<b>BEST CASE VALUE</b>

Corresponding author: Raihanatu Mohammed Hamid

✉ [raihanatumohdhamid@gmail.com](mailto:raihanatumohdhamid@gmail.com)

Department of Mathematical Sciences, Faculty of Science, Abubakar Tafawa Balewa University, Bauchi.

© 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved



Ackley	F=0.15, CR=0.15	-0.03683
Alpine 1	F=0.01, CR=0.01	0.01225
Alpine 2	F=0.05, CR=0.05	2.80213
BentCigar	F=0.4, CR=0.4	-0.12894
Griewank	F=0.2, CR=0.2	-0.14268
Happycat	F=0.01, CR=0.01	-0.11587
Modified Schwefel	F=0.01, CR=0.01	-0.11587
Powell	F=0.05, CR=0.05	-0.01062
Qing	F=0.03, CR=0.03	-0.82042
Quintic	F=0.2, CR=0.2	-0.03303
Ridge	F=0.8, CR=0.8	-0.19569
Rosenbrock	F=0.2, CR=0.2	0.03980
Solomon	F=0.15, CR=0.15	-0.09955
Schwefel	F=0.4, CR=0.4	0.90507
Schwefel221	F=0.6, CR=0.6	0.08968
Schwefel222	F=1.0, CR=1.0	0.11197
StyblinskiTang	F=0.1, CR=0.1	-4.91718
Sumsquare	F=0.2, CR=0.2	-0.01027
Zakharov	F=0.05, CR=0.05	-0.03074
Rastrigin	F=0.8, CR=0.8	-0.00804
MFO		
<b>BENCHMARK</b>	<b>PARAMETER COMBINATION</b>	<b>BEST CASE VALUE</b>
Ackley	NP=70	-0.04401
Alpine 1	NP=70	-0.03781
Alpine 2	NP=90	2.81646
BentCigar	NP=30	-0.46532
Griewank	NP=50	-0.23694
Happycat	NP=30	-0.30086
Modified Schwefel	NP=10	0.11831
Powell	NP=70	-0.00570
Qing	NP=80	-0.55411
Quintic	NP=10	-0.02132
Ridge	NP=40	-0.14196
Rosenbrock	NP=60	0.11020
Solomon	NP=100	0.34712
Schwefel	NP=30	-0.86808
Schwefel221	NP=60	-0.18139
Schwefel222	NP=70	-0.10041
StyblinskiTang	NP=50	-4.97776
Sumsquare	NP=30	-0.01176
Zakharov	NP=10	0.02365
Rastrigin	NP=80	-0.00486

Corresponding author: Raihanatu Mohammed Hamid

✉ [raihanatumohdhamid@gmail.com](mailto:raihanatumohdhamid@gmail.com)

Department of Mathematical Sciences, Faculty of Science, Abubakar Tafawa Balewa University, Bauchi.

© 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved

**PSO**

BENCHMARK	PARAMETER COMBINATION	BEST CASE VALUE
Ackley	NP=50, C1=0.15, C2=0.15, w=5.0, vMin= -5.0, vMax=5.0	-0.00145
Alpine 1	NP=90, C1=0.8, C2=0.8, w=8.0, vMin= -9.0, vMax=9.0	-0.01159
Alpine 2	NP=30, C1=0.05, C2=0.05, w=3.0, vMin= -3.0, vMax=3.0	2.80156
BentCigar	NP=60, C1=0.2, C2=0.2, w=6.0, vMin= -6.0, vMax=6.0	0.10895
Griewank	NP=90, C1=0.8, C2=0.8, w=8.0, vMin= -9.0, vMax=9.0	-0.13660
Happycat	NP=20, C1=0.03, C2=0.03, w=2.0, vMin= -2.0, vMax=2.0	0.09376
Modified Schwefel	NP=10, C1=0.01, C2=0.01, w=1.0, vMin= -1.0, vMax=1.0	-0.16507
Powell	NP=70, C1=0.4, C2=0.4, w=7.0, vMin= -7.0, vMax=7.0	0.00218
Qing	NP=60, C1=0.2, C2=0.2, w=6.0, vMin= -6.0, vMax=6.0	-2.17469
Quintic	NP=10, C1=0.01, C2=0.01, w=1.0, vMin= -1.0, vMax=1.0	0.00539
Ridge	NP=10, C1=0.01, C2=0.01, w=1.0, vMin= -1.0, vMax=1.0	0.11273
Rosenbrock	NP=20, C1=0.03, C2=0.03, w=2.0, vMin= -2.0, vMax=2.0	0.01032
Solomon	NP=10, C1=0.01, C2=0.01, w=1.0, vMin= -1.0, vMax=1.0	-0.25057
Schwefel	NP=20, C1=0.03, C2=0.03, w=2.0, vMin= -2.0, vMax=2.0	0.11495
Schwefel221	NP=50, C1=0.15, C2=0.15, w=5.0, vMin= -5.0, vMax=5.0	0.05887
Schwefel222	NP=30, C1=0.05, C2=0.05, w=3.0, vMin= -3.0, vMax=3.0	0.02462
StyblinskiTang	NP=50, C1=0.15, C2=0.15, w=5.0, vMin= -5.0, vMax=5.0	-5.00000
Sumsquare	NP=60, C1=0.2, C2=0.2, w=6.0, vMin= -6.0, vMax=6.0	0.01531
Zakharov	NP=30, C1=0.05, C2=0.05, w=3.0, vMin= -3.0, vMax=3.0	0.00376
Rastrigin	NP=30, C1=0.05, C2=0.05, w=3.0, vMin= -3.0, vMax=3.0	0.00748

**SA**

BENCHMARK	PARAMETER COMBINATION	BEST CASE VALUE
Ackley	delta=0.03, T=2000, deltaT=0.03	0.19315
Alpine 1	delta=0.01, T=1000, deltaT=0.01	0.02317
Alpine 2	delta=1.0, T=10000, deltaT=1.0	2.80719
BentCigar	delta=0.6, T=8000, deltaT=0.6	-0.27107
Griewank	delta=0.01, T=1000, deltaT=0.01	-0.25240
Happycat	delta=0.05, T=3000, deltaT=0.05	0.13873
Modified Schwefel	delta=0.01, T=1000, deltaT=0.01	-0.10779
Powell	delta=1.0, T=10000, deltaT=1.0	0.00595
Qing	delta=0.05, T=3000, deltaT=0.05	0.87247
Quintic	delta=0.05, T=3000, deltaT=0.05	-0.02745
Ridge	delta=1.0, T=10000, deltaT=1.0	-0.13024
Rosenbrock	delta=0.6, T=8000, deltaT=0.6	-0.03632
Solomon	delta=0.15, T=5000, deltaT=0.15	0.17794
Schwefel	delta=0.01, T=1000, deltaT=0.01	1.13244
Schwefel221	delta=0.03, T=2000, deltaT=0.03	-0.10274
Schwefel222	delta=0.03, T=2000, deltaT=0.03	0.14898
StyblinskiTang	delta=0.4, T=7000, deltaT=0.6	-5.00000

Corresponding author: Raihanatu Mohammed Hamid

✉ [raihanatumohdhamid@gmail.com](mailto:raihanatumohdhamid@gmail.com)

Department of Mathematical Sciences, Faculty of Science, Abubakar Tafawa Balewa University, Bauchi.

© 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved



Sumsquare	delta=0.01, T=1000, deltaT=0.01	-0.07320
Zakharov	delta=0.4, T=7000, deltaT=0.6	-0.01376
Rastrigin	delta=0.15, T=5000, deltaT=0.15	-0.01049

### Time Complexity

The time taken to run the experiment was presented in Table 4.12 where the time taken for each combination to start and complete successful was recorded. The minimum amount of time was then selected for every algorithm and the best times(s) of individual algorithm. With regard to time complexity, the best algorithm with minimum run time value of 3.52(secs) is CSO, when the parameter combinations are NP=70, =0.4 and alpha=0.4. The second best algorithm with also minimal time is SA with parameter combination delta=0.2, T=6000, deltaT=0.2. The third best algorithm with minimal time is PSO with run time of 3.6(secs) when the parameters are tuned to NP=50, pa=0.15 and alpha=0.15. The fourth algorithm is Bat algorithm with complete run time of 3.65secs with parameter

combination NP=40, A=0.1, r=0.1, Qmin= -4, Qmax=4. FPA is the fifth algorithm with run time of 3.69secs with combination NP=70 p=0.4, beta=0.4. The next algorithm is ABC with run time of 3.71secs when the algorithm parameters are set to be NP=40, Limit= 400. The seventh algorithm in terms of time is Hbat with best run time of 4.05sec when the parameters are F=1.0, CR=1.0. GA is the next algorithm with run time of 4.67secs and the parameter set that achieved the best result is NP=10, Ts=5, Mr=1, Cr=1. The next algorithm is MFO which run successfully at 4.69secs when the parameter is NP=20. However, the slowest algorithm out of the ten NIMA experimented in relation to time is FFA which completed a runtime of 7.3secs when the parameters were set to NP=70, alpha=7.0, betamin=7.0, gamma=7.0.

Section B: Look Up Table for NIMA with Regards Time Complexity.

regards to run time CSO outperforms followed by SA, PSO, Bat, FPA, ABC, Hbat, GA, MFO and lastly FFA.

Time taken for each parameters combination for the 10 algorithms was recorded upon completion of its run and the fastest run time is being selected for each algorithm for comparison analysis as presented in Table 3. With

The results are rounded to 5 decimal places. The algorithms are arranged in alphabetic order to aid easy reference. The look up table has added suggested way of getting optimal parameter values for running the popular NIMA.

Table 3: Look up table for popular nature inspired metaheuristic algorithms (Speed)

ALGORITHM	PARAMETER COMBINATION	RUN TIME(secs)
ABC	NP=40, Limit= 400	3.71
BAT	NP=40, A=0.1, r=0.1, Qmin= -4, Qmax=4	3.65
CSO	NP=70,pa=0.4 and alpha=0.4	3.52
FFA	NP=70,alpha=7.0, betamin=7.0, gamma=7.0	7.3
FPA	NP=70 p=0.4, beta=0.4	3.69
GA	NP=10,Ts=5, Mr=1, Cr=1	4.67
Hbat	F=0.01, CR=0.01	4.05
MFO	NP=20	4.69
PSO	NP=40, C1=0.1, C2=0.1, w=4.0, vMin= -4.0, vMax=4.0	3.61
SA	delta=0.2, T=6000, deltaT=0.2	3.58

Corresponding author: Raihanatu Mohammed Hamid

✉ [raihanatumohdhamid@gmail.com](mailto:raihanatumohdhamid@gmail.com)

Department of Mathematical Sciences, Faculty of Science, Abubakar Tafawa Balewa University, Bauchi.

© 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved

## CONCLUSION

In this paper, we provided a well-developed lookup table for nature inspired meta-heuristic algorithms parameter control settings for easing the work of the research community in chosen the appropriate parameter values. The nature inspired meta-heuristic algorithms look-up table developed has added to the literature another suggested way of getting the optimal combination of parameter setting values. It can eliminate the tedious work of comparative analysis required when selecting parameter setting combination for algorithms in solving optimization problems. The metaheuristics look-up table has the potential for improving time management, resources management and efficiency for the research community.

## REFERENCE

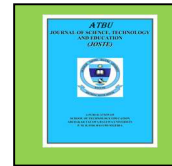
- Agarwal, P. and S. Mehta (2018). "Empirical analysis of five nature-inspired algorithms on real parameter optimization problems." *Artificial Intelligence Review* 50(3): 383-439.
- Akbari, M. and M. Henteh (2019). "Comparison of Genetic Algorithm (GA) and Particle Swarm Optimization Algorithm (PSO) for Discrete and Continuous Size Optimization of 2D Truss Structures." *Journal of Soft Computing in Civil Engineering* 3(2): 78-98.
- Bertolini, M., D. Mezzogori and F. Zammori (2019). "Comparison of new metaheuristics, for the solution of an integrated jobs-maintenance scheduling problem." *Expert Systems with Applications* 122: 118-136.
- Chassiakos, A. and G. Rempis (2019). "Evolutionary algorithm performance evaluation in project time-cost optimization." *Journal of Soft Computing in Civil Engineering* 3(2): 16-29.
- Chiroma, H., T. Herawan, I. Fister Jr, I. Fister, S. Abdulkareem, L. Shuib, M. F. Hamza, Y. Saadi and A. Abubakar (2017). "Bio-inspired computation: recent development on the modifications of the cuckoo search algorithm." *Applied Soft Computing* 61: 149-173.
- Ebrahimi, A., H. W. Jeon, S. Lee and C. Wang (2020). "Minimizing total energy cost and tardiness penalty for a scheduling-layout problem in a flexible job shop system: A comparison of four metaheuristic algorithms." *Computers & Industrial Engineering* 141: 106295.
- Ezugwu, A. E., A. K. Shukla, R. Nath, A. A. Akinyelu, J. O. Agushaka, H. Chiroma and P. K. Muhuri (2021). "Metaheuristics: a comprehensive overview and classification along with bibliometric analysis." *Artificial Intelligence Review*: 1-80.
- Goldberg, D. E. and J. H. Holland (1988). "Genetic algorithms and machine learning."
- Hasnan, K. b., A. M. Elewe, A. b. Nawawi and S. Tahir (2017). Comparative evaluation of firefly algorithm and MC-GPSO for optimal RFID Network Planning. 2017 8th International Conference on Information Technology (ICIT).
- Janah, M. and Y. Fujimoto (2018). Study comparison between firefly algorithm and particle swarm optimization for SLAM problems. 2018 International Power Electronics Conference (IPEC-Niigata 2018 -ECCE Asia).
- Karaboga, D. and B. Basturk (2007). "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm." *Journal of Global Optimization* 39(3): 459-471.
- Kaur, A. and A. P. Agrawal (2017). A comparative study of Bat and Cuckoo search algorithm for regression test case selection. 2017 7th International Conference on Cloud Computing, Data Science & Engineering - Confluence.
- Kaveh, A., R. A. Izadifard and L. Mottaghi (2020). "Optimal design of planar RC frames considering CO2 emissions using ECBO, EVPS and PSO metaheuristic algorithms." *Journal of Building Engineering* 28: 101014.
- Kennedy, J., R. C. Eberhart and Y. Shi (2005). chapter seven - The Particle Swarm. *Swarm Intelligence*. J. Kennedy, R. C. Eberhart and Y. Shi. San Francisco, Morgan Kaufmann: 287-325.
- Korkmaz, E. and A. P. Akgüngör (2018). "Comparison of artificial bee colony and flower pollination algorithms in vehicle

Corresponding author: Raihanatu Mohammed Hamid

✉ [raihanatumohdhamid@gmail.com](mailto:raihanatumohdhamid@gmail.com)

Department of Mathematical Sciences, Faculty of Science, Abubakar Tafawa Balewa University, Bauchi.

© 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved



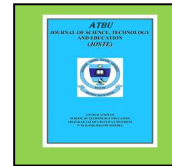
- delay models at signalized intersections." *Neural Computing and Applications*.
- Leal-Naranjo, J.-A., J.-A. Soria-Alcaraz, C.-R. Torres-San Miguel, J.-C. Paredes-Rojas, A. Espinal and H. Rostro-González (2019). "Comparison of metaheuristic optimization algorithms for dimensional synthesis of a spherical parallel manipulator." *Mechanism and Machine Theory* 140: 586-600.
- Luthra, I., S. K. Chaturvedi, D. Upadhyay and R. Gupta (2017). Comparative study on nature inspired algorithms for optimization problem. 2017 International conference of Electronics, Communication and Aerospace Technology (ICECA), IEEE.
- Ma, J., Z. Bi, T. O. Ting, S. Hao and W. Hao (2016). "Comparative performance on photovoltaic model parameter identification via bio-inspired algorithms." *Solar Energy* 132: 606-616.
- Mirjalili, S. (2015). "Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm." *Knowledge-Based Systems* 89: 228-249.
- Mohseni, S., A. C. Brent and D. Burmester (2020). "A comparison of metaheuristics for the optimal capacity planning of an isolated, battery-less, hydrogen-based micro-grid." *Applied Energy* 259: 114224.
- Moya, I., M. Chica and J. Bautista (2019). "Constructive metaheuristics for solving the Car Sequencing Problem under uncertain partial demand." *Computers & Industrial Engineering* 137: 106048.
- Oral, M. and S. S. Turgut (2018). Performance Analysis of Nature Inspired Computing Algorithms Under Hard Restrictions. 2018 Innovations in Intelligent Systems and Applications Conference (ASYU).
- Oral, M. and S. S. Turgut (2018). Performance Analysis of Relatively New Nature Inspired Computing Algorithms. 2018 Innovations in Intelligent Systems and Applications Conference (ASYU).
- Parpinelli, R., G. Plichoski, R. Silva and P. Narloch (2019). "A review of techniques for online control of parameters in swarm intelligence and evolutionary computation algorithms." *International Journal of Bio-Inspired Computation* 13: 1.
- Pavlyukevich, I. (2007). "Lévy flights, non-local search and simulated annealing." *Journal of Computational Physics* 226(2): 1830-1844.
- Rathasamuth, W. and S. Nootyaskool (2016). Comparison solving discrete space on flower pollination algorithm, PSO and GA. 2016 8th International Conference on Knowledge and Smart Technology (KST).
- Sachan, R. K., S. Gupta and D. S. Kushwaha (2018). Evaluation and Research Directions in Nature-Inspired Algorithms. 2018 5th IEEE Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON).
- Saeidian, B., M. S. Mesgari and M. Ghodousi (2016). "Evaluation and comparison of Genetic Algorithm and Bees Algorithm for location-allocation of earthquake relief centers." *International Journal of Disaster Risk Reduction* 15: 94-107.
- Salgotra, R., U. Singh and S. Saha (2018). "New cuckoo search algorithms with enhanced exploration and exploitation properties." *Expert Systems with Applications* 95: 384-420.
- Sapra, D., R. Sharma and A. P. Agarwal (2017). Comparative study of metaheuristic algorithms using Knapsack Problem. 2017 7th International Conference on Cloud Computing, Data Science & Engineering - Confluence.
- Singh, G. and B. S. Walia (2017). "Performance evaluation of nature-inspired algorithms for the design of bored pile foundation by artificial neural networks." *Neural Computing and Applications* 28(1): 289-298.
- Vrbančič, G., L. Brezočnik, U. Mlakar, D. Fister and I. Fister (2018). "NiaPy: Python microframework for building nature-inspired algorithms." *Journal of Open Source Software* 3(23): 613.
- Yang, X.-S., S. F. Chien and T. O. Ting (2015). Bio-inspired computation and optimization: an overview. *Bio-Inspired Computation in Telecommunications*, Elsevier: 1-21.

Corresponding author: Raihanatu Mohammed Hamid

✉ [raihanatumohdhamid@gmail.com](mailto:raihanatumohdhamid@gmail.com)

Department of Mathematical Sciences, Faculty of Science, Abubakar Tafawa Balewa University, Bauchi.

© 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved



- Yang, X.-S. and X. He (2016). Nature-Inspired Optimization Algorithms in Engineering: Overview and Applications. Nature-Inspired Computation in Engineering. X.-S. Yang. Cham, Springer International Publishing: 1-20.
- Yang, X.-S. and J. Papa (2016). Bio-inspired computation and its applications in image processing: an overview. Bio-inspired computation and applications in image processing, Elsevier: 1-24.

---

Corresponding author: Raihanatu Mohammed Hamid

✉ [raihanatumohdhamid@gmail.com](mailto:raihanatumohdhamid@gmail.com)

Department of Mathematical Sciences, Faculty of Science, Abubakar Tafawa Balewa University, Bauchi.

© 2023. Faculty of Tech. Education, ATBU Bauchi. All rights reserved