Enhanced Adaptive Simulated Based Artificial Gorilla Troop Optimizer for Global Optimisation

A. Bright*, A. K. Emmanuel, T. Elvis, F. A. Emmanuel
Department of Electrical and Electronic Engineering, Kwame Nkrumah University of Science and Technology, Kumasi, Ghana
*corresponding author's email: bennychrisbright@gmail.com

Abstract – *The enhancement of metaheuristic algorithms has been considered an important step in improving the solution quality of systems. In this paper, a modification to the traditional Gorilla Troop optimizer is proposed. The modification leverages the powerful properties of circle chaotic mapping and step adaptive simulation. The conventional algorithm, while effective in certain scenarios, exhibits limitations in handling complex and dynamically changing data sets. To address these shortcomings, a three-fold approach is proposed to enhance its performance. A circle chaotic mapping is integrated into the algorithm's initialization phase to enhance its sensitivity to initial conditions. The chaotic mapping effectively diversifies the search space, facilitating improved exploration and convergence to optimal solutions. Secondly, a step adaptive simulation is introduced as a means to dynamically adjust the simulation steps during runtime. Finally, the concept of adaptive simulation based on the state of the silverback gorilla (best solution) in the troop is used to simulate the exploitation phase to help the ASGTO overcome local optima entrapment and produce better solutions. The performance of the proposed ASGTO was assessed on twenty-two benchmark optimization functions and compared with the standard GTO, grey wolf optimizer (GWO), and whale optimization algorithm (WOA). The results showed that the proposed ASGTO outperformed the standard GTO, GWO, and WOA. ASGTO, GTO, GWO, and WOA attained global optimum values for 82%, 77%, 55%, and 59% of the 22 benchmark functions respectively. Consequently, the modified algorithm exhibits robustness and adaptability, making it applicable across various domains. The ASGTO is therefore recommended for adoption in solving optimization problems.*

Keywords: *artificial gorilla troop optimization, adaptive simulation, nature-inspired optimization, optimization*

Article History
Received 17 July 2023
Received in revised form 13 October 2023
Accepted 25 October 2023

I. Introduction

Metaheuristic algorithms have become widely acclaimed and exceptionally beneficial for addressing a myriad of optimization challenges, they have accumulated immense favour across industries. Their versatility and efficacy make them a top choice for professionals in various domains in recent times, due to the inability of traditional mathematical methods such as estimation methods, linear programming, etc. to solve complex practical problems [1]. They have found extensive applications across diverse fields, including but not limited to: medicine, engineering, etc. to solve patient classification problems, optimization of systems,[2] etc. Unlike heuristic algorithms, metaheuristic algorithms are generic in nature, making them adaptable for a multitude of distinct optimization problems. However, the stochastic search mechanism of these algorithms, while powerful and versatile, do not provide a guarantee of locating the absolute best solution with one kind of algorithm, in one try, for every optimization problem [3]. Therefore, researchers are tirelessly putting in much effort to develop more accurate, efficient, and robust algorithms to obtain excellent global optimal solutions to complex real-world problems.

Generally, metaheuristic algorithms are nature-inspired or based on natural phenomena and physical processes. They have been categorized into swarm intelligent algorithms, bio-inspired algorithms, physics and chemistry-based algorithms, and evolutionary algorithms [4]. Although numerous metaheuristic algorithms, exemplified by the likes of particle swarm optimization (PSO) [5], firefly algorithm [6], and differentiation evolution (DE) algorithm [7] Spotted Hyena Optimizer Algorithm (SHO) [8], Cuckoo Search Algorithm [9], Reptile Search Algorithm (RSA) [10]. Sine Cosine Algorithm [11], Harmony Search [12] have proven their effectiveness in tackling complex optimization problems, there is still a need for new and accurate algorithms due to the common issue of slow convergence, inaccurate results, local optima entrapment, etc. in the existing algorithms. Again, researchers are also working tirelessly to improve the general performances of the existing algorithms through, for example, hybridization of algorithms, and application of theories such as opposition-based learning, fuzzy systems, and rough set theory. These efforts are being made due to the need to address intricate large-scale ever-changing optimization problems efficiently and with speed.

Gorilla Troop Optimizer (GTO), a social group intelligence algorithm, drawing inspiration from the collective behavior of gorillas in a troop has proven to be an excellent competitor amongst the existing metaheuristic algorithms and has particularly performed exceptionally well when compared to popular algorithms of swarm intelligence, like the Grey Wolf Optimizer (GWO), Particle Swarm Optimization (PSO), Whale Optimization Algorithm (WOA), and Moth-Flame Optimization, (MFO), on various benchmark optimization functions [1]. The GTO has also been applied in the resolution of multifaceted energy-related problems [8]. In [13], GTO was applied to extract Photovoltaic (PV) parameters of single-diode and double-diode models which are critical for accurately representing the behaviour of photovoltaic (PV) cells and modules. Also, in [14] GTO was used for tunning TID based power system stabilizer to optimise its stability, ensuring the reliability and consistent functioning of the power system. To enhance the performance of the GTO further and also deal with the inherent problem of convergence inaccuracy and instability during solving complex problems, an improvement has been suggested in [15], In the work, circle chaotic mapping is used to facilitate diversity amongst the population and improve global search in GTO and opposition-based learning was employed to expand the search ranges of the gorillas to help avoid local optima entrapment. In [2] a fusion strategy for controlling shrinkage is introduced to broaden the exploration of the search space and mitigate search limitations. This approach enhances interaction between silverback gorillas and other members of the group, leading to improved global optimization performance. Regardless of these introductions into the GTO, it still suffers from entrapment in local optima for certain complex problems and inaccurate results. Hence, further novel ideas are needed to further improve the algorithm GTO.

This work presents a novel adaptive simulated gorilla troop optimizer (ASGTO) based on the concept of adaptive simulation. This proposition aims to address the prevalent issue of local optima entrapment and inaccurate results. To achieve this, an adaptive simulation is introduced into the exploitation phase of the GTO which simulates gorilla's behaviours based on the current state of the silverback (best solution) in the troop. This helps the GTO to reach convergence faster and produce accurate results. Also, a circle chaotic mapping is integrated into the algorithm's initialization phase, which imbues the system with enhanced sensitivity to initial conditions. The chaotic mapping effectively diversifies the search space, facilitating improved exploration and convergence to optimal solutions. Consequently, the modified algorithm exhibits heightened robustness and adaptability, rendering it suitable for applications across various domains. we also introduce step adaptive simulation as a means to dynamically adjust the simulation steps during runtime. By dynamically tuning the step size, the algorithm adapts its precision according to the complexity of the problem, effectively conserving computational resources without compromising accuracy. This innovation not only accelerates convergence in less intricate regions but also ensures precise convergence in areas where fine-grained exploration is critical.

II. Artificial Gorilla Troop Optimizer

The concept of Artificial Gorilla Troop Optimization (GTO) simulates the social group behaviors observed in gorillas. The algorithm simulates five unique characteristics of the gorillas in a troop. These are; the migration of gorillas to uncharted territories, their movement towards fellow gorillas, their migration to familiar locations, and their tendency to follow the lead of the silverback gorilla (best gorilla in the troop), and competition by matured males for adult females. These collectively constitute the exploration and exploitation aspects of the optimization process. The metaheuristic process of GTO for attaining optimum solution is guided by a set of rules. These rules are elaborated below.

The solution space is made up of three different solutions. These are, X as current position vector of gorilla, GX as candidate position vectors are created during each phase of the GTO. GX operates when its value surpasses that of the current solution and $X_{silverback}$ represent the global solution after each iteration.

- In the search process, only one gorilla becomes a silverback.
- During every iteration of the search procedure, a solution GX is created. If GX is better than the current solution X, GX it replaces X it. Otherwise, the solution remains in the memory GX.
- In the GTO, a search process is assumed to start with the worst solution (weakest member of the group) and gradually move towards the best solution (silverback) thereby improving the positions of all other gorillas.

A. Exploration Phase

Within the exploration phase, three methods are employed to simulate the behaviour of the gorillas. These are; movement to an undiscovered area, a movement towards a discovered area in the search space, and movement to other gorillas. The first movement ensures effective monitoring of the entire problem search space; the second movement is designed to explore the search space, whereas the third prevents the GTO from getting trapped in local optima. In the search process, all gorillas are considered as a candidate solution in each optimization stage, the best gorilla is designated as the silverback. These mechanisms are simulated randomly.

- Gorillas move to an undiscovered area when a variable $p > r$ and position update of gorilla is done according to equation (1).
- The mechanism of movement towards other gorillas is activated when, update of position is done using equation (2),
- Movement towards a discovered area is activated when, and update of gorilla position is done according to equation (3).

$$GX(t+1) = \begin{cases} (ub-lb)\times r_1 + lb, & r < p \\ (r_2 - C)\times X_1(t) + L\times Z\times X(t), & r \geq 0.5 \\ X(t) - L\times(L\times(X(t)-X_1(t)+r_3\times(X(t)-X_2(t)), & r < 0.5 \end{cases} \quad (1)\text{-}(3)$$

where, $Z = [-C, C]$.

In the equations above with the first line representing (1)-(3), $GX(t+1)$ is the candidate position vector of gorilla in the subsequent t iterations, $X(t)$ is the current position vector of gorilla, r, r_1, r_2, r_3 are random numbers within $[0,1]$ .The variable p is set at the initialization stage and has a value within $[0,1]$. It is employed to select which amongst the three mechanisms highlighted above must operate. ub and lb are respectively the upper and lower boundaries of the variables of the problems being solved. X_1 and X_2 are random gorilla and vector of gorilla candidate position respectively. The variables C and L are calculated using equation (4) and (5) respectively.

B. Exploitation phase

In the exploitation phase, the gorillas search within their own search space based on two behaviors; gorillas follow silverback gorilla who is the best solution in the troop or male gorillas compete amongst themselves for adult females. In the search process of GTO, selection of either of the two behaviors is randomly simulated with variables C and W, where C is defined according to equation (4) and W is a number between 0 and 1 chosen at the initialization of the GTO.

$$C = F \times \left(1 - \frac{t}{MaxIter}\right) \quad (4)$$

$$L = C \times l \quad (5)$$

$$F = \cos(2\times r_4) + 1$$

The variable r_4 represent a random number within $[0,1]$, t is a random number of the range $[-1,1]$, t signifies the current iteration and $MaxIter$ represent the maximum number of iterations to be executed. The rules for choosing either of the two behaviours using C and W are as follows;

Follow Silverback

Gorillas begin to follow the silverback when $C > W$. This behaviour is exhibited when the silverback is young, healthy and strong to lead the troop, make decisions, determine the groups movement and direct the troop towards a potential food source. In this state of the silverback, members of the troop follow the silverback well and obey instructions from the silverback. This is simulated according to equation (6).

$$GX(t+1) = L \times M \times \left(X(t) - X_{silverback} \right) + X(t) \qquad (6)$$

where $M = \left(\left| \frac{1}{N} \sum_{i=1}^{N} GX_i(t) \right|^g \right)^{\frac{1}{g}}$, $g = 2^L$, $L = C \times l$

Male gorillas compete for female

Male gorillas compete or fight amongst themselves for females when $C < W$. This occurs when young gorillas grow up and become stronger. The fight is violent and may last for many days. This behaviour is simulated with equation (7).

$$GX(i) = X_{silverback} - (X_{silverback} \times Q - X(t) \times Q) \times A \qquad (7)$$

$$Q = 2 \times r_5 - 1$$

$$A = \beta \times E$$

$$E = \begin{cases} N_1, rand \geq 0.5 \\ N_2, rand < 0.5 \end{cases}$$

In equation (7), Q represent the impact force with which the male gorillas compete, r_5 is a random number between $[0,1]$, A is a coefficient vector meant to determine the degree of violence in the fight amongst the gorillas. Again, β is parameter selected in the initialization of the algorithm and E is a parameter used to simulate the impact of violence on the dimensions of solutions. The exploitation phase of the GTO algorithm is completed with group formation where, the cost associated with all GX solutions is determined and compared to cost of X. At an iteration t, where $GX(t) < X(t)$, $GX(t)$ replaces $X(t)$ solution. Again, the best solution amongst the gorillas of the GTO population at iteration t becomes the silverback. The implementation of the GTO algorithm is shown in the flowchart in Fig.1.

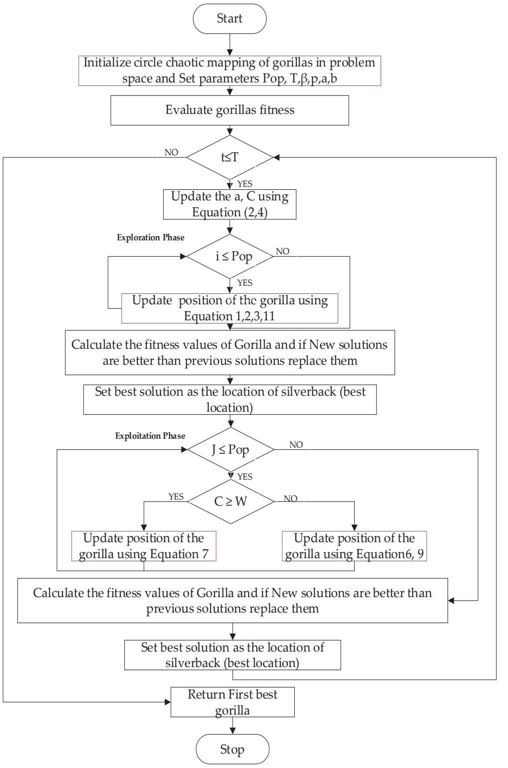

Fig. 1. Flowchart of algorithm implementation

III. Proposed Modification

The three proposed modifications are presented below.

A. Circle Chaotic Mapping Initialisation

Inspired by the work done in [15], circle chaotic mapping is utilized to improve the quality of the initial population. Chaotic mapping is a technique used in nonlinear systems that possess properties such as unpredictability, ergodicity, and randomness. Chaotic mapping is preferred over random distribution since it allows individuals within the initial population to facilitates a comprehensive exploration of the solution space, resulting in increased convergence speed and sensitivity. the initial population. Compared to commonly used Tent chaotic mapping and Logistic chaotic mapping, chaotic mapping is a widely accepted strategy for enhancing optimization performance. To further enhance population diversity and leverage the solution space information, the circle chaotic mapping technique is incorporated to improve the initialization process of the original GTO. The mathematical representation of the circle chaotic mapping technique is provided below in equation 8.

$$z_{k+1} = z_k + b - \frac{a}{2\pi}.\sin(2\pi z_k)\,\mathrm{mod}(1), z_k \in (0,1) \quad (8)$$

$$a = 0.5,\ b = 0.2$$

B. Adaptive simulation

In gorilla troop optimization (GTO), a global solution is achieved by both exploitation and exploration of the search space. This is achieved by randomly simulating the mechanisms that define the behaviours of the gorillas. The numerous mechanisms in the algorithm help it to achieve a better performance compared to other algorithms. Regardless of the performance of the GTO it still suffers from slower convergence and local optima entrapment in solving optimization problems. These deficiencies must be addressed which requires improvement. In this work, a novel adaptive simulated gorilla troop optimization is introduced which modifies the exploitation phase of the GTO. In the exploitation search process, GTO exploits two important mechanisms; follow silverback gorilla or male gorillas compete amongst themselves for females. These predominantly depends on the behaviour of silverback gorilla and are simulated randomly with variable C and W in the implementation of GTO. In this work, a new adaptive simulation is introduced to simulate the two mechanisms based on the state and behaviour of the silverback in the troop. This is done using equations (9) and (10). The proposed simulation leverages the state of the silverback gorilla and the search space in the exploitation phase to decide which of the two mechanisms gets to operate, rather than random simulation. In the proposed adaptive simulation, gorillas follow the silverback when $C \geq S$ and adult males gorillas compete for females when $C < S$. Hence the two mechanisms are simulated adaptively based on the valuof C and S.

$$S(t) = S_{\max} + (S_{\max} - S_{\min}) \times \frac{\cos(K(t)-1)}{\cos(K(t)+1)} \quad (9)$$

$$K(t) = \frac{fitness(silverback) - 1}{fitness(silverback) + 1} \quad (10)$$

In equation (9), $S_{\max}$ and $S_{\min}$ are maximum and minimum weights that determine the extent of silverback's behaviour at every iteration t. In this modification, $S_{\max}$ and $S_{\min}$ are chosen as 0.9 and 0.1 respectively, $\cos$ is the cosine function which reduces the effect of sudden changes of gorilla's behaviour over the course of iterations. The pseudocode of the new proposed algorithm is given below.

C. Self-adaptive and dynamic step-size

In the standard GTO updating formula, the random step typically involves a random number vector (Z) drawn from a uniform, Gaussian, or other distribution. However, this approach can lead to individual gorillas becoming trapped in local optima during the updating process, particularly for high-dimensional test functions where random turbulence can occur and slow down convergence. Consequently, the algorithm may fail to converge to an optimal value. To tackle this problem, a step modification factor ω is introduced to address any decrease in optimization accuracy in the standard GTO when the search dimension of an individual gorilla increases. This modification aims to enhance the algorithm's ability to handle higher-dimensional optimization problems effectively.

$$\omega = \beta^D * T * e^{(-t/T)} \quad (11)$$

t represents the current iteration, T represent the largest iteration D denotes the dimension of an individual gorillas, and $\omega \in [0,1]$. In this research, $\beta = 0.1$. In the operation of the algorithm, As the search dimension of an individual gorilla or the iteration number increases, the random step size undergoes a reduction. By diminishing the step size, the algorithm promotes a more focused exploration of the solution space, enabling the gorilla to navigate with greater precision and efficiency in higher-dimensional spaces or as the optimization process advances. This adaptive step size modulation optimizes the balance between exploration and exploitation for improved convergence towards optimal solutions.

IV. Experimental Setup For Testing the Proposed Modification

The effectiveness and accuracy of the proposed ASGTO were checked by testing it on 22 benchmark optimization functions in the literature. These functions consist of 10 unimodal and 12 multimodal functions, chosen for their varied complexity. Some details of the functions are listed in Table I. The proposed ASGTO was compared to the original GTO, grey wolf optimizer (GWO) and whale optimization algorithm. Table I shows the simulation parameters used for the various algorithms. All algorithms were run on the same computer, for all the functions. The specifications of the computer used are; Intel Core TM i7-10750 with CPU of 2.60 GHz and 16.0GB RAM and windows-based operating system. For each algorithm, first five separate runs were done for, each at 1000 iterations, and the following were recorded. The comparison is done on the basis of mean of best run, standard deviation of best run, global optimum, and average of global optimum of all runs.

TABLE I
SIMULATION DETAILS OF BENCHMARK OFTIMIZATION

SN	Function Name	Search Range	Optimum Value	Dimension
F_1	Ackley	[-32,32]	0	2
F_2	Beale	[-4.5,4.5]	0	2
F_3	Cross-in-Tray	[-10,10]	−2.06261218	2
F_4	Dekkers-Aarts	[-20,20]	−24771.09375	2
F_5	Easom	[-100,100]	-1	2
F_6	Goldstein-Price	[-2,2]	3	2
F_7	Happy Cat	[-2,2]	0	2
F_9	Matyas Function	[-10,10]	0	2
F_{10}	Powell Sum	[-1,1]	0	50
F_{11}	Quartic Function	[-1.28,1.28]	0	50
F_{12}	Rosenbrock	[-5,10]	0	50
F_{13}	Branin function	[-5, 10]	0..397887	2
F_{14}	Drop wave	[-10,10]	0	4
F_{15}	Eggholder	[-512,512]	-959.6407	10
F_{16}	Griewank	[-600,600]	0	4
F_{17}	Michalewicz	[0, Π]	-9.66	10
F_{18}	Rotated hyper ellipsoid	[-65.536,65.53]	0	10
F_{19}	Schwefel	[-500,500]	0	10
F_{20}	Shubert	[-10,10]	-186.73	10
F_{21}	Sphere	[-5.2,5.2]	0	10
F_{22}	Cross-in-Tray	[-1,1]	0	2

TABLE II
SIMULATION PARAMETERS OF ALGORITHMS

Algorithm	Parameter	Value
GTO	Max. no. iterations	1000
	W	0.8
	γ	0.03
ASGTO	Max. no. iterations	1000
	W	0.8
	γ	0.01
	S_{max}	0.9
	S_{min}	0.1
GWO	Max. no. of generations	1000
	Convergence constant a	[0,2]
WOA	Max. no. of generations	1000
	Convergence constant a	[0,2]
	Spiral factor b	1

V. Results and Discussions

A. Comparison of Global Optimum Values of Best Run

Table III compares results of the global optimum values of the best run obtained by ASGTO, GTO, GWO and WOA. It should be noted that, the best values are boldened. Comparing the globally optimum values of the benchmark functions to that obtained by ASGTO, the proposed ASGTO was able to attain the global optimum for eighteen (18) functions (F2, F3, F4, F5, F6, F8, F9, F11, F12, F13, F15, F17, F20, F21 and F22). However, the ASGTO could not attain the global optimum of functions F1, F7, F10 and F18 however the global optimum values attained were close to the expected values of the benchmark functions. On the other hand, GTO attained the global optimum of seventeen (17) functions excluding F1, F7, F10, F18 and F16 which it could not attain the expected values. Again, GWO was able to attain the exact optimum values for twelve (12) of the functions (F3, F4, F5, F6, F8, F9, F12, F13, F14, F17, F1 and F21) and could not attain the global optimum for the other ten (10) functions. Finally, WOA attained the global optimum value for thirteen (13) of the benchmark functions (F3, F4, F5, F6, F8, F9, F12, F13, F14, F15, F17, F19 and F21) and could not attain for nine (9) functions.

From the above discussions, ASGTO, GTO, GWO and WOA attained global optimum value for 82%, 77%, 55% and 59% of the 22 benchmark functions respectively. Again, ASGTO had the minimum optimum value for functions F7 and F10 and attained the same global optimum with GTO, GWO, WOA for F1. This clearly shows that, the proposed ASGTO outperformed the four (4) other algorithms in terms of attainment of global optimum values.

B. Mean and Standard Deviation of Best Run

Table IV compares results of the mean and standard deviation of convergence values of the best run amongst the five (5) separate runs. Here, the mean values are expected to be close to the global optimum values of the functions and the standard deviations are expected to be small as possible. This shows the consistency of the algorithms in attaining the global optimum values of the benchmark functions. From the table, ASGTO obtained the minimum mean and standard deviation values for fifteen (15) of the benchmark functions (F1, F2, F3, F5, F6, F10, F11, F12, F13, F14, F15, F16, F17, F20 and F21) representing 68% of the 22 functions. On the other hand, GTO, GWO and WOA had the minimum mean and standard deviation values for five (5), three (3) and one (1) representing 23%, 14% and 1% of the 22 benchmark functions respectively. However, ASGTO had a minimum mean for F16 and WOA had the minimum standard deviation for F16. It should be noted that the best values have been boldened. From the above discussion, it is clear that the proposed ASGTO was very consistent in attaining the global optimum of the functions and on this basis, the best algorithm in terms of mean and standard deviation of the best run.

C. Average of Global Optimum Values of all Runs

Table V compares the performances of ASGTO. GTO. GWO, WOA in terms of the average of global optimum values of all the five separate runs. This is meant to demonstrate the searchability of the algorithms to produce similar optimum values for each separate run. ASGTO and GTO were very consistent by producing average values same as the global optimal values of F1, F3, F4, F5, F6, F8, F9, F11, F12, F13, F14, F15, F17, F19, F20, F21 and F22 representing about 77% of the 22 benchmark functions. GWO also showed some level of consistency in attaining the global optimum values for the five separate runs by producing average values same as that of functions F3, F4, F5, F6, F8, F9, F12, F13, F14, F17, F18 and F21. This represents about 55% of the 22 benchmark functions. On the other hand, WOA showed the least level of consistency in searching for the global optimum values of the benchmark functions. It was able to attain average values same as the global optimum values of functions F3, F4, F6, F8, F9, F12, F14, F15, F17, F19 and F21 representing about 50% of the 22 benchmark functions. For functions F1, F7, F10, F16 and F18 which neither ASGTO nor GTO could attain average values same as their global optimum values, ASGTO had the closest average values for F7 and F10 whilst GTO had the closest for F16 and F18. All four algorithms had the same average value for function F1. Again, based on the discussion above, the proposed ASGTO has shown consistent and competitive performance to GTO, GWO and WOA.

D. Interval Plot of Convergence Values of Best Run

The interval plots of the benchmark functions are shown in Fig. 2 to Fig. 23. These are used to compare means of convergence values of the best run for each algorithm for 95% confidence interval (CI). It is also used to compare means of convergence values of the best run for each algorithm for 95% confidence interval (CI). It is also used to compare the variations in the convergence values of the algorithms and how consistent each was in attaining the optimum values of each benchmark function. The ASGTO showed significantly lowest variation in convergence values (optimum values) in the interval plot for all functions confirming its improved consistency and accuracy in attaining the convergence values. GTO also showed a low level of variations in all functions except for functions F1, F2, F4, F5, F6, F12 and F15 which it exhibited some high level of variations. On the other hand, GWO and WOA showed a very high variation in attaining the convergence values for almost all the functions except F4, F7, F8, F9, F16, F17 and F18. Both GWO and WOA showed very low level of variation in the interval plots for functions F7, F16, F17 and F18. Singularly, GWO showed low level of variation in functions F8 and F9 whilst WOA showed low level of variation in F4. Clearly, ASGTO is the better algorithm with consistency and low variation in the convergence values for the benchmark value, because the range of confidence interval or standard deviation of ASGTO for all 22 runs are significantly lower than the other three algorithms.the variations in the convergence values of the algorithms and how consistent each was in attaining the optimum values of each benchmark function. The ASGTO showed significantly lowest variation in convergence values (optimum values) in the interval plot for all functions confirming its improved consistency and accuracy in attaining the convergence values. GTO also showed a low level of variations in all functions except for functions F1, F2, F4, F5, F6, F12 and F15 which it exhibited some high level of variations. On the other hand, GWO and WOA showed a very high variation in attaining the convergence values for almost all the functions except F4, F7, F8, F9, F16, F17 and F18. Both GWO and WOA showed very low level of variation in the interval plots for functions F7, F16, F17 and F18. Singularly, GWO showed low level of variation in functions F8 and F9 whilst WOA showed low level of variation in F4. Clearly, ASGTO is the better algorithm with consistency and low variation in the convergence values for the benchmark value, because the range of confidence interval or standard deviation of ASGTO for all 22 runs are significantly lower than the other three algorithms.

TABLE III
COMPARISON OF GLOBAL OPTIMUM VALUE OF BEST RUN

SN	Optimum value	ASGTO	GTO	GWO	WOA
F_1	0	-8.88E-16	**-8.88E-16**	-8.88E-16	-8.88E-16
F_2	0	0.00E+00	0.00E+00	2.73E-10	3.16E-16
F_3	−2.06261218	-2.06E+00	-2.06E+00	-2.06E+00	-2.06E+00
F_4	−24771.09375	-2.48E+04	-2.48E+04	-2.48E+04	-2.48E+04
F_5	-1	-1.00E+00	-1.00E+00	-1.00E+00	-1.00E+00
F_6	3	3.00E+00	3.00E+00	3.00E+00	3.00E+00
F_7	0	2.75E-07	9.27E-06	1.84E-06	1.14E-01
F_8	0	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F_9	0	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F_{10}	0	6.69E-07	2.33E-05	1.69E-04	2.55E-05
F_{11}	0	0.00E+00	0.00E+00	1.06E-07	1.30E-11
F_{12}	0..397887	3.98E-01	3.98E-01	3.98E-01	3.98E-01
F_{13}	0	-1.00E+00	-1.00E+00	-1.00E+00	-1.00E+00
F_{14}	-959.6407	-9.60E+02	-9.60E+02	-9.60E+02	-9.60E+02
F_{15}	0	0.00E+00	0.00E+00	1.23E-02	0.00E+00
F_{16}	-9.66	-9.57E+00	-9.28E+00	-8.32E+00	-8.07E+00
F_{17}	0	2.36E+05	2.36E+05	2.36E+05	2.36E+05
F_{18}	0	1.44E-04	1.27E-04	1.43E+03	9.23E-03
F_{19}	-186.73	-1.87E+02	-1.87E+02	-1.87E+02	-1.87E+02
F_{20}	0	0.00E+00	0.00E+00	1.04E-186	6.49E-214
F_{21}	0	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F_{22}	0	0.00E+00	0.00E+00	1.18E-180	4.67E-20

TABLE IV
COMPARISON OF RESULTS OF BEST RUN

SN	Optimum value	ASGTO	GTO	GWO	WOA
F_1	Mean	5.26E-04	2.16E-03	1.15E-02	2.24E-02
	SD	7.64E-03	2.58E-02	1.77E-01	2.75E-01
F_2	Mean	7.16E-06	1.77E-03	3.28E-04	9.07E-04
	SD	1.16E-04	2.30E-02	7.38E-03	2.71E-02
F_3	Mean	-2.06E+00	-2.06E+00	-2.06E+00	-2.06E+00
	SD	7.24E-06	9.72E-06	1.09E-04	2.53E-04
F_4	Mean	-2.48E+04	-2.47E+04	-2.47E+04	-2.47E+04
	SD	9.88E+00	3.37E+02	3.37E+02	9.16E+00
F_5	Mean	-9.99E-01	-9.87E-01	-9.96E-01	9.71E-01
	SD	4.60E-03	4.51E-01	5.57E-02	6.43E-02
F_6	Mean	3.00E+00	3.00E+00	3.00E+00	3.01E+00
	SD	1.03E-04	2.58E-02	7.39E-02	1.88E-01

F_7	Mean	9.02E-03	6.59E-03	6.20E-04	1.21E-01
	SD	5.26E-02	4.41E-02	5.93E-03	3.98E-02
F_8	Mean	1.21E-06	3.95E-07	2.10E-06	3.54E-05
	SD	2.28E-05	5.32E-06	4.09E-05	1.08E-03
F_9	Mean	2.29E-168	1.07E-164	1.25E-184	3.19E-145
	SD	0.00E+00	0.00E+00	0.00E+00	7.14E-144
F_{10}	Mean	1.40E-03	2.05E-03	1.48E+00	8.62E-01
	SD	1.18E-02	3.20E-02	1.75E+01	1.35E+01
F_{11}	Mean	1.13E-04	3.23E-04	2.16E-03	2.37E-03
	SD	2.93E-03	3.15E-03	3.69E-02	7.27E-02
F_{12}	Mean	3.98E-01	3.98E-01	3.99E-01	3.98E-01
	SD	8.75E-04	3.88E-03	3.08E-03	2.52E-03
F_{13}	Mean	-1.00E-01	-1.00E-01	-1.00E-01	-9.99E-01
	SD	3.34E-04	4.19E-04	2.93E-03	1.08E-02
F_{14}	Mean	-9.60E+02	-9.59E+02	-9.59E+02	-9.59E+02
	SD	1.98E-02	6.45E-01	6.76E+00	1.27E+01
F_{15}	Mean	2.96E-03	6.13E-03	4.22E-02	7.96E-02
	SD	4.53E-02	7.65E-02	6.12E-01	4.63E-01
F_{16}	Mean	-9.37E+00	-9.02E+00	-5.91E+00	-7.83E+00
	SD	6.89E-01	7.36E-01	1.12E+00	5.60E-01
F_{17}	Mean	2.36E+05	2.36E+05	2.36E+05	2.36E+05
	SD	1.78E-09	1.78E-09	1.78E-09	1.78E-09
F_{18}	Mean	1.38E+01	3.69E+00	1.73E+03	4.48E+01
	SD	1.54E+02	6.94E+01	2.29E+02	2.08E+02
F_{19}	Mean	-1.87E+02	-1.87E+02	-1.87E+02	-1.87E+02
	SD	2.52E-01	2.34E-01	2.08E+00	3.29E+00
F_{20}	Mean	2.13E-07	3.31E-04	5.39E-02	1.09E-01
	SD	1.57E-06	6.07E-03	1.02E+00	1.71E+00
F_{21}	Mean	1.48E-08	1.21E-07	5.98E-07	1.34E-06
	SD	1.86E-07	1.43E-06	1.52E-05	3.84E-05
F_{22}	Mean	1.58E-03	2.76E-04	1.92E+01	1.82E+01
	SD	2.51E-02	2.34E-03	3.34E+02	3.04E+02

TABLE V
COMPARISON OF AVERAGE OF GLOBAL OPTIMUM VALUES OF ALL RUNS

SN	ASGTO	GTO	GWO	WOA
F_1	-8.88E-16	-8.88E-16	-8.88E-16	5.33E-16
F_2	0.00E+00	0.00E+00	3.89E-09	5.44E-14
F_3	-2.06E+00	-2.06E+00	-2.06E+00	-2.06E+00
F_4	-2.48E+04	-2.48E+04	-2.48E+04	-2.48E+04
F_5	-1.00E+00	-1.00E+00	-1.00E+00	-2.48E+04
F_6	3.00E+00	3.00E+00	3.00E+00	3.00E+00
F_7	4.28E-06	3.78E-05	4.83E-06	2.04E-01
F_8	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F_9	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F_{10}	5.39E-06	2.63E-05	4.17E-04	1.12E-03

F_{11}	0.00E+00	0.00E+00	4.17E-04	5.49E-09
F_{12}	3.98E-01	3.98E-01	3.98E-01	3.98E-01
F_{13}	-1.00E+00	-1.00E+00	-1.00E+00	-9.87E-01
F_{14}	-9.60E+02	-9.60E+02	-9.60E+02	-9.60E+02
F_{15}	0.00E+00	0.00E+00	4.46E-03	0.00E+00
F_{16}	-8.23E+00	-8.79E+00	-7.26E+00	-7.35E+00
F_{17}	2.36E+05	2.36E+05	2.36E+05	2.36E+05
F_{18}	2.86E-04	1.27E-04	1.14E+03	7.36E+02
F_{19}	-1.87E+02	-1.87E+02	-1.87E+02	-1.87E+02
F_{20}	0.00E+00	0.00E+00	3.10E-178	1.35E-207
F_{21}	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F_{22}	0.00E+00	0.00E+00	7.30E-179	9.79E-205

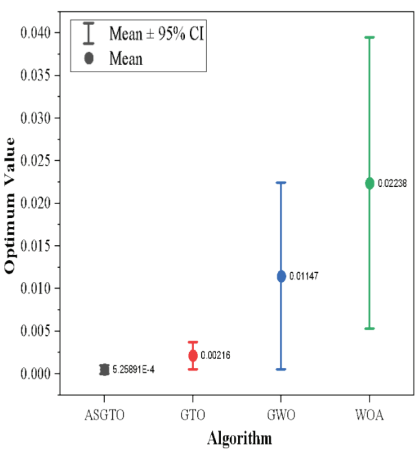

Fig. 2. Interval Plot of Convergence for Ackley (F1)

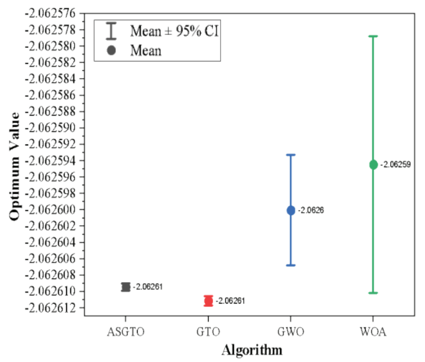

Fig. 4. Interval Plot of Convergence for Cross-in-Tray (F3)

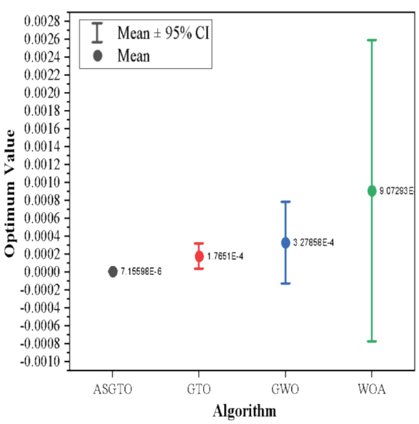

Fig. 3. Interval Plot of Convergence for Beale (F2)

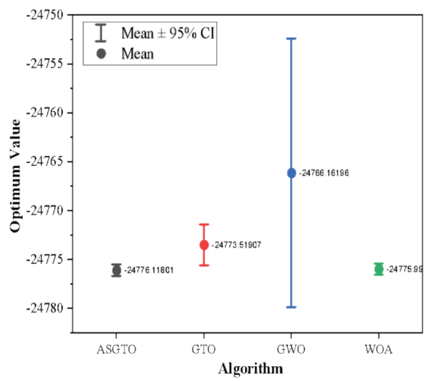

Fig. 5. Interval Plot of Convergence for Dekkers-Aarts (F4)

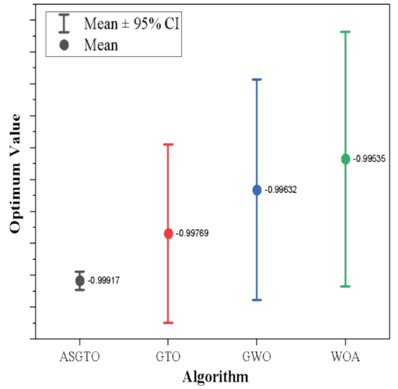

Fig. 6. Interval Plot of Convergence for Easom(F5)

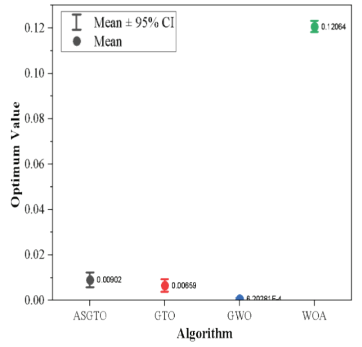

Fig. 9. Interval Plot of Convergence for Matyas Function (F8)

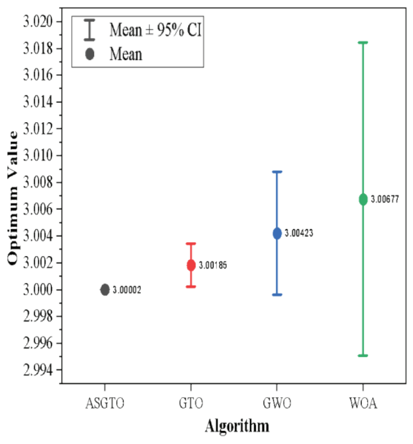

Fig. 7. Interval Plot of Convergence for Goldstein-Price (F6)

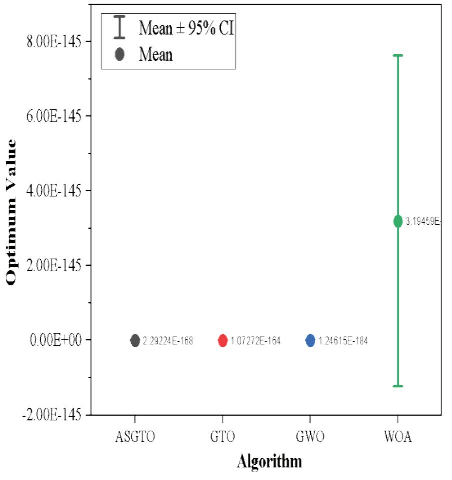

Fig. 10. Interval Plot of Convergence for Powell Sum (F9)

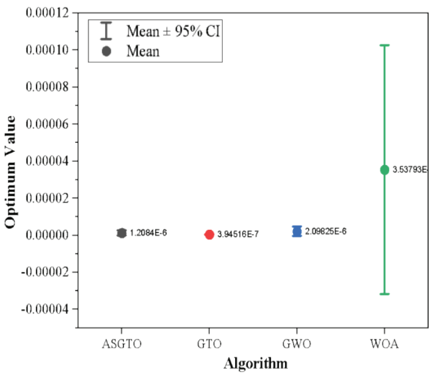

Fig. 8. Interval Plot of Convergence for Happy Cat (F7)

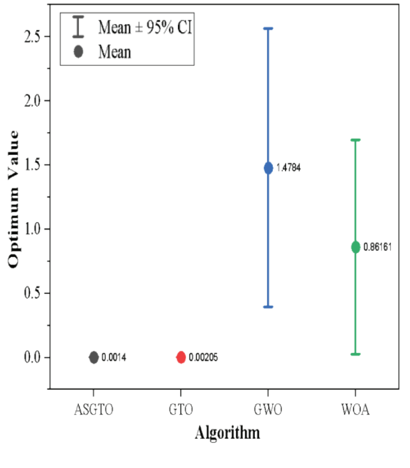

Fig. 11. Interval Plot of Convergence for Quartic Function (F10)

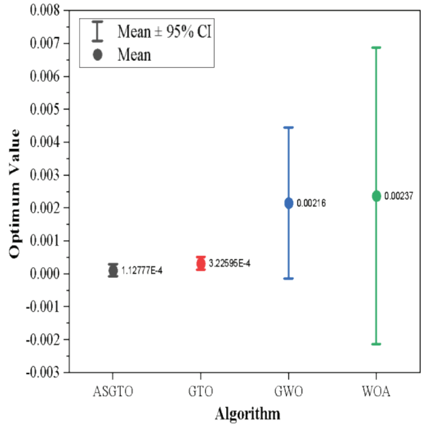

Fig. 12. Interval Plot of Convergence for Rosenbrock (F11)

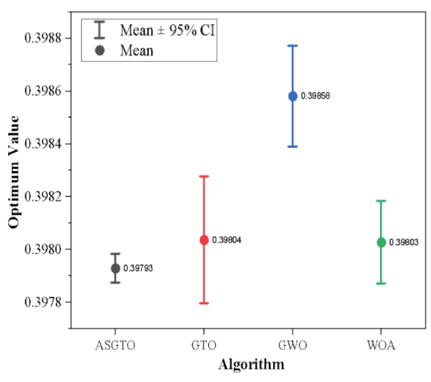

Fig. 13. Interval Plot of Convergence for Branin function (F12)

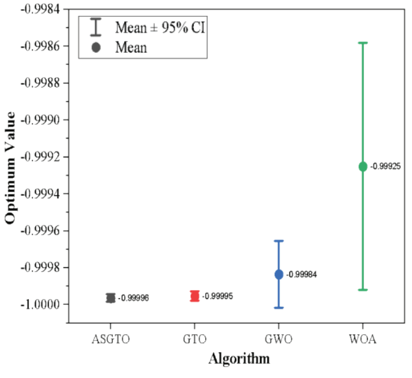

Fig. 14. Interval Plot of Convergence for Drop wave (F13)

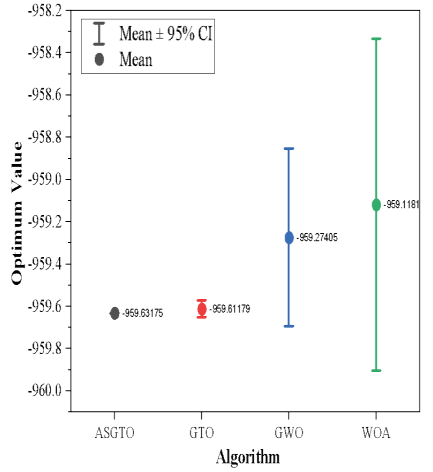

Fig. 15. Interval Plot of Convergence for Eggholder (F14)

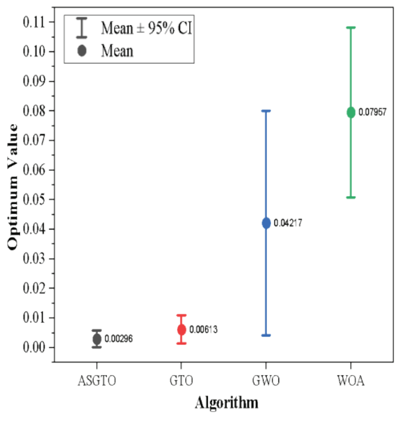

Fig. 16. Interval Plot of Convergence for Griewank (F15)

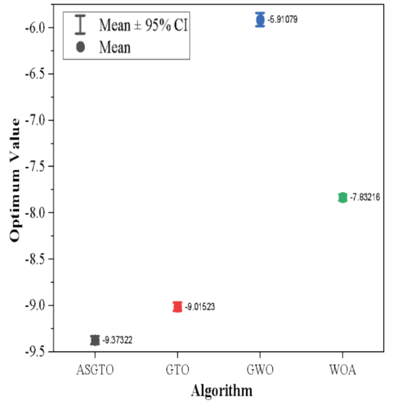

Fig. 17. Interval Plot of Convergence for Michalewicz (F16)

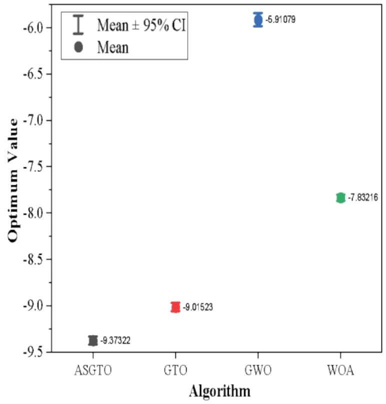

Fig. 17. Interval Plot of Convergence for Michalewicz (F16)

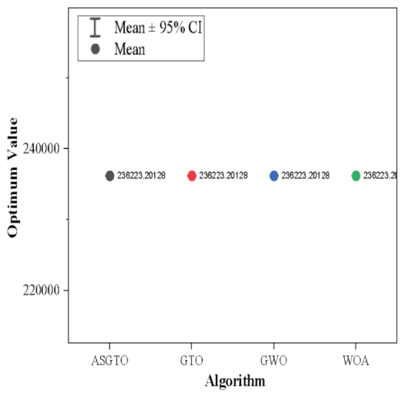

Fig. 18. Interval Plot of Convergence for Rotated hyper ellipsoid (F17)

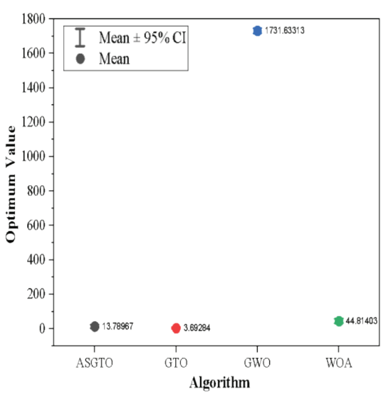

Fig. 19. Interval Plot of Convergence for Schwefel (F18)

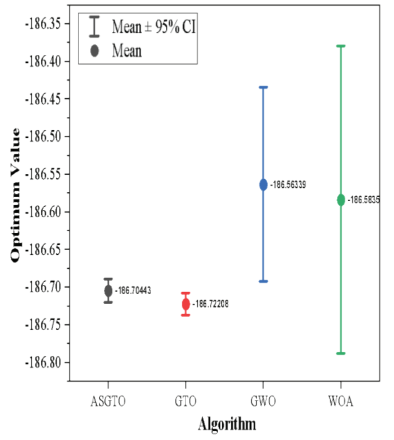

Fig. 20. Interval Plot of Convergence for Shubert (F19)

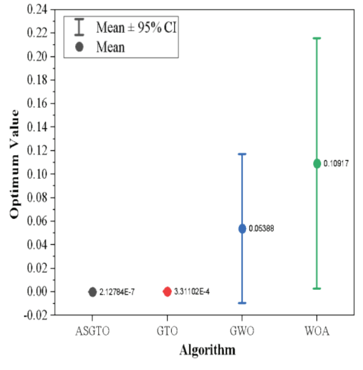

Fig. 21. Interval Plot of Convergence for Sphere (F20)

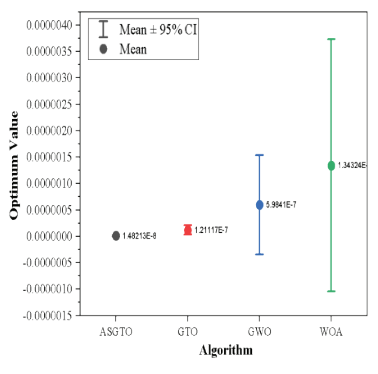

Fig. 22. Interval Plot of Convergence for Sum of difference (F21)

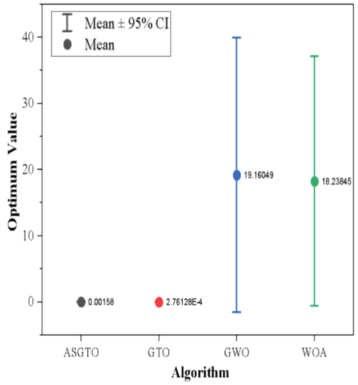

Fig. 23. Interval Plot of Convergence for Sum square (F22)

VI. Conclusion

A novel modification to the artificial gorilla troop optimizer (GTO) has been proposed to improve its performance. The modified version of the GTO is named adaptive simulated gorilla troop optimizer (ASGTO). An adaptive simulation based on the current state of the best gorilla (silverback) in the troop is proposed in the exploration phase of the GTO to help it overcome the problem of local optima entrapment and also improve its optimal solutions. The proposed ASGTO has been compared with well know algorithms in the literature; grey wolf optimizer (GWO), whale optimization algorithm (WOA) and the GTO using twenty-two benchmark optimization functions. The performance of the algorithms is assessed in terms of global optimum value of best run, mean of best run, the standard deviation of best run, average of optimum value of all runs and interval plot of convergence values of the best run to show the statistical difference in performance between the algorithms. The results obtained shows clearly that, the proposed ASGTO outperformed GTO, GWO and WOA in about 82% of the 22 benchmark functions in terms of global optimal value, 68% in reference to the mean and standard deviation of best run and 77% in relation to the average of optimum values of all runs. The ASGTO also showed the lowest variation in convergence values of the best run for all functions. This indicates a significant improvement in the GTO and the ability of the ASGTO to avoid local optima entrapment. The proposed ASGTO is recommended for use by researchers to solve optimization problems in diverse fields due to its excellent performance. The ASGTO is currently being used to solve optimal capacitor placement problems in the electrical distribution system.

Conflict of Interest

The authors declare no conflict of interest in the publication of this research article.

Author Contributions

A. Bright conceptualized the research, performed the simulation, and drafted the original paper. T. Elvis analyzed the results and assisted in the data curation, methodology, and drafting of the paper, A. K. Emmanuel and F. A. Emmanuel fully supervised the research, and reviewed and edited the paper.

References

[1] B. Abdollahzadeh, F. Soleimanian Gharehchopogh, and S. Mirjalili, "Artificial gorilla troops optimizer: A new nature-inspired metaheuristic algorithm for global optimization problems," Int. J. Intell. Syst., vol. 36, no. 10, pp. 5887–5958, 2021.

[2] J. You et al., "Modified Artificial Gorilla Troop Optimization Algorithm for Solving Constrained Engineering Optimization Problems," Mathematics, vol. 11, no. 5, 2023, doi: 10.3390/math11051256.

[3] Mirjalili S, Mirjalili SM, Lewis A. Grey wolf optimizer. Adv Eng Software. 2014; 69:46-61.

[4] Mirjalili S, Lewis A. The whale optimization algorithm. Adv Eng Software. 2016; 95:51-67.

[5] Kennedy, J.; Eberhart, R.C. Particle swarm optimization. In Proceedings of the IEEE International Conference on Neural Networks, Perth, WA, Australia, 27 November–1 December 1995; pp. 1942–1948.

[6] Yang, X.S. Firefly algorithms for multimodal optimization. In Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2009; Volume 5792, pp. 169–178.

[7] Price, K.; Storn, R.; Lampinen, J. Differential Evolution: A Practical Approach to Global Optimization; Springer: Berlin/Heidelberg, Germany, 2005.

[8] G. Dhiman and V. Kumar, "Spotted hyena optimizer: A novel bio-inspired based metaheuristic technique for engineering applications," Adv. Eng. Softw., vol. 114, pp. 48–70, 2017, doi: 10.1016/j.advengsoft.2017.05.014.

[9] A. H. Gandomi, X. S. Yang, and A. H. Alavi, "Erratum: Cuckoo search algorithm: A metaheuristic approach to solve structural optimization problems (Engineering with Computers DOI:10.1007/s00366-011-0241-y)," Eng. Comput., vol. 29, no. 2, p. 245, 2013, doi: 10.1007/s00366-012-0308-4.

[10] L. Abualigah, M. A. Elaziz, P. Sumari, Z. W. Geem, and A. H. Gandomi, "Reptile Search Algorithm (RSA): A nature-inspired meta-heuristic optimizer," Expert Syst. Appl., vol. 191, no. November 2021, p. 116158, 2022, doi: 10.1016/j.eswa.2021.116158.

[11] S. Mirjalili, "SCA: A Sine Cosine Algorithm for solving optimization problems," Knowledge-Based Syst., vol. 96, pp. 120–133, 2016, doi: 10.1016/j.knosys.2015.12.022.

[12] Geem ZW, Kim JH, Loganathan GV. A new heuristic optimization algorithm: harmony search. Simulation. 2001;76(2):60-68.

[13] A. Ginidi, S. M. Ghoneim, A. Elsayed, R. El-Sehiemy, A. Shaheen, and A. El-Fergany, "Gorilla troops optimizer for electrically based single and double-diode models of solar photovoltaic systems," Sustain., vol. 13, no. 16, 2021.

[14] M. A. El-Dabah, S. Kamel, M. Khamies, H. Shahinzadeh, and G. B. Gharehpetian, "Artificial Gorilla Troops Optimizer for Optimum Tuning of TID Based Power System Stabilizer," 2022 9th Iran. Jt. Congr. Fuzzy Intell. Syst. CFIS 2022, no. 2, pp. 1–5, 2022, doi: 10.1109/CFIS54774.2022.9756463.

[15] Y. Xiao, X. Sun, Y. Guo, S. Li, Y. Zhang, and Y. Wang, "An Improved Gorilla Troops Optimizer Based on Lens Opposition-Based Learning and Adaptive β-Hill Climbing for Global Optimization," Comput. Model. Eng. Sci., vol. 130, no. 3, pp. 1–36, 2022.