IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 17, 2022, accepted January 26, 2022, date of publication January 31, 2022, date of current version February 14, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3147821

Ebola Optimization Search Algorithm: A New Nature-Inspired Metaheuristic Optimization Algorithm

OLAIDE NATHANIEL OYELADE[1], ABSALOM EL-SHAMIR EZUGWU[1], TEHNAN I. A. MOHAMED[1], AND LAITH ABUALIGAH[2,3]

[1]School of Mathematics, Statistics, and Computer Science, University of KwaZulu-Natal, Pietermaritzburg Campus, Pietermaritzburg, KwaZulu-Natal 3201, South Africa
[2]Faculty of Computer Sciences and Informatics, Amman Arab University, Amman 11953, Jordan
[3]School of Computer Sciences, Universiti Sains Malaysia, George Town, Pulau Pinang 11800, Malaysia

Corresponding author: Absalom El-Shamir Ezugwu (ezugwua@ukzn.ac.za)

ABSTRACT Nature computing has evolved with exciting performance to solve complex real-world combinatorial optimization problems. These problems span across engineering, medical sciences, and sciences generally. The Ebola virus has a propagation strategy that allows individuals in a population to move among susceptible, infected, quarantined, hospitalized, recovered, and dead sub-population groups. Motivated by the effectiveness of this strategy of propagation of the disease, a new bio-inspired and population-based optimization algorithm is proposed. This study presents a novel metaheuristic algorithm named Ebola Optimization Search Algorithm (EOSA) based on the propagation mechanism of the Ebola virus disease. First, we designed an improved SIR model of the disease, namely SEIR-HVQD: Susceptible (S), Exposed (E), Infected (I), Recovered (R), Hospitalized (H), Vaccinated (V), Quarantine (Q), and Death or Dead (D). Secondly, we represented the new model using a mathematical model based on a system of first-order differential equations. A combination of the propagation and mathematical models was adapted for developing the new metaheuristic algorithm. To evaluate the performance and capability of the proposed method in comparison with other optimization methods, two sets of benchmark functions consisting of forty-seven (47) classical and thirty (30) constrained IEEE-CEC benchmark functions were investigated. The results indicate that the performance of the proposed algorithm is competitive with other state-of-the-art optimization methods based on scalability, convergence, and sensitivity analyses. Extensive simulation results show that the EOSA outperforms popular metaheuristic algorithms such as the Particle Swarm Optimization Algorithm (PSO), Genetic Algorithm (GA), and Artificial Bee Colony Algorithm (ABC). Also, the algorithm was applied to address the complex problem of selecting the best combination of convolutional neural network (CNN) hyperparameters in the image classification of digital mammography. Results obtained showed the optimized CNN architecture successfully detected breast cancer from digital images at an accuracy of 96.0%. The source code of EOSA is publicly available at https://github.com/NathanielOy/EOSA_Metaheuristic.

INDEX TERMS Ebola virus, metaheuristic algorithm, optimization problems, constrained benchmark functions, image classification, convolutional neural network.

I. INTRODUCTION

Ebola virus represents the virus causing the Ebola virus disease (EVD). The disease was first so named in the Democratic Republic of the Congo (DRC) in 1976. A widespread catastrophic outbreak was reported in late 2013 in the West African regions, including Sierra Leone, Liberia, Mali, Nigeria, and Senegal. It is widely reported that the virus made its entry into the human population through consumption or contact with infected animals such as fruit bats [1]–[3]. This animal-to-human infection led to person-to-person infection, becoming an epidemic across the West African region.

Contrary to the novel corona virus (COVID-19), the EVD person-to-person transmission occurs only when the infected

The associate editor coordinating the review of this manuscript and approving it for publication was Zhenzhou Tang.

person exhibits some form of signs and symptoms associated with Ebola. This transmission is aided by contact with any form of body fluid of an infected person. A healthy person comes in contact with infected objects since the Ebola virus can survive on dry surfaces such as doorknobs and countertops for several hours [4], [5]. The hemorrhagic disease, known to be notoriously fatal, has been reported to have mortality rates ranging from 25% to 90%, with an average of 50% mainly due to fluid loss rather than blood loss [6], [7]. Although the experimental Ebola vaccine proved highly protective against EVD, the transmission rate from the infected to the susceptible population is alarming. The high survival rate of EBOV in body fluids, including breast milk, saliva, urine, semen, cerebrospinal fluid, aqueous humor, blood, blood derivatives, and detected in amniotic fluid, tears, skin swabs, and stool by reverse transcription (RT)-PCR, presents a very high infection and transmission rate. This implies that a one-time virus entry into a susceptible population through a single individual has a high propagation rate.

A close study of the propagation strategy of the EVD and the resulting propagation model inspired the metaheuristic algorithm proposed in this study. Deriving computational solutions from natural phenomena has promoted a field of computing referred to as nature-inspired computing. A broader view of this aspect of computing may well relate to the field of Artificial Intelligence (AI) and Computational Intelligence (CI), where computational systems are designed by synthesizing behaviors of organisms or natural phenomena [8]–[10]. Metaheuristic algorithms are nature-inspired optimization solutions with high performance. They often require low computing capacity, which has successfully solved complex real-life problems in engineering, medical sciences, and sciences, especially in areas concerning swarm intelligence based algorithms [11]–[20]. These optimization algorithms are designed without specific reference to a particular problem. They are often categorized by performing a local or global search, handling single-solutions or whole populations, using memory, and adopting a greedy or iterative search process. The techniques often achieve near-optimal solutions to large-scale optimization problems due to their highly flexible manner of operation and ability to learn quickly owing to their natural or biological systems from which their designs were inspired.

A subfield of natural computation consists of biology-inspired techniques, also referred to as bio-inspired algorithms or computational biology. These techniques are stochastic, far from the design of deterministic heuristics. This feature has made it possible to represent the biological evolution of nature, hence capable of being used as a global optimization solution. Recently, bio-inspired optimization algorithms have helped support machine learning to address the optimal solutions to complex problems in science and engineering [21]. The bio-inspired algorithms combine biological concepts with mathematics and computer sciences and are classified as Evolutionary Algorithms (EA), Biology, and Swarm Intelligence (SI). Although the last two categories are often combined and referred to as swarm intelligence, not all bio-inspired algorithms have the swarm feature. Examples of evolutionary algorithms are Genetic Algorithms (GA) [22], Genetic Programming (GP), Differential Evolution (DE), the Evolution Strategy (ES), Coral Reefs Optimization Algorithm (CRO) [23], and Evolutionary Programming (EP). Examples of SI-based algorithms are: food foraging behavior of honeybees Artificial Bee Colony (ABC) [24], [25], echolocation ability Ant-lion Optimizer (ALO), luciferin induced glowing behavior Bees Algorithm (BAO), Bat Algorithm (BOA) [26], hunting behavior Barnacles Mating Optimizer (BMO), swarming around hive by honey bees Cuckoo Optimization Algorithm (COA), echo-cancellation Cuckoo Search Optimization (CSO) [27], hunting behavior and social hierarchy Dolphin Echolocation Optimization (DEO), social interaction and food foraging Dragonfly algorithm (DFA), Static and dynamic swarming behavior Deer Hunting Optimization (DHO), Pollination process of flowers Fire-fly Algorithm (FFA), Food foraging behavior Hunting search (FFO), bubble-net hunting Fruit Fly Optimization Algorithm (FOA), obligate brood parasitic behavior Flower-Pollination Algorithm (FPA), navigation and foraging behaviors Grasshopper Optimization Algorithm (GOA), spiral flying path of moth Glowworm Swarm Optimization (GSO), cuckoos' survival efforts Grey Wolf Optimizer (GWO) [28], flashing light patterns Moth-Flame Optimization (MFO), Mating behavior Manta Ray Foraging Optimizer (MRFO), Hunting behavior of humans SailFish optimizer (SFO), Group hunting behavior Salp Swarm Algorithm (SSA), and Hunting mechanism of Whale Optimization Algorithm (WOA) [29]. Others are Blue Monkey Optimization (BMO) [30], Arithmetic Optimization Algorithm (AOA) [31]–[33], Aquila Optimizer (AO) [34], Reptile Search Algorithm (RSA) [35], and Sandpiper Optimization Algorithm (SOA) [36].

These EA and SI-based algorithms have demonstrated good performance in solving real-world complex combinatorial problems, which are considered a fundamentally vital and critical task. In addition, studies have shown their capability to efficiently scale up to handle large-scale problems as opposed to traditional optimization methods, which are more effective for small-scale problems [37]. Further research in bio-inspired computing areas will lead to achieving similar and better new optimization algorithms capable of solving modern-day optimization problems. Our study showed that exploring the propagation model of diseases with endemic and pandemic natures may yield an outperforming optimization algorithm with interesting performance in solving real-world optimization problems. This study considered that optimization algorithms' exploration and exploitation phases are practically coupled into the natural order and strategy of propagation of these diseases. Studies confirm that finding a good balance between exploitation and exploration of the problem search space for an optimization algorithm determines its ability to find a globally optimal solution [38], [39]. The exploration phase often allows for finding

candidate solutions that are not neighbor to the current solution, while exploitation maintains its search in the neighborhood. Hence, we found a balance of the two scenarios in the disease propagation model for escape from a local optimum without neglecting good solutions in the neighborhood.

In this study, we propose a novel metaheuristic algorithm referred to as the Ebola Optimization Search Algorithm (EOSA), inspired by the Ebola virus disease and its propagation model (a preprint has previously been published [40]). We derived the novel algorithm through a careful study of our implementation of the SIR model of the disease. Particularly, our algorithm's novelty brought into metaheuristic design lies in the mechanism to balance between the exploration and exploitation phases. Secondly, the algorithm demonstrates an inherent ability to use a dynamic mechanism to update solutions as they transit through susceptible profitably, infection, quarantine, recovered, and hospitalized compartments. Initialization of solutions in the population follows the natural pattern of the disease through the application of a stochastic model. To quantitatively measure how fit a given solution is in solving the problem, give intuitive results, and discover the best or worst candidate solution, the resulting optimization algorithm is investigated on about forty-seven (47) classical benchmark optimization functions [41] and more than thirty (30) CEC functions [42]. In summary, the main contributions of this research are as follows:

i. An improved SIR model of Ebola disease and a modified mathematical model is designed to aid the proposed algorithm.
ii. We design a new nature-inspired metaheuristic algorithm using the models in (i).
iii. Applied EOSA to optimize the hyperparameters of a CNN architecture to image classification problem detecting breast cancer.
iv. Several experiments are conducted using over 89 mathematical optimization problems, including the classical benchmark functions and IEEE-CEC test suite, which are considered challenging test problems in the literature to evaluate the efficiency of the proposed EOSA.
v. Validation of the obtained numerical results using statistical analysis test further supports the superiority claim of the proposed EOSA optimization method over the existing state-of-the-art metaheuristic algorithms.

The rest of the paper is organized as follows. Section 2 provides an overview of the Ebola virus diseases. The proposed propagation model, mathematical model, and algorithmic design for the EOSA algorithm are given in Section 3. Section 4 details the benchmark functions applied to evaluate the performance of the proposed algorithm. Also, this section lists and discusses the parameterization and assignment of initial values used for experimentation. A discussion on results obtained is presented in Section 5, including numerical simulations that support the proposed propagation model. A detailed comparative analysis of the performance of EOSA and similar algorithms is also presented in the section. In Section 6, we give concluding remarks on how our novel optimization algorithm fits in the literature, its real-life applicability, and perspectives for future works.

II. RELATED WORK

This section summarises the Ebola virus disease, its propagation technique, and relevant SIR-based models that support this study. Also, considering the nature of the optimization algorithm proposed in this study which shares some principles of biology, we review studies that have developed bio-inspired optimization algorithms.

A. THE EBOLA VIRUS (EBOV) AND EBOLA VIRUS DISEASES (EVD): THE PROPAGATION MECHANISM

Ebola viruses result in what is known as the Ebola virus disease (EVD) once they successfully infect the host, suggesting victimization of the host. They are classified among the family of Filoviridae viruses, which are recognized by their different shapes of short or elongated branched filaments sizing up to 14,000 nanometers in length [6]. About six different species of the EBOV have been reported to exist. Bundibugyo Ebola virus, Ebola-Zaire virus, Tai Forest Ebola virus, and Sudan Ebola virus account for large flare-ups or outbreaks in Africa.

Exposure of a human individual to the virus through pathogenic agents or a contaminated environment initiates a population-based infection and after that, propels the spread of the disease. Direct contact with infected individuals spurs the propagation and spread of the virus. This contact relies on broken skin or mucous membranes in the eyes, nose, mouth, or other openings. It is assumed that such openings in the human body allow for body fluids (e.g. urine, saliva, sweat, faeces, vomit, breast milk, amniotic fluid, blood, and semen) bearing the virus to be transmitted to other susceptible individuals. Another host to the Ebola virus, which may transmit the disease to a healthy or susceptible individual, is a contaminated environment. An environment, such as medical equipment, clothes, bedding, and other related utensils, is considered contaminated if the body fluid of an infected individual has been spilt within or upon such an environment or object. Whereas an infected individual and a contaminated environment appear to have enhanced the propagation of EVD, infected animals consumed by humans have also been shown to propagate the disease [43]. These animals include bats, chimpanzees, fruit bats, and forest antelope, often hunted for food. Another propagative mechanism of the EBOV is culturally driven by burial practices in most affected populations and regions, with transmission occurring through contact with infected dead bodies. Meanwhile, note that the Ebola virus is not propagated through the air.

The application of different strategies, including case-based management approach, surveillance and contact tracing, quarantine of infected cases, infection prevention and control practices, and safe burial rites, has been adopted to revive and survive infected cases. However, infected cases remain positive while the virus remains in their blood. The infection and propagation rate of the EBOV presents an

appealing computational solution to numerous problems and so motivates the design of the proposed metaheuristic algorithm. While it appears that the solutions for mitigating the spread of the virus are suggestive of scaling down the infection rate, we argue that some other factors are still contributory to the propagation model. For instance, it is widely reported that the time-scale from symptom onset to death is an average of 10 days in 50–90% of cases [44].

To formalize and apply the propagation model of EBOV, we review some susceptible-infection-recovery (SIR) models. This is necessary for mainstreaming the concept proposed in the study. An interesting SIR model, based on EBOV, combining agent-based and compartmental models, has been presented [5]. The authors suggested that the hybrid model can switch from one paradigm to another on a stochastic threshold. The agent-based model consists of Susceptible (S), Infected (I), Hospitalized (H), Recovered (R), Funeral (F), and Dead (D). The Exposed (E) item was added to make up seven (7) compartments in the compartmental model. The SIR-based model was proposed to model the movement of individuals in a population from one compartment to another in both paradigms. For instance, individuals may move from Susceptible (S), Infected (I), to Hospitalized (H), based on a pre-existing computed rate. One external compartment considered in the literature is the influence of EBOV-carrying animals like bats. The assumption made was that since these animals can infect the human population without them (the animals) becoming ill, they present a reservoir-like mechanism for the virus in the SIR model.

Furthermore, the authors assumed that the rate of infection and hospitalization between infected individuals who will recover or die is the same, the deceased individual is buried in unsafe practices, and that recovered individuals are removed from the system. This SIR model presents a foundation for the modeling and implementation of the optimization algorithm proposed in this study. We considered that the compartments defined by Tanade *et al.* (year) work demonstrate the possibility to monitor and simulate the propagation model of the EBOV for the optimization task in our study.

In related work, Berge *et al.* (year) also modeled the propagation model of EBOV using the SIR-type model. The novelty of the study was the addition of the role of the indirect environmental transmission on the dynamics of EVD and to assess the effect of such a feature on the long run of the disease [45]. The authors showed that factoring direct and indirect transmission of EBOV into an SIR model promotes a system where the virus always exists in a population, increasing the propagation rate. Taking a cue from the novelty of this work in addition to that of Tanade *et al.* (year), we adapted the model proposed in this study to support the concept of direct and indirect transmission promoted by Berge *et al.* (year). Both studies supported their SIR models with mathematical models and further simulation to validate the performance of their model.

Similarly, Yet [46] successfully represented the basic interactions between EBOV and wild-type Vero cells in vitro . Rafiq *et al.* (year) also proposed the SEIR model, which mathematical model supported demonstrating the dynamics and illustrating the stability pattern of the Ebola virus in the human population. Their mathematical model is in the form of a couple of linear differential equations. The authors applied their SEIR model to study the disease-free equilibrium (DFE) and endemic equilibrium (EE) to report the stability of the model. Another study investigating the spread of EVD in India is [47] hoping to find EBOV transmission in the region through an SEIR model. Using ordinary differential equations, the study represented the SEIR model as a mathematical model and simulated it using a spatiotemporal epidemiology modeler (STEM). Rachah and Torres [47] also applied a mathematical model to study the outbreak of EBOV and eventually the EVD. The novelty of this study is the addition of vaccination to the proposed model. We found this appealing considering the role of the vaccine in stemming the tide of the infected population. Whereas most SEIR approaches have often adopted the stochastic method for the simulation of the model, Okyere *et al.* [48] considered using a deterministic scheme for designing models and studying the infection rate of EBOV. As an improvement to the work of Rachah and Torres (year), which factored in vaccination, the study also captured treatment and educational campaigns as time-dependent control functions in the SEIR model proposed.

This study developed a comprehensive SEIR-based model with more compartments, considering the above review. The proposed SEIR model factored in the notion of quarantine, which we found to play a role in curtailing EBOV propagation. In addition, we modeled the SEIR model to allow for the inclusion of the influence of vaccines in the pace of the growth of infection among a given population. The SEIR model was then formulated using an ordinary differential equation. This presented a good understanding of the design of the proposed metaheuristic algorithm. The resulting model is detailed in Section 3 and its supporting mathematical model.

B. METAHEURISTIC OPTIMIZATION ALGORITHMS: BIOINSPIRED-BASED ALGORITHMS

Bio-inspired optimization algorithms represent a class of metaheuristic algorithms whose principles are inspired by biology and natural phenomena. Generally, these algorithms have successfully been applied to solve different optimization problems in engineering and other related fields [49]. This category of algorithms exploits the basic processes of nature and then translates them into rules or procedures, which are then modeled computationally for solving complex real-life problems [50]–[57]. They are mostly population-based algorithms, and examples of such are Satin Bowerbird Optimizer (SBO), Earthworm Optimisation Algorithm (EOA), Wildebeest Herd Optimization (WHO), Virus Colony Search (VCS), Slime Mould Algorithm (SMA), Invasive Weed Colonization Optimization (IWO), Biogeography-Based Optimization (BBO), Coronavirus Optimization

Algorithm (COA), Emperor Penguin Salp Swarm Algorithm (ESA). Although evolutionary-based algorithms like GA and DE and swarm-based algorithms like PSO, WOA, and ABC share some characteristics of a biology-inspired algorithm, we have chosen to limit our review to those listed.

ESA is a hybrid of two phenomena drawn from the Salp Swarm Algorithm and Emperor Penguin. The behaviour of the two creatures is modelled to achieve ESA. Comparing the proposed algorithm with similar metaheuristic algorithms, authors [58] revealed that the algorithm demonstrated good performance based on sensitivity, scalability, and convergence analyses. Coronavirus Optimization Algorithm (COA) based on its propagation strategy, and another variant, namely Coronavirus Herd Immunity Optimizer (CHIO) based on human immunity, has been proposed. The COA proposed in [59] and CHIO in [60] leveraged infection and herd immunity. The effectiveness of COA was evaluated by applying it to the design of the convolutional neural network (CNN) problem, while CHIO proved robust at real-world engineering problems. Earthworm Optimisation Algorithm (EOA), also referred to as EWA, is a metaheuristic algorithm whose inspiration was drawn from the reproductive nature of the earthworm [61]. The mechanism involves two reproduction strategies where the first strategy allows for a parent to reproduce only one offspring while the other allows for more than one offspring. This reproducibility is controlled by the Cauchy mutation approach allowing for crossover operators.

Biogeography-Based Optimization (BBO) solves its optimization problem by implementing the geographical distribution and positioning of biological organisms [62]. Alluding to the fact that BBO's features are similar to those of GAs, the authors drew inspiration from the original mathematical model of the biogeography of organisms to derive BBO. Experimentation shows that BBO successfully solved real-world sensor selection problems to detect the status of aircraft engines and a selection of 14 benchmark optimization functions. Invasive Weed Optimization (IWO) is an optimization algorithm that has been widely applied to numerous problems and is based on numerical stochastic optimization algorithms learnt from the invasive nature of weeds [63]. The aggressive invasive nature of weeds allows for colonizing the environment against other economically viable plants. Knowing that this is a disadvantage agricultural-wise, the concept has benefited from solving optimization problems. The resulting algorithm was successfully applied to engineering problems, namely optimization and tuning the robust controller and well-known benchmark functions. Satin Bowerbird Optimization (SBO) is a biology-based optimization algorithm whose inspiration was drawn from the phenomenon of the male satin bowerbird's capability of attracting the female for breeding. [64]. The Satin Bowerbird Optimizer (SBO) algorithm has been successfully applied to the optimization problem in estimating the efforts needed to develop software. Wildebeest Herd Optimization (WHO) is a bio-inspired metaheuristic algorithm rooted in the behavior of wildebeest when searching for food [65]. A lookout for grazing land often guides the search with a high vegetation density. The WHO exploits the following natural characteristics of herds of wildebeest to achieve its performance: local search capability of wildebeest due to limited eyesight, look out for sparsely grazed region to avoid crowded grazing, exploitation of past experiences to explore regions with a high density of vegetation, starvation avoidance strategy deployed through the transition to new regions or location, and lastly, herd-based movement to avoid predators.

The propagation strategy of the virus in the host environment can sometimes be aggressive and often overwhelm the whole environment. Authors [39] proposed Virus Colony Search (VCS) motivated by this mechanism. The VCS exploration and exploitation phases leverage the propagation approach of the virus through diffusion or infection of the host environment. VCS has been successfully applied to the classic benchmark functions and the modern CEC2014 benchmark functions and real-life problems regarding energy consumption management [66]. Slime Mould Algorithm (SMA) optimization algorithm is based on a fungus named slime mould, which inhabits cold and humid places [67]. The algorithm's authors explored the nutritional stage, also referred to as plasmodium of the organism, for its design. They have a mechanism for multiple food sources and at the same time form a connected venous network so that they can even grow to more than 900 square centimeters depending on food availability. Using a mathematical model, the authors were able to simulate the process of producing positive and negative feedback of the propagation wave of slime mould based on bio-oscillator to form the optimal path for connecting food with excellent exploratory ability and exploitation propensity. SMA was successfully applied to solve engineering problems, including cantilever, welded beam, and pressure vessel structure problems.

While we acknowledge that these are not exhaustive and many new optimization algorithms inspired by natural processes are being developed, they provide the reader with a general understanding of the inspiration and principle behind such a class of algorithms.

III. METHODOLOGY: EOSA METAHEURISTIC ALGORITHM

Understanding how an SEIR-based model works in the propagation of a disease is important to appealing for the design of an optimization algorithm. Hence, this section presents an improved SEIR model based on recent literature on EVD. Secondly, a presentation of the procedural flow of EOSA and the corresponding flow chart are presented and discussed. Lastly, to formalize the proposed optimization algorithm, we represent the SEIR model using a mathematical model and then the algorithm.

A. SIR MODEL OF EOSA

SEIR-based models designed for EVD have been proposed in the literature to monitor both direct and indirect propagation of the disease in the affected population [45] and [5]. This study adopts and adapts two relevant models from

the existing SEIR models by identifying and adding new compartments perceived as omitted. These compartments are the contaminated environment serving as a reservoir of the virus, vaccination and quarantine, denoted by PE, V, and Q, respectively. This became necessary considering that the Ebola virus and disease are not propagated among the human population except by an individual infected from the reservoir. Also, the roles played by vaccination and quarantined infected individuals have impacted on the propagation rate of the virus. This perception is supported by recent studies [68]–[71]. This therefore necessitated the re-modeling of the propagation model which now yielded the SEIR-HDFVQ: Susceptible (S), Exposed (E), Infected (I), Recovered (R), Hospitalized (H), Death or dead (D), Funeral (F), Vaccinated (V), and Quarantine (Q). Also, in designing the model, we considered that an insignificant number of recovered cases might still retain the virus in their body fluid, which has potency for infecting healthy individuals [72]–[74]. Since this study's interest was to leverage the propagation model of the EVD for developing an optimization algorithm, it became necessary to explore all factors supporting increased infection.

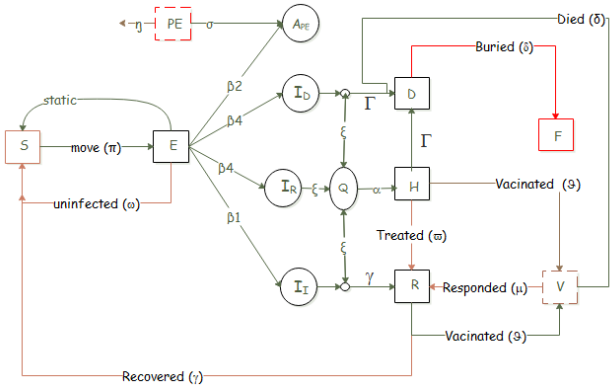


FIGURE 1. The SEIR-HDVQ propagation model of the proposed EOSA metaheuristic model.

The model of the SEIR-HDVQ is shown in Figure 1, and the listing of its parameters is presented in Table 1. The propagation of EVD is assumed to provide a suitable manner for solving some optimization problems considering its aggressive infection rate is overwhelming communities. The Figure assumes a population of susceptible individuals whose exposure could trigger the population of other subgroups. Exposed individuals, contaminated environments, and agent reservoirs can randomly draw arbitrary individuals from the susceptible into the category of infected, which may be due to exposure to any individual from the subgroups of the infected. Subgroups of infected individuals are infected from the dead individual, infected individual, recovered individual, contaminated environment, and agent-reservoir. We show that the virus has the potential of decaying in its contaminated environment. Furthermore, the propagation model shows that the infected cases could die without going to a hospital and recover without hospitalization. An assumption made in this study was classifying every vaccinated case as hospitalized. Also, we assumed that both the hospitalized (H) and non-hospitalized cases could transit into the dead (D). At the same time, those recovered (R) from vaccination (V) are returned to the susceptible (S).

TABLE 1. Notations and description for variables and parameters for SEIR-HDVQ.

Symbols	Data Type	Descriptions
S	Vector	Susceptible individuals
E		Exposed individuals
I		Infected individuals
H		Hospitalized infected individuals
R		Recovered infected individuals
D		Diseased from infection individuals
V		Vaccinated infected individuals
PE		Agents capable of infecting individuals
π	Scalar	Recruitment rate of susceptible human individuals
η		Decay rate of Ebola virus in the environment
α		Rate of hospitalization of infected individuals
Γ		Disease-induced death rate of human individuals
β_1		Contact rate of infectious human individuals
β_2		Contact rate of pathogen individuals/environment
β_3		Contact rate of deceased human individuals
β_4		Contact rate of recovered human individuals
γ		Recovery rate of human individuals
τ		Natural death rate of human individuals
δ		Rate of burial of deceased human individuals
ϑ		Rate of vaccination of individuals
ϖ		Rate of response to hospital treatment
μ		Rate response to vaccination
ξ		Rate of quarantine of infected individuals

The rates of change of variables or parameters applied in this study are summarized in Table 1. The values of most of these parameters are already predetermined by related studies on EVD and are detailed in Section 4.

B. FLOWCHART OF EOSA

1. Motivated by the performance of the SEIR-HDVQ model, we derived the design of the EOSA algorithm. The formalization of the EOSA algorithm is achieved from the following procedure: Initialize all vector and scalar quantities which are individuals and parameters: Susceptible (S), Infected (I), Recovered (R), Dead (D), Vaccinated (V), Hospitalized (H), and Quarantine (Q).
2. Randomly generate the index case (I_1) from susceptible individuals.
3. Set the index case as the global best and current best, and compute the fitness value of the index case.
4. While the number of iterations is not exhausted and there exists at least an infected individual, then
   a. Each susceptible individual generates and updates their position based on their displacement. Note that the further an infected case is displaced, the more the number of infections, so that short displacement describes exploitation, otherwise exploration.
      i. Generate newly infected individuals (nI) based on (a).
      ii. Add the newly generated cases to I.

b. Compute the number of individuals to be added to H, D, R, B, V, and Q using their respective rates based on the size of I.
c. Update S and I based on nI.
d. Select the current best from I and compare it with the global best.
e. If the condition for termination is not satisfied, go back to step 6.

5. Return global best solution and all solutions.

In Figure 2 below, the flow chart of the proposed EOSA metaheuristic algorithm is shown.

The flowchart presents the flow of process and information as a buildup from the procedure described above. The detailing shows the various levels of initialization and conditional checking. Also, the computation leading to the exploration and exploitation stages of the proposed EOSA metaheuristic algorithm are demonstrated. Lastly, the procedure for updating all subgroups is identified. In the following subsection, the algorithm's mathematical model, as it applies to the flowchart, is presented and discussed.

C. MATHEMATICAL MODEL OF EOSA

To update the positions of each exposed individual, Equation (1) applies:

$$mI_i^{t+1} = mI_i^t + \rho M\,(I) \tag{1}$$

where ρ represents the scale factor of displacement of an individual, mI_i^{t+1} and mI_i^t are the updated and original positions respectively at time t and $t+1$. $M(I)$ is the movement rate made by individuals and is defined thus:

$$M\,(I) = srate * rand\,(0,1) + M(Ind_{best}) \tag{2}$$

$$M\,(s) = lrate * rand\,(0,1) + M(Ind_{best}) \tag{3}$$

The exploitation stage is designed based on the assumption that the infected individual either stays within a distance of zero (0), or is displaced within a limit not exceeding *srate* - where *srate* denotes short distance movement. The exploration phase is founded on the fact that the infected individual moves beyond the average neighborhood range *lrate*. The consideration in this study is that the farther the displacement, the more the number of individuals in S are exposed to infection. Both cases are shown in Equations (2) and (3). The *srate* and *lrate* are regulated by a *neighborhood* parameter such that when *neighborhood* is >= 0.5, an individual has moved beyond the *neighborhood* leading to the mega infection; otherwise it remains within the *neighborhood,* which curbs infection.

1) INITIALIZATION OF SUSCEPTIBLE POPULATION

At the beginning, an initial population is generated by random number distribution whose initial positions are all zero (0). The individual is generated as shown in Equation (4). The $\boldsymbol{U_i}$ and $\boldsymbol{L_i}$ denote the upper and lower bounds respectively for the $\boldsymbol{i^{th}}$ individual, where I ranges from 1,2,3... N, in the population size.

$$individual_i = L_i + rand\,(0,1) * (U_i + L_i) \tag{4}$$

The selection of the current best is computed on the set of infected individuals in time t as seen in Equation (5):

$$bestS = \begin{cases} gBest, & fitness(cBest) < fitness\,(gBest) \\ cBest, & fitness\,(cBest) \geq fitness\,(gBest) \end{cases} \tag{5}$$

where ***bestS***, ***gBest*** and ***cBest*** all denote the best solution, global best solution, and current best solution at time ***t***; ***fitness*** represents the objective function applied to the problem. We distinguish ***gBest*** and ***cBest*** as infected individuals who are ***Superspreader*** and ***Spreader*** of the Ebola virus, respectively.

Update of Susceptible (S), Infected (I), Hospitalized (H), Exposed (E), Vaccinated (V), Recovered (R), Funeral (F), Quarantine (Q), and a system of ordinary differential equations govern dead (D) based on those in [45] and [5]. Differential calculus is a branch of calculus that is a branch in mathematics. The former deals with the rate of change of one quantity concerning another, while the latter deals with finding different properties of integrals and derivatives. In our case, the application of differential calculus intends to obtain the rates of change of quantities S, I, H, R, V, D, and Q with respect to time ***t***. Hence, the Equations (6)-(12) are as follows:

$$\frac{\partial S(t)}{\partial t} = \pi - (\beta_1 \mathrm{I} + \beta_3 \mathrm{D} + \beta_4 \mathrm{R} + \beta_2\,(\mathrm{PE})\,)\,S - (\tau \mathrm{S} + \Gamma \mathrm{I}) \tag{6}$$

$$\frac{\partial I(t)}{\partial t} = (\beta_1 \mathrm{I} + \beta_3 \mathrm{D} + \beta_4 \mathrm{R} + \beta_2\,(\mathrm{PE})\,\lambda)\,S - (\Gamma + \gamma)\,I - (\tau)\mathrm{S} \tag{7}$$

$$\frac{\partial H(t)}{\partial t} = \alpha \mathrm{I} - (\gamma + \varpi)\mathrm{H} \tag{8}$$

$$\frac{\partial R(t)}{\partial t} = \gamma I - \Gamma R \tag{9}$$

$$\frac{\partial V(t)}{\partial t} = \gamma I - (\mu + \vartheta)V \tag{10}$$

$$\frac{\partial D(t)}{\partial t} = (\tau \mathrm{S} + \Gamma \mathrm{I}) - \delta D \tag{11}$$

$$\frac{\partial Q(t)}{\partial t} = (\pi \mathrm{I} - (\gamma \mathrm{R} + \Gamma \mathrm{D})) - \xi Q \tag{12}$$

We assume that Equations (6-11) are *scalar functions*, meaning that each has one number as a value, which can be represented as a float. This is not far removed from some common scalar differential equations and their corresponding f functions, such as exponential growth of money or populations governed by scalar differential equations: $u' = \alpha u$, where u *is the* growth rate.

We determine the rate of change of the population of susceptible individuals and then apply it to the current size of the susceptible vector to obtain the number of susceptible individuals at time ***t***. The same procedure is applied to compute the set of individuals in vectors I, H, R, V, D, and Q using rates described in Table 1. This study assumes the

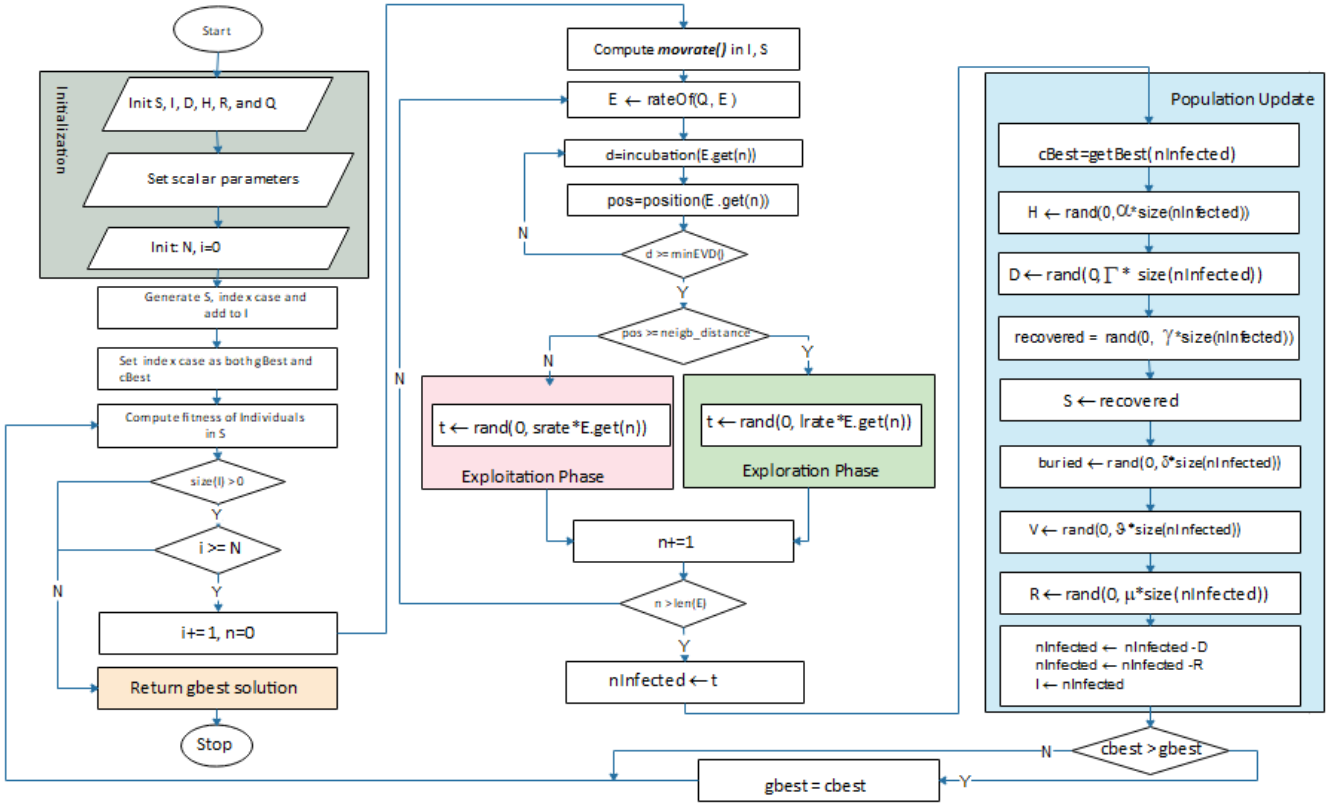


FIGURE 2. Flowchart of the proposed EOSA metaheuristic algorithm.

initial conditions S(0) = S0, I(0) = I0, R(0) = R0, D(0) = D0, P(0) = P0, and Q(0) = Q0 where our t follows after the epoch, and δ in equation (11) is for the burial rate. Equation (12) models the rate of quarantine of infected cases of Ebola.

D. ALGORITHM DESIGN OF EOSA

The pseudo-code of the proposed EOSA metaheuristic algorithm is shown in Algorithm 1. Lines 1-7 of the algorithm show the initialization phase. To naturalize the concept that not all infected cases have the potency for recruiting newly infected individuals, on Line 8 we show that some I are drawn into quarantine status so that the remaining fraction of I infect S population. On Lines 10-24, new infections are generated from S and then added to I. Since R, V, H, and V are only derivable from I, Lines 25-29 of Algorithm 1 generate individuals using corresponding equations of subgroups. Logically, recovered and dead cases need to be removed from I before the next iteration. In our demonstration, recovered cases are added back to S while dead individuals are replaced in S with new cases – to promote the idea of new births as shown on Lines 29-31. Finally, the best solution is computed, and the termination criterion is checked so that when satisfied, the algorithm terminates, otherwise return to Line 7. To demonstrate the usability of the algorithm, we follow on in the next section for experimental setup, configurations, and parameter definition.

IV. EXPERIMENTAL SETUP

This section presents the computational environment applied for experimentation. First, we show the control parameter settings and variable assignment, then a listing of the benchmark functions applied to the algorithm, and finally detail the evaluation criteria.

A. CONFIGURATION OF THE EXPERIMENTAL SETUP

Exhaustive experimentation evaluated the proposed EOSA in a workstation environment with the following configurations: Intel (R) Core i5-7500 CPU 3.40GHz, 3.41GHz; RAM of 16 GB 64-bit Windows 10 OS for each configuration of the system on the network. A total of ten (10) existing metaheuristic algorithms were implemented and experimented with for comparative purposes with the EOSA algorithm. This study executed each algorithm twenty (20) times to ensure fairness in each algorithm's evaluation. Also, five hundred (500) epochs were covered in each run. The runs of 20 for each algorithm allowed computing the average values for all metrics.

B. PARAMETERS OF EOSA METAHEURISTIC ALGORITHM

The design and selection of EOSA's parameters and corresponding values assumed the natural definitions generated from those reported in the literature. In this study, we adopted the rates reported in studies that have extensively evaluated

Algorithm 1 EOSA metaheuristic algorithm

Result: Best solution
Input: objfunc, lb, ub, epoch, psize, evdincub
Output: solution, gbest

$S, E, I, H, R, V, Q, sols \leftarrow \emptyset$;
$S \leftarrow createSusceptibleIndvd(psize, S)$ *using* $Eq.4$;
$icase \leftarrow generatedIndexCase(S)$;
$gbest, cbest \leftarrow icase$;
while $e \leq epoch \wedge len(I) > 0$ **do**
  $Q \leftarrow rand(0, Eq.12 \times I)$;
  $fracI = I - Q$;
  for $i \leftarrow 1$ ***to*** $len(fracI)$ **do**
    $pos_i \leftarrow movrate()$ *using* $Eq.1$;
    $d_i \leftarrow rand()$;
    $newI \leftarrow \emptyset$;
    if $d_i > evdincub$ **then**
      $neighborhood \leftarrow prob(pos_i)$;
      if $neighborhood < 0.5$ **then**
        $tmp \leftarrow rand(0, Eq.7 \times I \times srate)$;
      end
      else
        $tmp \leftarrow rand(0, Eq.7 \times I \times lrate)$;
      end
      $newI+ \leftarrow tmp$;
    end
    $I+ \leftarrow newI$;
  end
  $h \leftarrow rand(0, Eq.8 \times I), H+ \leftarrow h$;
  $r \leftarrow rand(0, Eq.9 \times I), R+ \leftarrow r$;
  $v \leftarrow rand(0, Eq.10 \times h), V+ \leftarrow v$;
  $d \leftarrow rand(0, Eq.11 \times I), D+ \leftarrow d$;
  $I+ \leftarrow I - add(r, d)$;
  $S+ \leftarrow r$;
  $S- \leftarrow d$;
  $cbest = fitness(objfunc, I)$;
  if $cbest > gbest$ **then**
    $gbest = cbest$;
    $sols \leftarrow gbest$;
  end
end
return gbest, sols;

the SEIR models. These studies relied on the WHO data for the evaluation of their models. All these parameters have been described in Section 3, where the SEIR-HDVQ model was presented.

In Table 2, the initial value for each parameter is defined. Considering the stochastic nature of EOSA, which is characteristic of biology-based optimization algorithms, values for some parameters are randomly assigned. The problem size applied for all experimentation is fixed at one hundred (100). We note that these values remain fixed for all experiments on the benchmark functions.

TABLE 2. Notations and description of variables and parameters for SEIR-HDVQ.

Symbols	Descriptions	Range
π	Recruitment rate of susceptible human individuals	0.1
β_1	Contact rate of infectious human individuals	0.1
β_2	Contact rate of pathogen individuals/environment	0.1
β_3	Contact rate of deceased human individuals	0.1
β_4	Contact rate of recovered human individuals	0.1
Γ	Disease-induced death rate of human individuals	Random values initialized within the range of 0-1
γ	Recovery rate of human individuals	
η	Decay rate of Ebola virus in the environment	
α	Rate of hospitalization of infected individuals	
τ	Natural death rate of human individuals	
δ	Rate of burial of deceased human individuals	
ϑ	Rate of vaccination of individuals	
ϖ	Rate of response to hospital treatment	
μ	Rate response to vaccination	
ξ	Rate of quarantine of infected individuals	

C. BENCHMARK FUNCTIONS

To evaluate the effectiveness of the proposed EOSA metaheuristic algorithm, this study applied forty-seven (47) standard and high dimensional functions for this purpose. These functions are listed in Table 3 and are subsequently used for performance comparison with similar metaheuristic algorithms. We listed the names, mathematical representation, and range of the functions. We also evaluated the algorithm using the IEEE-CEC benchmark functions to demonstrate exhaustive experimentation.

Whereas many test functions are continuous, they are categorized into four (4). Test functions characterized by unimodal, convex, and multidimensional forms are first class. They represent a class of test functions with interesting functions with cases capable of slowing down convergence or even yielding a poor convergence. The resulting convergence trails from such a slow pace to a single global extremum. The second class consists of test functions of type multimodal, two-dimensional with few local extremes. This test function category appeals to situations where we intend to test the quality of standard optimization procedures in an anticipated hostile environment. This hostile environment describes problem domains with only a few local extremes with a single global one. The third and fourth classes represent a list of test functions known as multimodal two-dimensional with a huge number of local extremes, and multimodal multidimensional, with a huge number of local extremes. It has been shown that these test functions work well for situations where the quality of intelligent and resistant optimization algorithms are tested [41], [75]–[79].

D. EVALUATION METHOD

The following metrics were considered in the performance evaluation: mean, median, standard deviation, maximum values, minimum or worst values, average values, overall convergence time, and average execution time. In addition to

TABLE 3. Standard benchmark functions used for the experimentation: Dimensions (D), Multimodal (M), Non-separable (N), Unimodal (U), Separable (S).

ID	Function name	Range	Model of the function	D	Type	Min
F1	Ackley	[−32, 32]	$f(x) = -20e^{\left(-0.2\sqrt{\frac{1}{n}\sum_1^n x_i^2}\right)} - -e^{\left(\frac{1}{n}\sum_1^n \cos(2\pi x_i)\right)} + 20 + e^{(1)}$	30	MN	0
F2	Alpine	[−10, 10]	$f(x) = \sum_{i=1}^{n} \lvert x_i \sin(x_i) + 0.1x_i \rvert$	N	MN	0
F3	Brown	[-1, 4]	$f(x) = \sum_{i=1}^{n-1} (x_i^2)^{(x_{i+1}^2+1)} + (x_{i+1}^2)^{(x_i^2+1)}$	N	UN	0
F4	Bent Cigar	[-100,100]	$f_{20}(x) = x_1^2 + 10^6 \sum_{i=2}^{D} x_i^2$	N	MS	0
F5	Composition1	[-100,100]	g1=Rosenbrock's Function F29 g2=High Conditioned Elliptic Function F15 g3=Rastrigin's Function F27	5		
F6	Composition2	[-100,100]	g1=Ackley's Function F1 g2=High Conditioned Elliptic Function F15 g3=Griewank Function F10 g4=Rastrigin's Function F27	3		
F7	Dixon and Price	[-10, 10]	$f_{18}(x) = 10^6 x_1^2 \sum_{i=2}^{D} x_i^2$	30	UN	0
F8	Discus Function	[−100, 100]	$f(x) = (x_1 - 1)^2 + \sum_{i=2}^{n} i(2x_i^2 - x_{i-1})^2$	N	U	
F9	Fletcher–Powel	[−100, 100]	$f(x) = 100\left\{\left[x_3 - 10\theta(x_1,x_2)\right]^2 + \left(\sqrt{x_1^2 + x_2^2} - 1\right)^2\right\} + x_3^2$ Where $2\pi\theta(x_1,x_2) = \begin{cases} \tan^{-1}\frac{x_2}{x_1}, & if x_1 \geq 0 \\ \pi - \tan^{-1}\frac{x_2}{x_1}, & otherwise \end{cases}$	N	MN	0.0001
F10	Griewank	[−600, 600]	$f(x) = 1 + \sum_{i=1}^{n} \frac{x_i^2}{1400} - \prod_{i=1}^{n} \cos(\frac{x_i}{\sqrt{i}})$	30	MN	0
F11	Generalized Penalized Function 1	[-50, 50]	$f(x) = \frac{\pi}{n} X \left\{10sin^2(\pi y_i) + \sum_{i=1}^{n-1}(y_i - 1)^2 [1 + 10sin^2(\pi y_{i+1})] + (y_n - 1)^2\right\} + \sum_{i=1}^{n} u(x_i, a, k, m)$ Where $y_i = 1 + \frac{1}{4}(x_i + 1)$, $u(x_i,a,k,m) = \begin{cases} k(x_i - a)^m & if\ x_i > a \\ 0 & if -a \leq x_i \leq a \\ k(-x_i - a)^m & if\ x_{i<-a} \end{cases}$ a=10, k=100, m=4	n	M	0
F12	Generalized Penalized Function 2	[-5.12, 5.12]	$f(x) = 0.1\ X \left\{sin^2(3\pi x_1) + \sum_{i=1}^{n-1}(x_i - 1)^2 [1 + sin^2(3\pi x_{i+1})] + (x_n - 1)^2[1 + sin^2(2\pi x_n)]\right\} + \sum_{i=1}^{n} u(x_i, a, k, m)$ Where $u(x_i,a,k,m) = \begin{cases} k(x_i - a)^m & if\ x_i > a \\ 0 & if -a \leq x_i \leq a \\ k(-x_i - a)^m & if\ x_{i<-a} \end{cases}$ a=5, k=100, m=4	N	M	0
F13	Holzman 2 function	[-100,100]	$f(x) = \sum_{i=1}^{n} ix_i^4$	N		
F14	HGBat	[-100,100]	$f_{23}(x) = \left\lvert (\sum_{i=1}^{D} x_i^2)^2 - (\sum_{i=1}^{D} x_i)^2 \right\rvert^{1/2} + (0.5\ \sum_{i=1}^{D} x_i^2 + \sum_{i=1}^{D} x_i)/_D + 0.5$	30	M	
F15	High Conditioned Elliptic	[-100,100]	$f_{23}(x) = \sum_{i=1}^{D} (10^6)^{\frac{i-1}{D-1}} x_i^2$	N		
F16	Hybrid1	[-100,100]	g1 : Zakharov Function F45 g2 : Rosenbrock Function F29 g3: Rastrigin's Function F27	3	UN	0
F17	Hybrid2	[-100,100]	g1 : High Conditioned Elliptic Function F15 g2 : Ackley's Function F1 g3: Rastrigin's Function F27 g4: HGBat Function F14 g4: Discus Function F8	3	MN	0
F18	Inverted Cosine Mixture	[-1,1]	$f_{14}(x) = 0.1n - (0.1\ \sum_{i=1}^{n} \cos(5\pi x_i) - \sum_{i=1}^{n} x_i^2)$	N	MS	-0.1x(n)
F19	Lévy 3 function	[−10, 10]	$f(x) = \sum_{i=1}^{n-1}\left[0.5 + \frac{sin^2(\sqrt{100x_i^2 + x_{i+1}^2}) - 0.5}{1 + 0.001(x_i^2 - 2x_ix_{i+1} + x_{i+1}^2)^2}\right]$	N		
F20	Levy	[−10, 10]	$f_{12}(x) = \sum_{i=1}^{n}(x_i - 1)^2 [sin^2(3\pi x_{i+1})] + sin^2(3\pi x_1) + \lvert x_n - 1 \rvert [1 + sin^2(3\pi x_n)]$	2	MN	0
F21	Levy and Montalo	[−5, 5]	$f_{17}(x) = 0.1(sin^2(3\pi x_1)) + \sum_{i=1}^{n}(x_i - 1)^2(1 + sin^2(3\pi x_{1+1})) + (x_n - 1)^2(1 + sin^2(2\pi x_n))$	N	MS	0
F22	Noise	[−1.28, 1.28]	$f_7(x) = \sum_{i=1}^{n} x_i^4 + random[0,1)$	N		
F23	Pathological function	[-100,100]	$f(x) = \sum_{i-1}^{5} icos((i-1)x_1 + i) \sum_{j=1}^{5} jcos((j+1)x_1 + j)$	N	MN	0
F24	Perm	[-20, 20]	$f(x) = \sum_{k=1}^{n}\left[\sum_{i=1}^{n}(i_k + \beta)((\frac{x_i}{i})^k - 1)\right]^2$	N	MN	0
F25	Powel	[-4, 5]	$f(x) = (x_1 + 10x_2)^2 + 5(x_3 + x_4)^2 + (x_2 - 2x_3)^4 + 10(x_1 - x_4)^4$	N	UN	0
F26	Quartic	[-128, 128]	$f_6(x) = \sum_{i=1}^{n} ix_i^4$	30	MS	0
F27	Rastrigin	[−5.12, 5.12]	$f_9(x) = \sum_{i=1}^{n}[x_i^2 - 10\cos(2\pi x_i) + 10]$	30	MN	0

TABLE 3. *(Continued.)* **Standard benchmark functions used for the experimentation: Dimensions (D), Multimodal (M), Non-separable (N), Unimodal (U), Separable (S).**

F28	Rotated hyperellipsoid	[−100, 100]	$f_3(x) = \sum_{i=1}^{n}\left(\sum_{j=1}^{i} x_j\right)$	N	U	0
F29	Rosenbrock	[−30, 30]	$f(x) = \sum_{i=1}^{n-1}[100\,(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	30	UN	0
F30	Schwefel 2.26	[-500, 500]	$f(x) = \sum_{i=1}^{n}[-x_i \sin(\sqrt{\lvert x_i \rvert})]$	N	MS	−418.983
F31	Schwefel 1.2	[−100, 100]	$f(x) = \sum_{i=1}^{n}(\sum_{j=1}^{i} x_j)^2$	30	UN	0
F32	Schwefel 2.22	[−100, 100]	$f(x) = \sum_{i=1}^{n}\lvert x_i \rvert + \prod_{i=1}^{n}\lvert x_i \rvert$	30	UN	0
F33	Schwefel 2.21	[−100, 100]	$f(x) = \max\{\lvert x_i \rvert, 1 \le i \le n\}$	N	US	0
F34	Sphere	[−100, 100]	$f_1(x) = \sum_{i=1}^{n} x_i^2$	30	US	0
F35	Step	[−100, 100]	$f(x) = \sum_{i=1}^{n}(floor(x_i) + 0.5)^2$	30	US	0
F36	Sum/SumSquares Function	[−10, 10]	$f(x) = \sum_{i=1}^{n} i x_i^2$	30	US	0
F37	Sum-Power	[−1, 1]	$f_8(x) = \sum_{i=1}^{n}\lvert x_i \rvert^2$	N	US	0
F38	Sum of Different Power	[-100,100]	$f_{21}(x) = \sum_{i=1}^{d}\lvert x_i \rvert^{i+1}$	N	US	0
F39	SR-F4	[-100,100]	Shifted and Rotated Bent Cigar Function	N		
F40	SR-F38	[-100,100]	Shifted and Rotated Sum of Different Power Function	N		
F41	SR-F45	[-100,100]	Shifted and Rotated Zakharov Function	N		
F42	SR-F29	[-100,100]	Shifted and Rotated Rosenbrock's Function	N	MN	0
F43	SR-F27	[-100,100]	Shifted and Rotated Rastrigin's Function	N	MS	0
F44	Wavy 1	[-100,100]	$f(x) = \sum_{i=1}^{n} x_i^2 + (\sum_{i=1}^{n} 0.5 i x_i)^2 + (\sum_{i=1}^{n} 0.5 i x_i)^4$	2	MS	0
F45	Zakharov	[−5, 10]	$f(x) = \frac{1}{n}\sum_{i=1}^{n} 1 - \cos(10x_i)\, e^{-\frac{1}{2}x_i^2}$	10	UN	0
F46	Salomon	[−100, 100]	$f_{19}(x) = 1 - cos\left(2\pi \sqrt{\sum_{i=1}^{n} x_i^2}\right) + 0.1\sqrt{\sum_{i=1}^{n} x_i^2}$	N	MN	0
F47	Weierstrass Function	[-0.5, 0.5]	$f(x) = \sum_{i=1}^{D}(\sum_{i=0}^{20}[0.5^k \cos(2\pi . 3^k (x_i + 0.5))])$	50	MN	0

these, we applied the outcome of the proposed EOSA and related optimization algorithms to statistical tests to evaluate their performance in terms of convergence to determine algorithms capable of generating similar final solutions.

V. RESULT AND DISCUSSION

A detailed performance evaluation of the outcome of the experimentation carried out in Section 4 is presented and discussed in this Section. First, we study the performance of the proposed SEIR-HDVQ model to determine how it effectively describes the natural phenomenon. Performance evaluation uses the values obtained from applying the optimization algorithms to the test functions. Compared with other methods, the proposed algorithm's performance is statistically analysed. Also, the application of the algorithm to medical image classification using convolutional neural network (CNN) architectures is presented.

A. SIMULATION OF EVD PROPAGATION BASED ON SEIR-HDVQ MODEL

The simulation of the proposed SEIR-HDVQ model applied to EOSA during experimentation is demonstrated. The result

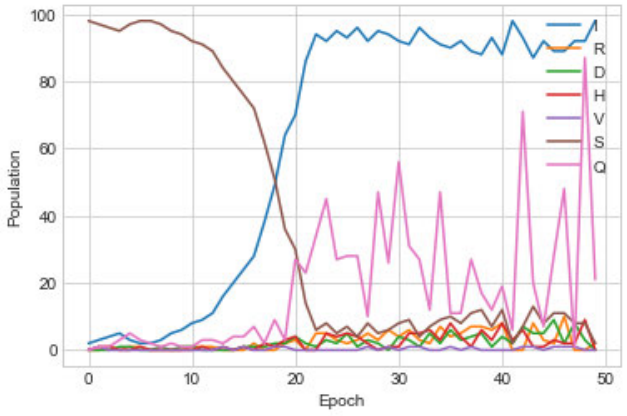


FIGURE 3. An estimated propagation curve of the Ebola virus and disease based on the simulation with randomly generated data while experimenting with the EOSA optimization algorithm. The curve illustrates variations in the values of Susceptible (S), Infected (I), Recovered (R), Hospitalized (H), Dead (D), Vaccinated (V), and Quarantine (Q) using the SEIR-HDVQ model.

is reported to investigate the naturalization tendency of the SEIR-HDVQ model as obtained in the real-life propagation model for the EVD and EBOV.

In Figure 3, the curves for Susceptible (S), Infected (I), Recovered (R), Hospitalized (H), Dead (D), Vaccinated (V), and Quarantine (Q) are captured so that they show the rate at which each compartment rises and falls within a period of fifty (50) epochs. In the Figure, we observed that the early phase of the infection outbreak rose against the susceptible population. The Figure also revealed the response to the rising infection rate through quarantine measures, such that as infection rose, the number of quarantine individuals also increased – a measure to stem the outbreak. Meanwhile, we noticed that recovery and death rate curves wobbled along with infection and quarantine.

B. PERFORMANCE OF EOSA WITH SIMILAR METAHEURISTIC ALGORITHMS USING CLASSICAL FUNCTIONS

The performance of EOSA was compared with nine (9) different optimization algorithms, namely Artificial Bee Colony (ABC), Whale Optimization Algorithm (WOA), Butterfly Optimization Algorithm (BOA), Particle Swarm Optimization (PSO), Differential Evolution (DE), Genetic Algorithm (GA), Henry Gas Solubility Optimization Algorithm (HGSO), Blue Monkey Optimization (BMO), and Sandpiper Optimization Algorithm (SOA). The experimentation, which was executed for five hundred (500) iterations and twenty (20) different runs, applied forty-seven (47) standard benchmark functions.

Table 4 lists the outcome for the best, worst, mean, median, and standard deviation for each of the 47 functions. An overview of the results showed that although EOSA outperformed most of the algorithms in most cases for the 47 functions, some interesting differences were noticed, which appears to group the outcome into two. Whereas EOSA demonstrated a very close performance compared to ABC, WOA, BOA and PSO, we observed that EOSA's outcome compared with DE, GA, and HGSO was significantly better. For example, for F1-4, F6-7, F12, F14-15, F18, F20-23, F2, F29-30, F32-36, F38, F40-43, and F46-47, EOSA clearly achieved the best values in all cases. However, in the cases of F5, F9-11, F13, F16-17, F19, and F25-26, EOSA was outperformed based on WOA and BOA, ABC, WOA, and PSO, ABC, BOA and PSO, ABC and PSO, ABC, ABC, WOA, and BOA, respectively. Meanwhile, we found some situations for the 47 functions where there was no clear superiority of EOSA over similar algorithms, neither were the similar algorithms able to demonstrate clear superiority. These cases are found in F28, F31, F37, F39, F44, and F45, where we observed that ABC and PSO beat EOSA, beaten by WOA, BOA and PSO, ABC, matched values in ABC and PSO, beaten by WOA, and beaten by ABC, WOA, and PSO respectively

As shown in Table 4 for the 47 functions, the values obtained for the worst revealed a strong competition between EOSA and ABC, WOA, BOA, and PSO. We discovered that only in the cases of F3, F6, F12, F18, F22-23, F27, F29, F28, and F40-41 were the values of EOSA better than those listed earlier and also completely outperformed those same algorithms in the cases of F13, F21, F25, and F32. However, we noticed that the discrepancies reported by these algorithms compared to EOSA were not significantly large. Meanwhile, DE, GA, and HGSO maintained a significant variation in the values obtained for best and worst computations. This implied that the proposed EOSA algorithm and its competitive related algorithms (ABC, WOA, BOA, and PSO) performed significantly well over DE and GA. Meanwhile, we found the performance of SOA and HGSO very competitive with that of EOSA. For instance, when investigating the overall superiority of each algorithm compared with others, the following were observed: EOSA(12), ABC(4), WOA(1), BOA(1), PSO(1), DE(1), GA(2), BMO(0), HGSO(19), and SOA(21). This shows that both HGSO and SOA had more occurrences of superiority. Interestingly, when each method was compared with EOSA, the following were observed: ABC/EOSA(15/28), WOA/EOSA(5/28), BOA/EOSA(2/28), PSO/EOSA(4/28), DE/EOSA(0/28), GA/EOSA(0/28), BMO/EOSA(0/28), HGSO/EOSA(23/21), and SOA/EOSA(22/20).

We note that although other similar metaheuristics algorithms may compete with the EOSA method on the sets of the various mathematical benchmark functions tested in this study, it is noteworthy that the comparison with the selected algorithms showed that the performance of EOSA is very competitive.

To confirm the outstanding performance and superiority of the EOSA, the convergence curves of the history of solutions are graphed in Figure 4. The benchmark functions F1, F2, F3, F4, F7, F8, F20, F25, F26, F27, F43, and F45. We observed that in all cases, the curve of the plots descended appreciably. The implication is that the EOSA algorithm is a very competitive optimization algorithm that can discover optimum solutions in exploration and exploitation operations. To demonstrate the superiority of EOSA, when its convergence curve was plotted against those of state-of-the-art algorithms, we found that its solutions were significant.

The convergence of EOSA using the benchmark functions as compared with ABC, WOA, PSO, and GA was graphed and is illustrated in Figure 5. The Figures showed that in most cases, the best values for all the algorithms were descending from their initial peak values to a lower value as the training improved over some epochs. The graphing was achieved by obtaining the best values for each case of EOSA, ABC, WOA, PSO, and GA at 1, 50, 100, 200, 300, 400, 500 iteration points. The Figure confirms that the best values for GA are often larger than the others except in a few cases where ABC also has some large values. However, ABC, WOA, PSO, and the proposed EOSA do not just have low values for the best cases but appear to only drop in value for small fractions across all the iterations. This is why their rise and fall concerning their curves was not so pronounced compared to that of GA and sometimes ABC.

Perspective views of some selected functions, alongside the search history, are shown in Figures 6 and 7. Again,

TABLE 4. Comparison of best, worst, mean, median and standard deviation values for ABC, WOA, BOA PSO, EOSA, DE, GA, HGSO, SOA, and BMO metaheuristic algorithms using the classical benchmark functions over 500 runs and 100 population size.

		ABC	WOA	BOA	PSO	DE	GA	BMO	EOSA	HGSO	SOA
F1	best	**0.046635003**	0.046548228	0.046606998	0.046595289	19.96184399	9.834179892	19.52206321	0.046579079	**4.44E-16**	**4.44E-16**
	worst	20.90485862	0.046548228	0.046606998	0.046595289	20.91052673	19.84903108	20.95614049	0.046590768	17.47294741	19.99999996
	mean	19.35207569	0.046548228	0.046606998	0.046595289	20.0105497	10.33766475	19.70969609	0.046579273	0.224608623	12.97192291
	median	19.21059815	0.046548228	0.046606998	0.046595289	19.96347193	10.09803718	19.65349837	0.046579079	4.44E-16	19.99999996
	sd	0.943908994	4.86E-18	5.20E-18	5.20E-18	0.146978912	0.961561963	0.212554077	1.24E-06	1.441674873	9.521168159
F2	best	0.002749312	0.002795749	0.002762226	0.002778424	202.2092691	39.09706932	1397.614847	**0.002735207**	**1.15E-59**	**0**
	worst	243.8707278	0.002795749	0.002762226	0.002778424	243.8368578	181.9837599	2048.825326	0.002786444	151.3617491	2602.467649
	mean	33.43141849	0.002795749	0.002762226	0.002778424	223.0877822	43.99604306	1407.489662	0.002735311	1.067724803	16.73286102
	median	7.44293501	0.002795749	0.002762226	0.002778424	219.8379837	41.29894943	1397.761667	0.002735207	1.70E-31	0
	sd	52.1262757	4.12E-19	3.90E-19	4.34E-19	13.7309761	10.72063849	52.38240712	2.29E-06	10.27440175	166.9166469
F3	best	0.000414246	0.000406423	0.00041358	0.00040821	506.4395978	923.2199937	346510.1811	**0.000341339**	200	200
	worst	1459.126847	0.000406423	0.00041358	0.00040821	1511.839293	1242.053574	657747.7077	0.000402738	937.124982	1073805.498
	mean	290.8930668	0.000406423	0.00041358	0.00040821	766.5204487	940.9341817	349618.6135	0.000341484	203.9456023	38278.09761
	median	202.646981	0.000406423	0.00041358	0.00040821	689.7884683	931.4994158	346510.1811	0.000341339	200	200
	sd	219.6111673	3.79E-20	6.23E-20	7.32E-20	242.3395941	28.42301313	21747.40609	2.87E-06	44.96951031	145726.5681
F4	best	2.47E-12	2.48E-12	2.50E-12	**2.44E-12**	1.46E+11	4106464761	1.04867E+11	**2.44E-12**	**8.23E-104**	**0**
	worst	2.60168E+11	2.48E-12	2.50E-12	2.44E-12	2.58E+11	1.35E+11	1.26382E+11	2.44E-12	1.07E+11	2.55822E+11
	mean	2.04696E+11	2.48E-12	2.50E-12	2.44E-12	2.06E+11	5647478875	1.04953E+11	2.44E-12	489880801.7	9790935384
	median	2.00955E+11	2.48E-12	2.50E-12	2.44E-12	2.09E+11	4340687923	1.04867E+11	2.44E-12	1.73E-49	0
	Sd	13029284872	4.24E-28	3.23E-28	3.43E-28	39224298697	7415345765	1357996889	4.27E-17	5813315095	37694547763
F5	best	1.26E-06	**1.23E-06**	**1.23E-06**	1.24E-06	293464.1073	10387.08092	209785.6196	1.24E-06	**0**	**0**
	worst	513842.7078	1.23E-06	1.23E-06	1.24E-06	510854.4233	271266.3555	260316.6212	1.24E-06	198959.4441	541810.3844
	mean	408834.0144	1.23E-06	1.23E-06	1.24E-06	389379.6679	13319.70525	210095.2125	1.24E-06	1007.87092	4497.718731
	median	401536.9082	1.23E-06	1.23E-06	1.24E-06	370976.4288	10947.76684	209785.6196	1.24E-06	0	0
	sd	25604.25205	1.16E-22	1.59E-22	1.16E-22	70462.61054	14783.10391	3904.480646	9.72E-11	11466.50545	48286.12828
F6	best	2.40E-18	2.44E-18	**2.37E-18**	2.44E-18	1.49E+17	53323511.38	209785.6196	**2.37E-18**	44.10261447	44.33953792
	worst	1.45E+17	2.44E-18	2.37E-18	2.44E-18	1.49E+17	2.52E+16	209785.6196	2.40E-18	1.04E+16	1.70E+17
	mean	6.40E+16	2.44E-18	2.37E-18	2.44E-18	1.49E+17	5.83E+13	209785.6196	2.37E-18	3.01E+13	2.59E+16
	median	5.70E+16	2.44E-18	2.37E-18	2.44E-18	1.49E+17	1670980321	209785.6196	2.37E-18	44.12595826	44.33953792
	sd	1.59E+16	2.31E-34	2.50E-34	3.85E-34	19.2	1.14E+15	209785.6196	1.55E-21	5.15E+14	6.12E+16
F7	best	2.86E-12	2.82E-12	2.76E-12	2.86E-12	2.71E-146	865.7595089	260316.6212	**2.57E-12**	**1.02E-241**	**0**
	worst	100289061.5	2.82E-12	2.76E-12	2.86E-12	24647958.47	652837.5949	260316.6212	2.80E-12	54779492.97	5871510.079
	mean	256702.4063	2.82E-12	2.76E-12	2.86E-12	85506.69928	26635.78996	260316.6212	2.57E-12	114542.9249	11743.02016
	median	548.2299318	2.82E-12	2.76E-12	2.86E-12	3.40E-69	5236.830702	260316.6212	2.57E-12	1.47E-122	0
	sd	4643773.174	3.64E-28	2.83E-28	3.43E-28	1297580.095	84911.8986	260316.6212	1.06E-14	2464983.02	262319.2
F8	best	**1.01E-10**	1.02E-10	1.02E-10	**1.01E-10**	152250.3059	11996.3679	210095.2125	**1.01E-10**	**1.22E-109**	**0**
	worst	2219235.401	1.02E-10	1.02E-10	1.01E-10	1083569.996	242291.8529	210095.2125	1.02E-10	1472206.755	482941.467
	mean	260786.7052	1.02E-10	1.02E-10	1.01E-10	225703.8591	23799.47495	210095.2125	1.01E-10	5060.550052	27644.49411
	median	247109.3042	1.02E-10	1.02E-10	1.01E-10	202596.2197	14016.66679	210095.2125	1.01E-10	6.40E-57	0
	sd	113140.258	1.81E-26	1.23E-26	1.42E-26	99559.98645	29724.76944	210095.2125	3.63E-14	73338.7961	103305.6524
F9	best	1.08E-19	1.16E-19	1.13E-19	1.04E-19	**1.00E-24**	911.4244623	209785.6196	**9.62E-20**	0.15590465	10460.04926
	worst	18770598227	1.16E-19	1.13E-19	1.04E-19	92699584231	1419691418	209785.6196	9.90E-20	3.69E+11	3496139751
	mean	59781217.28	1.16E-19	1.13E-19	1.04E-19	238666361.4	5408441.124	209785.6196	9.62E-20	745438916.6	156886793.6
	median	1010.651298	1.16E-19	1.13E-19	1.04E-19	4.31E-07	1084.6816	209785.6196	9.62E-20	0.426743166	10460.04926
	sd	1012213432	1.44E-35	1.69E-35	1.87E-35	4423166355	78721555.39	209785.6196	1.25E-22	16495993498	1781951160
F10	best	**0.000268707**	0.000271867	0.000271857	0.000272074	1305.148346	37.97271859	3904.480646	0.000271605	**0**	**0**
	worst	2365.081755	0.000271867	0.000271857	0.000272074	2333.501515	1211.67998	3904.480646	0.000271631	758.1933658	62.89778761
	mean	1944.545227	0.000271867	0.000271857	0.000272074	1869.057115	52.82688221	3904.480646	0.000271605	3.586552218	2.147357315
	median	1927.15985	0.000271867	0.000271857	0.000272074	1869.71041	42.05742491	3904.480646	0.000271605	0	0
	sd	118.8407645	4.34E-20	3.52E-20	4.07E-20	346.0828863	66.89207141	3904.480646	1.29E-09	41.44826606	9.95495815
F11	best	**1.71E-10**	1.76E-10	1.72E-10	1.73E-10	2394966487	4.159217166	11210910151	**1.72E-10**	1.02650854	1.187374865
	worst	2594482104	1.76E-10	1.72E-10	1.73E-10	2567488472	678040718.4	33979497609	1.74E-10	429173991.9	78169609340
	mean	975017964	1.76E-10	1.72E-10	1.73E-10	2524020304	1702811.703	11346936651	1.72E-10	1632585.894	5291097513
	median	835513938.7	1.76E-10	1.72E-10	1.73E-10	2558711648	7.277752467	11210910151	1.72E-10	1.102714822	1.187374865
	sd	310620202.2	2.46E-26	1.94E-26	1.55E-26	63720410.11	30979999.46	1491667445	1.13E-13	22580371.19	18968405055
F12	best	0.00537744	0.005388969	0.005480457	0.005391917	59.10789733	6.976934078	14981554900	**0.003344178**	9.850801272	9.800000008
	worst	106.5810109	0.005388969	0.005480457	0.005391917	107.3461889	63.34963883	27264639920	0.004373574	43.96722626	1.03559E+11
	mean	11.41374051	0.005388969	0.005480457	0.005391917	81.28584828	8.490634765	15100740067	0.003346353	10.05723641	6446484084
	median	3.041798449	0.005388969	0.005480457	0.005391917	76.95222037	7.556693713	14981554900	0.003344178	9.888497941	9.800001513
	sd	19.1491529	6.94E-19	8.67E-19	6.51E-19	15.49249829	3.739496644	1118187202	4.69E-05	1.893231338	21793987519

TABLE 4. *(Continued.)* **Comparison of best, worst, mean, median and standard deviation values for ABC, WOA, BOA PSO, EOSA, DE, GA, HGSO, SOA, and BMO metaheuristic algorithms using the classical benchmark functions over 500 runs and 100 population size.**

F13	best	**3.65E-10**	3.74E-10	3.66E-10	3.73E-10	985634985.2	17956.26097	354906921.5	**3.74E-10**	4950	4950
	worst	1384897512	3.74E-10	3.66E-10	3.73E-10	1361818654	464120523.3	554301224	3.75E-10	340835146.7	1457114306
	mean	889078912.1	3.74E-10	3.66E-10	3.73E-10	1254140695	1772015.04	355869483	3.74E-10	1336939.589	24765556.99
	median	851795363.5	3.74E-10	3.66E-10	3.73E-10	1307126803	58264.23368	354906921.5	3.74E-10	4950	4950
	sd	94316100.35	3.88E-26	5.17E-26	5.95E-26	129934049.2	22235444.11	11397084.8	5.34E-14	17687769.58	165548118.7
F14	best	2.45E-06	2.45E-06	2.44E-06	2.42E-06	150745.5792	4173.394244	123695.6791	**2.43E-06**	0.5	0.5
	worst	263287.4203	2.45E-06	2.44E-06	2.42E-06	262848.6996	139559.2807	140863.5674	2.44E-06	75064.78913	264461.4534
	mean	205450.4186	2.45E-06	2.44E-06	2.42E-06	207583.4194	5816.112605	123809.6376	2.43E-06	366.1429711	7460.348419
	median	201684.2779	2.45E-06	2.44E-06	2.42E-06	207654.9189	4635.050797	123695.6791	2.43E-06	0.5	0.5
	sd	12935.14014	2.33E-22	3.39E-22	2.33E-22	39242.75828	7628.700089	1149.279977	1.64E-09	4110.579697	43620.22993
F15	best	2.77E-11	2.80E-11	2.74E-11	2.62E-11	1756599198	6640180.976	5249654907	**2.70E-11**	**1.37E-109**	**0**
	worst	10468218437	2.80E-11	2.74E-11	2.62E-11	11188908118	5342156920	5387808019	2.73E-11	5205941788	11951712694
	mean	5419343203	2.80E-11	2.74E-11	2.62E-11	3873865217	64605838.05	5250556989	2.70E-11	25839038.24	544547935
	median	5089210478	2.80E-11	2.74E-11	2.62E-11	3205180965	16145847.47	5249654907	2.70E-11	4.70E-55	0
	sd	849483013.5	5.49E-27	4.85E-27	4.20E-27	2157772540	295281070.4	10787972.77	3.21E-14	296296090.2	2085434676
F16	best	**0.313183449**	0.313642786	0.313833832	0.313841468	2.953082635	2.813194614	3.018322626	0.313411811	**0**	0.282743339
	worst	3.06604621	0.313642786	0.313833832	0.313841468	3.058624131	3.049294965	3.054685226	0.313489127	3.039113132	3.077126984
	mean	2.874483091	0.313642786	0.313833832	0.313841468	2.979055804	2.837486143	3.021427866	0.313412095	0.048640529	2.086799332
	median	2.858339278	0.313642786	0.313833832	0.313841468	2.973774936	2.828083314	3.018322626	0.313411811	0	3.015928947
	sd	0.123478977	4.16E-17	4.16E-17	3.05E-17	0.023765099	0.034795524	0.004978037	4.61E-06	0.3529256	1.294374757
F17	best	**2.74E-11**	2.79E-11	2.77E-11	2.74E-11	1682987744	8124313.895	4551480105	2.78E-11	**5.04E-111**	**0**
	worst	11088768702	2.79E-11	2.77E-11	2.74E-11	10720727065	5046542229	8598523561	2.78E-11	5836543916	9258865743
	mean	5418948553	2.79E-11	2.77E-11	2.74E-11	3789413735	69537793.4	4568206643	2.78E-11	36392422.41	306432178.2
	median	5064658325	2.79E-11	2.77E-11	2.74E-11	3014132286	18167648.55	4551480105	2.78E-11	4.49E-55	0
	sd	885758421.1	4.20E-27	4.36E-27	4.52E-27	2163046329	295379702.5	218692204.3	3.39E-15	357170953.9	1576023936
F18	best	0.065264947	0.055534741	0.065630095	0.065272135	11.29812226	4.736804364	9702.905179	**0.046648475**	**0**	**0**
	worst	11.507642	0.065329978	0.065630095	0.065272135	11.44580276	10.09038934	13005.22209	0.058911739	8.480262936	26494.77467
	mean	2.862967765	0.057169632	0.065630095	0.065272135	11.40128365	5.190692848	9721.634657	0.046680024	0.114397103	1112.80629
	median	1.559911469	0.055534741	0.065630095	0.065272135	11.44302638	4.917419975	9702.905179	0.046648475	0	0
	sd	2.575656552	0.003463235	1.04E-17	7.63E-18	0.064592634	0.739157175	241.8816905	0.000605181	0.754170922	5314.558487
F19	best	**0.018649694**	0.018759907	0.018764486	0.018698899	45.32931639	42.22016808	48.92531007	0.018690559	**0**	**0**
	worst	46.70160803	0.018759907	0.018764486	0.018698899	46.47403905	45.65285146	49.46297122	0.018746242	46.27336925	49.15319717
	mean	35.5326682	0.018759907	0.018764486	0.018698899	45.65823826	42.34684559	48.93334111	0.018690719	1.694447863	22.75724929
	median	34.73786127	0.018759907	0.018764486	0.018698899	45.65811462	42.33630398	48.93884261	0.018690559	1.134235292	0
	sd	2.631279754	2.60E-18	2.26E-18	1.91E-18	0.31817398	0.272793571	0.024653396	2.87E-06	5.762532227	23.95566483
F20	best	0.000250786	0.000246696	0.000249866	0.000252195	13.25336872	40.41662168	1985.011032	**0.000246298**	11.51810893	100
	worst	1477.956124	0.000246696	0.000249866	0.000252195	1437.269586	820.6332779	58896.95022	0.000247986	669.1654077	9900
	mean	106.6199225	0.000246696	0.000249866	0.000252195	225.3094727	57.42482498	3143.086142	0.000246305	29.56622817	3803.34477
	median	14.32225309	0.000246696	0.000249866	0.000252195	64.6037593	46.02516801	2022.308723	0.000246298	19.93840466	100
	sd	235.2224648	2.98E-20	4.61E-20	3.39E-20	322.074818	49.11711471	5972.685084	1.05E-07	42.50885548	4678.235965
F21	best	0.001479242	0.001517382	0.001515007	0.001483363	6.978693002	20.75932802	2756.842679	**0.001456843**	0.75338742	1
	worst	323.037448	0.001517382	0.001515007	0.001483363	315.7329145	185.6204308	66396.36154	0.001525863	148.4719403	9801
	mean	20.33662271	0.001517382	0.001515007	0.001483363	57.82178336	25.69890585	4962.328706	0.001457027	1.674645626	1525.750151
	median	2.67861904	0.001517382	0.001515007	0.001483363	28.30660489	22.13469236	2756.842679	0.001456843	0.753953307	1
	sd	48.64424139	2.49E-19	1.73E-19	1.63E-19	69.53001349	11.49308344	6195.735087	3.40E-06	9.058474116	3023.737509
F22	best	0.000267348	0.000268108	0.000264328	0.000267984	978.4644232	0.098573292	11115767807	**0.000122824**	4.79E-05	1.27E-05
	worst	1681.083806	0.000268108	0.000264328	0.000267984	1747.934015	561.5855192	30628194357	0.00019473	247.2965395	66900841776
	mean	115.8899526	0.000268108	0.000264328	0.000267984	1447.842362	2.207505659	11289828183	0.000122984	0.73019635	2358522420
	median	2.416271301	0.000268108	0.000264328	0.000267984	1484.787158	0.135816598	11115767807	0.000122824	8.23E-05	2.10E-05
	sd	300.0316907	4.61E-20	3.39E-20	4.07E-20	279.6249961	26.86476596	1265782073	3.33E-06	11.68303402	11754549618
F23	best	-302.5008149	-262.1755865	-182.7543715	-294.1230158	-295.6844661	**0.000444**	-280.0800046	**-16.54091663**	-270.8437899	-224.6238531
	worst	-2.431610245	-226.2954225	-172.0744886	-204.9868298	-221.660774	0.645516	-240.7591748	-2.618649088	-220.8575849	-215.7436347
	mean	-295.8858845	-259.1191967	-182.7082986	-284.8981188	-290.057282	0.016006	-279.344917	-16.45742878	-266.4443466	-224.6060927
	median	-301.8715247	-260.0368176	-182.7543715	-292.770422	-293.8833165	0.000444	-280.0800046	-16.54091663	-268.0098591	-224.6238531
	sd	19.05436824	4.240456788	0.678269937	15.03776577	9.805001586	0.079374	5.15414341	1.074758903	6.182581047	0.396738107

TABLE 4. *(Continued.)* **Comparison of best, worst, mean, median and standard deviation values for ABC, WOA, BOA PSO, EOSA, DE, GA, HGSO, SOA, and BMO metaheuristic algorithms using the classical benchmark functions over 500 runs and 100 population size.**

F25	best	**1.73E-05**	2.02E-05	2.19E-05	1.99E-05	2.05E-29	0.005818419	0.011111583	2.22E-05	**2.98E-112**	**0**
	worst	24.75787505	2.02E-05	2.19E-05	1.99E-05	15.6011816	5.764924377	3824.507576	2.25E-05	14.05993536	39306.76795
	mean	0.336241769	2.02E-05	2.19E-05	1.99E-05	0.128138027	0.035160231	27.12806016	2.22E-05	0.036388951	1178.47146
	median	0.004759136	2.02E-05	2.19E-05	1.99E-05	1.17E-14	0.007171468	0.011111583	2.22E-05	4.30E-51	0
	sd	1.716830508	3.05E-21	4.07E-21	2.71E-21	1.159489759	0.298540229	299.9427521	5.48E-08	0.653830959	6517.476442
F26	best	1.37E-10	**1.36E-10**	1.39E-10	1.39E-10	2908169157	36768.89195	214910548.7	**1.38E-10**	2.36E-07	**0**
	worst	3850449321	1.36E-10	1.39E-10	1.39E-10	3685252577	1215212126	537648991.6	1.39E-10	526426653.9	1301561560
	mean	2393795142	1.36E-10	1.39E-10	1.39E-10	3376494490	4715971.794	217711828	1.38E-10	1701158.819	59387303.87
	median	2309813185	1.36E-10	1.39E-10	1.39E-10	3399311165	165504.1559	214910548.7	1.38E-10	2.36E-07	0
	sd	243210513.4	1.42E-26	1.94E-26	1.42E-26	300829422.2	58549154.5	19690676.13	3.00E-14	25984261.2	261751624
F27	best	0.00047527	0.00047985	0.000471891	0.000473586	1347.380357	749.9920473	111779.9462	**0.000448786**	**0**	**0**
	worst	1603.785958	0.00047985	0.000471891	0.000473586	1602.675545	1280.232203	149320.5269	0.000470662	1184.58447	250211.9608
	mean	451.6267178	0.00047985	0.000471891	0.000473586	1440.929879	772.4357583	111855.0274	0.000448847	11.72060634	20756.77123
	median	323.4669501	0.00047985	0.000471891	0.000473586	1418.535827	759.7515676	111779.9462	0.000448786	0	0
	sd	270.0398663	8.40E-20	5.15E-20	5.96E-20	79.40879312	44.56932118	1677.1861	1.10E-06	99.55250013	62564.29441
F28	best	1.07E-08	9.70E-09	9.31E-09	1.00E-08	466932.353	12844.73759	580885.0864	**9.25E-09**	**1.67E-114**	**0**
	worst	1022160.738	9.70E-09	9.31E-09	1.00E-08	1011320.133	495929.0155	696318.8725	9.93E-09	884512.6177	1219837.382
	mean	373004.6264	9.70E-09	9.31E-09	1.00E-08	561177.5755	20051.98473	581361.7031	9.25E-09	5008.283742	112653.329
	median	344810.3587	9.70E-09	9.31E-09	1.00E-08	537590.8983	15053.61311	580885.0864	9.25E-09	2.68E-54	0
	sd	81563.46341	1.24E-24	1.36E-24	1.82E-24	106070.0836	28150.07853	7287.632145	3.71E-11	49908.74417	348558.6146
F29	best	4.59E-10	4.56E-10	**4.50E-10**	4.62E-10	921421360.4	16533.59328	23577837147	**4.51E-10**	98.86771563	98.97310275
	worst	1085749209	4.56E-10	4.50E-10	4.62E-10	1105359990	365586773.6	48156845615	4.57E-10	237130959.5	1.37169E+11
	mean	391773892	4.56E-10	4.50E-10	4.62E-10	1041990575	1472601.153	23631867107	4.51E-10	746936.1114	13444857087
	median	327193480.7	4.56E-10	4.50E-10	4.62E-10	1048794876	58273.67518	23577837147	4.51E-10	98.88822056	98.97342237
	sd	141540445.4	6.98E-26	6.98E-26	9.31E-26	68984880.73	17684842.71	1099678157	3.24E-13	11579159.04	40781735453
F30	best	-8836.361886	-41041.09361	-82.284136	-8.37E+13	-10720.24772	**0.026042**	-1316.844379	-1.77E+23	-19627.1811	-5947.963107
	worst	-0.06884986	-14838.49714	-55.03621215	-4236.486476	-5231.572209	0.116496	-1184.623989	-0.073719104	-4632.384271	-5440.211109
	mean	-8466.742315	-40720.29286	-82.15043677	-4.29E+13	-9668.267461	0.051781	-1312.348886	-1.75E+23	-15952.69912	-5553.947557
	median	-8755.05295	-41028.05197	-82.284136	-5.09E+13	-9966.438769	0.039345	-1316.844379	-1.77E+23	-16808.49531	-5440.211109
	sd	735.9984565	1938.702109	1.754877788	3.92E+13	1097.48294	0.035806	23.96218707	1.76E+22	3447.147118	211.6930067
F31	best	9.28E-09	1.08E-08	1.06E-08	1.04E-08	466276.9769	13040.44756	384649.7947	**8.71E-09**	**1.04E-116**	**0**
	worst	1183271.593	1.08E-08	1.06E-08	1.04E-08	1097796.317	507957.0629	740832.8204	8.99E-09	922710.1499	863804.7952
	mean	403966.8473	1.08E-08	1.06E-08	1.04E-08	568102.224	20382.15347	393375.2467	8.71E-09	4580.700889	73204.8105
	median	369094.2037	1.08E-08	1.06E-08	1.04E-08	547682.4572	15390.81415	384649.7947	8.71E-09	6.64E-59	0
	sd	95470.2388	1.08E-24	1.74E-24	1.86E-24	109961.5849	28563.43898	47420.47325	1.61E-11	50858.65247	208770.6124
F32	best	7.03E-167	1.69E-166	3.43E-166	2.37E-165	4.99E+133	445.0616521	2.59E+112	**1.07E-166**	1.73E-58	**0**
	worst	8.34E+147	1.69E-166	3.43E-166	2.37E-165	2.60E+146	3.58E+143	6.68E+134	4.44E-166	4.82E+135	6.82E+142
	mean	1.68E+145	1.69E-166	3.43E-166	2.37E-165	6.15E+144	7.26E+140	1.34E+132	1.26E-166	9.64E+132	2.46E+141
	median	1.40E+106	1.69E-166	3.43E-166	2.37E-165	7.01E+138	447.2618729	2.59E+112	1.07E-166	6.20E-31	0
	sd	3.73E+146	0	0	0	3.61E+145	1.60E+142	2.98E+133	0	2.15E+134	1.27E+142
F33	best	0.010001246	0.010001466	0.010000903	0.010000615	94.68547115	17.33513207	68.80927063	**0.009960159**	**1.52E-57**	**0**
	worst	94.81833964	0.010001466	0.010000903	0.010000615	94.68547115	81.02072384	83.07017055	0.010001165	57.37997572	94.76232851
	mean	87.26062108	0.010001466	0.010000903	0.010000615	94.68547115	19.58660243	68.88238942	0.009960264	0.520898377	4.947618901
	median	86.78228412	0.010001466	0.010000903	0.010000615	94.68547115	18.15536627	68.80927063	0.009960159	4.91E-27	0
	sd	4.179693275	7.81E-19	7.81E-19	1.39E-18	8.53E-15	5.107576845	0.738723155	1.95E-06	4.002551529	19.8768878
F34	best	**2.39E-06**	2.43E-06	2.49E-06	2.44E-06	146564.3791	4151.903716	115950.2416	2.42E-06	**2.56E-92**	**0**
	worst	257895.3838	2.43E-06	2.49E-06	2.44E-06	257656.427	134925.0854	141106.7489	2.43E-06	88039.69375	246579.8016
	mean	208128.671	2.43E-06	2.49E-06	2.44E-06	200833.5123	5709.442348	116122.0799	2.42E-06	687.5080256	12308.21307
	median	204667.8202	2.43E-06	2.49E-06	2.44E-06	204872.6698	4541.823055	115950.2416	2.42E-06	4.43E-29	0
	sd	13021.70178	1.91E-22	3.39E-22	2.33E-22	40515.11285	7456.617438	1764.409793	1.87E-10	5515.873891	41248.60509
F35	best	2.47E-06	**2.43E-06**	2.45E-06	2.44E-06	149354.2611	4296.604798	105811.7963	**2.43E-06**	19.93229264	23.75105029
	worst	257643.1306	2.43E-06	2.45E-06	2.44E-06	261762.2494	138652.6906	142526.1384	2.44E-06	91297.1041	275194.0507
	mean	205778.2306	2.43E-06	2.45E-06	2.44E-06	204473.9907	5798.078168	106135.334	2.43E-06	401.736332	18094.64593
	median	201888.0566	2.43E-06	2.45E-06	2.44E-06	204296.1293	4439.333834	105811.7963	2.43E-06	21.13665172	23.75504078
	sd	13031.29588	3.18E-22	3.18E-22	2.33E-22	39642.75925	7590.203072	2641.313194	5.73E-10	4823.963514	49792.43003
F36	best	4.66E-06	**4.61E-06**	4.71E-06	4.72E-06	66187.97685	1582.956209	4841373.363	**4.61E-06**	**1.63E-119**	**0**
	worst	125317.1139	4.61E-06	4.71E-06	4.72E-06	124112.1235	66662.76425	6254822.832	4.66E-06	39912.02993	12635019.5
	mean	27848.95235	4.61E-06	4.71E-06	4.72E-06	92944.77725	2388.574257	4844727.324	4.61E-06	153.7211311	1875621.432
	median	18307.7666	4.61E-06	4.71E-06	4.72E-06	90158.00805	1737.54854	4841373.363	4.61E-06	2.68E-60	0
	sd	20873.64614	4.24E-22	7.62E-22	5.93E-22	20130.48263	3711.651491	63632.19489	2.90E-09	2089.954545	4042875.797

TABLE 4. *(Continued.)* **Comparison of best, worst, mean, median and standard deviation values for ABC, WOA, BOA PSO, EOSA, DE, GA, HGSO, SOA, and BMO metaheuristic algorithms using the classical benchmark functions over 500 runs and 100 population size.**

F37	best	**0.000305122**	0.023335766	0.024544119	0.024575999	14.4645922	0.409466028	75383.73047	0.011110877	**6.78E-122**	**0**
	worst	25.8151169	0.024556773	0.024544119	0.024575999	25.64204569	13.6419851	127395.5693	0.016303029	11.58066632	250961.7722
	mean	3.086996871	0.023597731	0.024544119	0.024575999	19.72266104	0.56836581	75782.90796	0.011125287	0.060472667	25204.24569
	median	0.118511189	0.023335766	0.024544119	0.024575999	18.95500975	0.447697234	75383.73047	0.011110877	3.98E-60	0
	sd	5.923666804	0.000498763	3.64E-18	3.64E-18	3.843309197	0.757549897	3617.945334	0.000258258	0.658790887	75254.99169
F38	best	8.34E-200	4.53E-200	3.09E-200	2.58E-200	1.03E+153	1.50E+61	1.93E+142	**1.58E-200**	2.11E-62	**0**
	worst	1.21E+178	4.53E-200	3.09E-200	2.58E-200	2.36E+175	1.89E+172	5.25E+172	2.18E-200	4.92E+167	2.69E+169
	mean	2.42E+175	4.53E-200	3.09E-200	2.58E-200	1.34E+173	3.78E+169	1.05E+170	1.58E-200	9.84E+164	5.92E+167
	median	1.22E+162	4.53E-200	3.09E-200	2.58E-200	1.35E+160	3.27E+81	1.93E+142	1.58E-200	1.96E-34	0
F39	best	**2.46E-12**	2.48E-12	2.47E-12	**2.46E-12**	1.49E+11	4087014600	Inf	**2.46E-12**	78062850.06	Inf
	worst	2.60533E+11	2.48E-12	2.47E-12	2.46E-12	2.61E+11	1.35E+11	89553505792	2.47E-12	75834606829	97000259.63
	mean	2.02724E+11	2.48E-12	2.47E-12	2.46E-12	2.05E+11	5631425626	1.43031E+11	2.46E-12	752360824.7	2.58607E+11
	median	1.99043E+11	2.48E-12	2.47E-12	2.46E-12	2.00E+11	4422413197	90384844125	2.46E-12	83482644.88	6069855096
	sd	12748137431	3.43E-28	3.23E-28	3.03E-28	39244378992	7381450963	89553505792	2.02E-16	5182164328	97001094.52
F40	best	3.01E-70	3.60E-70	2.93E-70	2.61E-70	8.61E+48	1200.2428	5490765292	**2.02E-70**	1200.00029	36750667194
	worst	4.50E+54	3.60E-70	2.93E-70	2.61E-70	7.42E+54	1.37E+51	5.41E+39	2.53E-70	2.72E+54	1200.052508
	mean	1.09E+52	3.60E-70	2.93E-70	2.61E-70	5.52E+53	5.37E+48	2.73E+48	2.02E-70	1.08E+52	9.95E+52
	median	3.36E+45	3.60E-70	2.93E-70	2.61E-70	9.18E+50	1201.064629	5.46E+45	2.02E-70	1200.000481	3.18E+51
	sd	2.07E+53	3.58E-86	6.64E-86	4.02E-86	1.69E+54	8.19E+49	5.41E+39	2.66E-72	1.71E+53	1200.052508
F41	best	5.69E-28	5.72E-28	5.76E-28	5.76E-28	3.91E+26	1.13E+27	1.22E+47	**5.53E-28**	5.69E+26	1.75E+52
	worst	1.09E+27	5.72E-28	5.76E-28	5.76E-28	1.04E+27	1.13E+27	1.00E+27	5.65E-28	9.89E+26	2.27E+26
	mean	9.31E+26	5.72E-28	5.76E-28	5.76E-28	5.32E+26	1.13E+27	1.02E+27	5.53E-28	6.56E+26	2.27E+26
	median	9.26E+26	5.72E-28	5.76E-28	5.76E-28	5.00E+26	1.13E+27	1.00E+27	5.53E-28	6.32E+26	2.27E+26
	sd	5.15E+25	7.17E-44	7.17E-44	5.83E-44	1.31E+26	1.61E+24	1.00E+27	5.93E-31	7.69E+25	2.27E+26
F42	best	4.27E-06	4.29E-06	4.26E-06	4.16E-06	31976.73516	1188.933345	1.82E+24	**4.09E-06**	421.3277663	**0**
	worst	108535.2566	4.29E-06	4.26E-06	4.16E-06	108929.7038	48344.70056	37631.68352	4.17E-06	32891.91673	422.2025529
	mean	62929.59696	4.29E-06	4.26E-06	4.16E-06	61436.09831	1620.118778	64503.91697	4.09E-06	559.237889	60903.28867
	median	60073.02152	4.29E-06	4.26E-06	4.16E-06	57366.81193	1310.196265	37707.951	4.09E-06	421.5626519	4801.913145
	sd	7334.315184	5.08E-22	5.08E-22	4.24E-22	25706.21856	2466.571014	37631.68352	5.31E-09	1710.546195	422.2025529
F43	best	**0.000330323**	0.000332408	0.000333684	0.000333589	2266.842103	1643.429176	1300.792126	0.00033233	942.528339	15411.43357
	worst	2479.036365	0.000332408	0.000333684	0.000333589	2493.325685	2169.822427	2007.45709	0.000333577	2069.438537	950.5313267
	mean	1909.413375	0.000332408	0.000333684	0.000333589	2359.039142	1679.482761	2225.834549	0.000332333	964.0878109	2467.418871
	median	1848.634581	0.000332408	0.000333684	0.000333589	2344.668872	1670.803483	2008.384352	0.00033233	944.5733014	1056.128478
	sd	157.8611421	3.79E-20	4.88E-20	3.79E-20	66.81285479	46.64354787	2007.45709	5.80E-08	104.1875831	950.5315637
F44	best	2.09E-29	1.78E-29	1.89E-29	2.03E-29	438320.8487	110941.4238	10.88064761	**1.79E-29**	592240.2576	248.3259763
	worst	2.76E+24	1.78E-29	1.89E-29	2.03E-29	1.10E+24	1.04E+24	1.10E+24	1.93E-29	1.92E+23	1.06326E+12
	mean	1.43E+22	1.78E-29	1.89E-29	2.03E-29	3.47E+21	3.14E+21	1.10E+24	1.79E-29	6.82E+20	8.71E+18
	median	4.57E+18	1.78E-29	1.89E-29	2.03E-29	463752.3183	137938.6535	1.10E+24	1.79E-29	614132.7943	6.98E+17
	sd	1.59E+23	2.24E-45	3.01E-45	2.59E-45	5.64E+22	5.05E+22	1.10E+24	6.78E-32	1.03E+22	1.06326E+12
F45	best	0.305715386	**0.276970281**	0.308726297	0.305443053	2.713949275	2.704585339	134217728	0.30646382	**0**	2.36E+18
	worst	2.834155251	0.307895716	0.308726297	0.305443053	2.830118506	2.794061234	3.013277789	0.307290867	2.6140882	0
	mean	1.740456859	0.281499603	0.308726297	0.305443053	2.735885851	2.704932219	3.041146049	0.306468731	0.045569177	3.078245603
	median	1.619931442	0.276970281	0.308726297	0.305443053	2.723019587	2.704585339	3.018598946	0.30646382	0	2.568373274
	sd	0.264814711	0.010094625	3.61E-17	3.61E-17	0.029287729	0.005105112	3.014733916	6.02E-05	0.286856855	3.015928947
F46	best	2.46E-05	2.46E-05	2.44E-05	**2.43E-05**	14557.36666	409.0949855	0.0066293	2.44E-05	**8.28E-108**	1.070226414
	worst	26682.39023	2.46E-05	2.44E-05	2.43E-05	26064.68463	13842.05832	11754.09992	2.44E-05	9942.048708	0
	mean	20454.91167	2.46E-05	2.44E-05	2.43E-05	20192.37282	565.2503799	16802.43577	2.44E-05	57.57076258	26688.82014
	median	20071.41036	2.46E-05	2.44E-05	2.43E-05	19688.65321	451.293936	12087.53875	2.44E-05	5.07E-48	3146.908833
	sd	1324.806232	2.71E-21	3.22E-21	3.22E-21	3845.341638	751.0738746	11754.09992	4.47E-09	599.829055	0
F47	best	0.005888844	0.00587983	0.005866289	0.005919446	99.71266277	14.12662111	874.3251923	**0.005858814**	**0**	8510.439225
	worst	131.4206624	0.00587983	0.005866289	0.005919446	129.9725436	98.5658473	0	0.005890008	68.10191817	0
	mean	31.30562957	0.00587983	0.005866289	0.005919446	115.3842895	16.63768708	79.9484651	0.005858927	0.393527918	0
	median	7.982565255	0.00587983	0.005866289	0.005919446	114.8670662	14.94988815	0.611058095	0.005858814	0	0
	sd	39.53996915	6.07E-19	7.81E-19	5.64E-19	10.10326971	6.603864967	0	1.84E-06	4.244564029	0
Total Count		**4**	**1**	**1**	**1**	**1**	**2**	**0**	**12**	**19**	**21**
SOTA/EOSA		**15/28**	**5/28**	**2/28**	**4/28**	**0/28**	**0/28**	**0/28**	**28**	**23/21**	**22/20**

Note: state-of-the-art (SOTA)

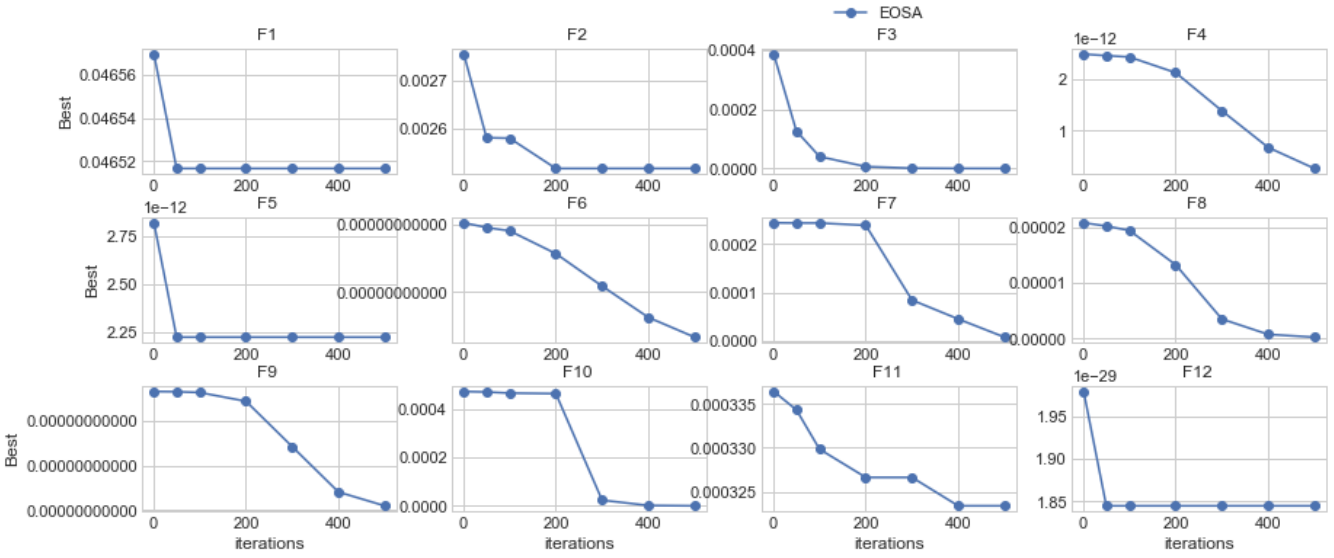


FIGURE 4. Convergent curves of EOSA on some selected standard benchmark functions over 1, 50, 100, 200, 300, 400 and 500 epochs.

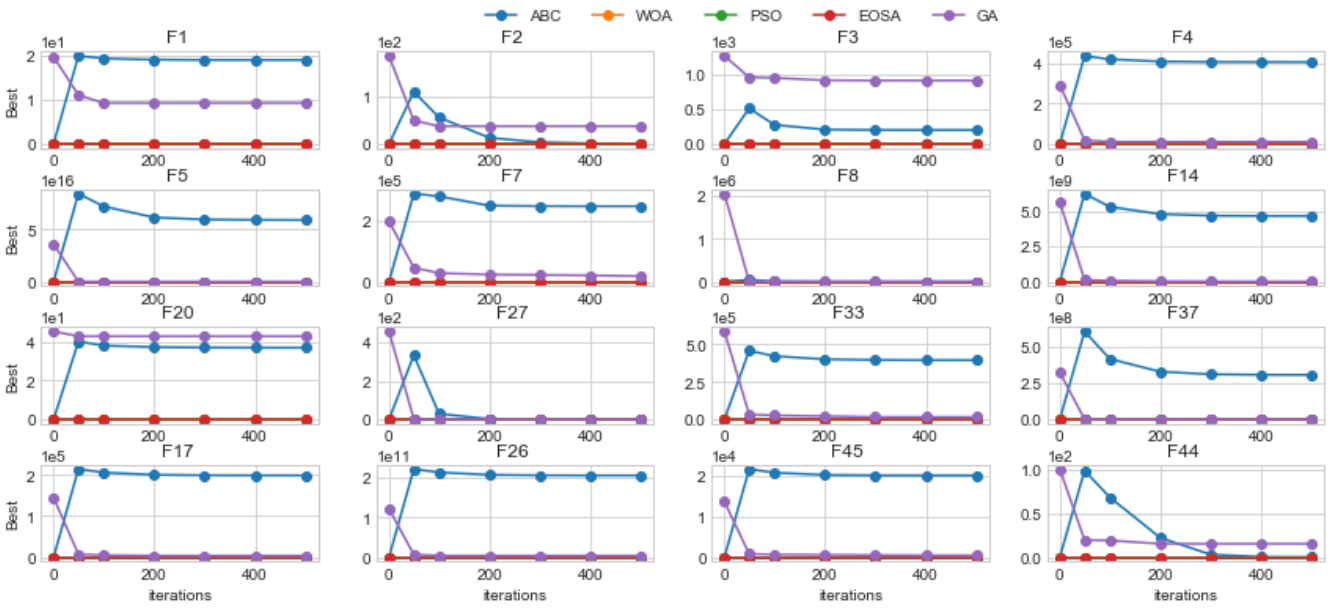
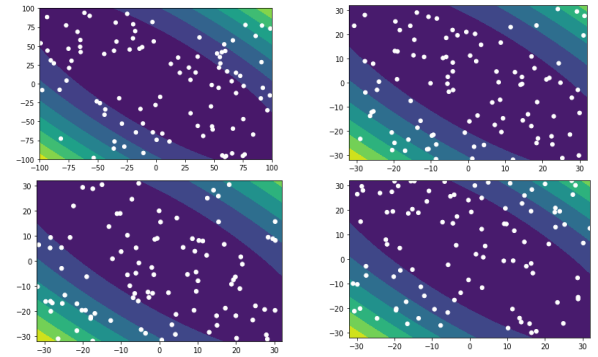


FIGURE 5. Convergent curves of EOSA and related optimization algorithms in some selected standard benchmark functions.

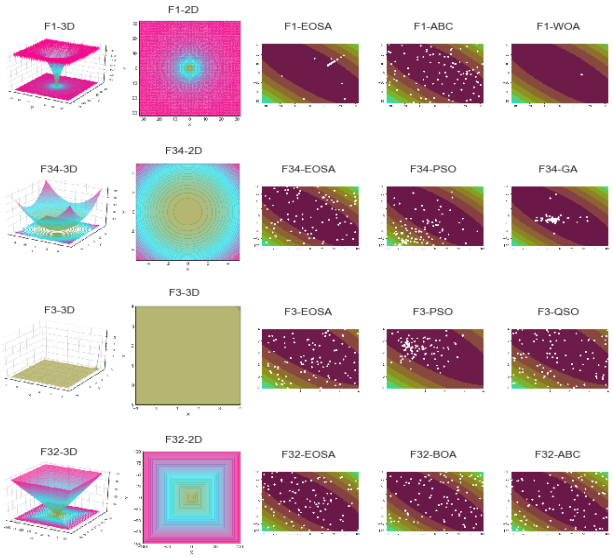
to compare the performance of EOSA with other similar optimization algorithms, the illustrations in the Figures were graphed with all the selected algorithms. The outcome for F1function showed that EOSA appeared to converge its points more closely compared with ABC while WOA showed only a point. However, in the cases of F3 and F34, PSO and GA converged their point more closely than EOSA. Meanwhile, as in the case of the F1 function, we found that EOSA converged its points more closely than BOA and ABC.

In Figure 6a, we show the search history of the proposed EOSA algorithm at the initial iteration. This allows for a comparison of the initial solutions with the final solutions, which is shown in Figure 6b.

The second illustration is shown in Figure 6b. Only in the cases of F7 and F28 was GA able to cluster its points more closely than EOSA, PSO, BOA, DE, and ABC. While PSO attempts to achieve a clearer cluster in F28, EOSA also performs well in clustering points or solutions in F7, F28, F45, and F46.



(a)



(b)

FIGURE 6. a. A perspective view of the search history for functions F1, F3, F32, and F34 for EOSA optimization algorithm at the initial iteration. b. A 3D and 2D perspective view for functions F1, F3, F32, and F34 t their corresponding search history for EOSA and related optimization algorithms.

C. EVALUATION OF EOSA ON CONSTRAINED CEC BENCHMARK FUNCTIONS

The result of experimentation with constrained functions of CEC was also collected and is reported in this subsection. The CEC functions consist of fourteen (14) functions, and we further experimented with thirty (30) hybrids of the 14 CEC functions. The derivation for the hybrid functions is listed in Table 5. C1-C8 and C10 are shifted (S) versions of their corresponding CEC functions, while C9 and C11-C16 are shift-rotate (SR) versions of their corresponding CEC functions. Functions C17-C22 represent shift versions of a combination of some CEC functions, while those of C23-C30 are hybrids of predefined CEC hybrids. The total number of CEC-based functions used for the experiments is forty-four (44).

Tables 6, 7, 8, 9, and 10 outlines the experiment's outcome with the forty-four (44) CEC functions regarding the best values, mean values, standard deviation, worst values, and median values, respectively. The results compare the performances of ABC, WOA, BOA PSO, DE, GA, and HGSO with EOSA based on the categories of outcome each Table outlines. Regarding the best values, the result showed that EOSA outperformed all the related algorithms based on the CEC01, CEC03, CEC05-06, C3-4, C7-8, C15-16, C19-28, and C30 functions. Whereas it failed when compared with those related algorithms based on CEC07-CEC11, C2, C17-18, and C29 functions, we found that it competed with the optimization algorithms based on CEC02 CEC04, CEC12-14, C1, C6, and C9-14functions.

Similarly, the worst values for the 44 functions, as shown in its corresponding Table, also reveal an exciting performance for the proposed EOSA. We discovered that EOSA obtained the same values with WOA, BOA, and PSO using C5, C13, and C16 functions, whereas the corresponding values for ABC, DE, GA and HGSO were significantly large. Also, EOSA leapt in-between its values when compared with WOA, BOA, and PSO based on CEC_F1-4, CEC_F7-8,

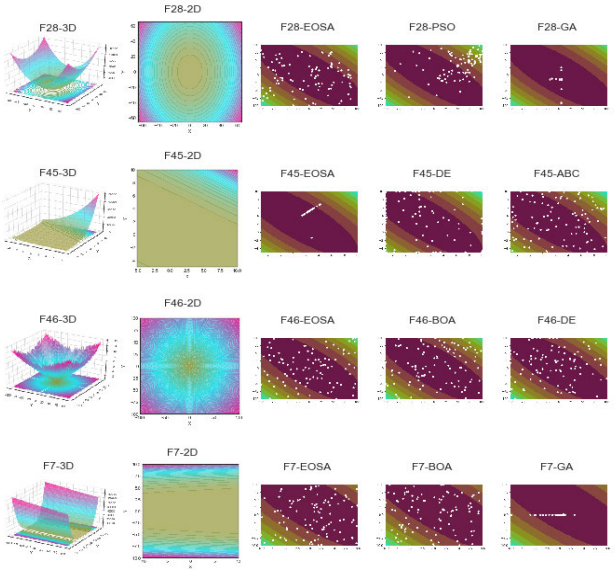


FIGURE 7. A 3D and 2D perspective view for functions F7, F28, F45, and F46 and their corresponding search history for EOSA and related optimization algorithms.

TABLE 5. Listing of thirty (30) hybrids of CEC functions applied to the proposed EOSA algorithm where the shift is (S), and Shift rotate (SF).

Function	Hybrid CEC Functions	Function	Hybrid CEC Functions
C1	S CEC01	**C16**	SR CEC14
C2	S CEC02	**C17**	S [CEC09, CEC08, CEC01]
C3	S CEC03	**C18**	S [CEC02, CEC12, CEC08]
C4	S CEC04	**C19**	S [CEC07, CEC06, CEC04, CEC14]
C5	S CEC05	**C20**	S [CEC12, CEC03, CEC13, CEC08]
C6	S CEC06	**C21**	S [CEC14, CEC12, CEC04, CEC09, CEC01]
C7	S CEC07	**C22**	S [CEC10, CEC11, CEC13, CEC09, CEC05]
C8	S CEC08	**C23**	S (1,2,3,4,5) [C04, C01, C02, C03, C01]
C9	SR CEC08	**C24**	S (1,2,3) [C10, C09, C14]
C10	S CEC09	**C25**	S (1,2,3) [C11, C09, C01]
C11	SR CEC09	**C26**	S (1,2,3,4,5) [C11, C13, C01, C06, C07]
C12	SR CEC10	**C27**	S (1,2,3,4,5) [C14, C09, C11, C06, C01]
C13	SR CEC11	**C28**	S (1,2,3,4,5) [C15, C13, C13, C11, C16, C1]
C14	SR CEC12	**C29**	S (4,5,6) [C17, C18, C19]
C15	SR CEC13	**C30**	S (1,2,3) [C20, C21,C22]

TABLE 6. Comparison of best values for EOSA with ABC, WOA, BOA PSO, DE, GA, and HGSO metaheuristic algorithms using the CEC functions over 20 runs and 100 population size.

	ABC	WOA	BOA	PSO	DE	GA	HGSO	EOSA
CEC 01	2.78E-11	2.78E-11	2.75E-11	2.78E-11	1.67E+09	650045 1	**1.22E-111**	2.75E-11
CEC 02	2.49E-12	2.45E-12	2.48E-12	2.49E-12	1.43E+11	4.17E+09	**2.15E-104**	2.48E-12
CEC 03	1.02E-10	1.02E-10	1.02E-10	1.03E-10	150261 .7	8666.0 65	**2.06E-112**	1.01E-10
CEC 04	**3.68E-12**	3.70E-12	3.73E-12	3.73E-12	1.09E+11	109909 1	98.8298 8	3.71E-12
CEC 05	0.0457 19	0.0457 11	0.0457 04	0.0457 04	20.000 05	18.252 92	**4.44E-16**	0.0456 69
CEC 06	0.0058 31	0	0.0058 99	0.0058 73	0	116.58 91	**0**	0.0057 42
CEC 07	0.0096 43	0.0097 74	0.0097 01	0.0097 43	41.819 79	2.2003 58	**0**	0.0097 35
CEC 08	2.44E-06	2.40E-06	2.45E-06	2.44E-06	136298 .4	5267.7 92	**0**	2.43E-06
CEC 09	2.92E-05	**2.91E-05**	2.92E-05	2.94E-05	14531. 82	564.26 94	0.00127 3	2.94E-05
CEC 10	0.0022 92	0	0.0061 14	0.0061 48	0	41.941 61	**0**	0.0058 66
CEC 11	0.0004 8	**0.0004 73**	0.0004 86	0.0004 86	789.20 33	31.148 32	1.23066 3	0.0004 83
CEC 12	2.42E-06	2.43E-06	**2.40E-06**	2.41E-06	149732	4302.3 36	0.49996 3	2.42E-06
CEC 13	**2.35E-18**	2.40E-18	2.44E-18	2.39E-18	1.53E+17	331603 79	43.5147 3	2.37E-18
CEC 14	0.0197 93	0.0198 27	0.0197 73	0.0198 08	48.826 36	45.963 47	**0**	0.0197 95
C01	2.73E-11	2.81E-11	2.77E-11	**2.76E-11**	1.78E+09	616788 4	175873 0	**2.76E-11**
C02	2.47E-12	2.45E-12	2.46E-12	**2.46E-12**	1.49E+11	3.96E+09	792570 90	2.48E-12
C03	1.01E-10	1.01E-10	1.02E-10	1.02E-10	154556 .7	8413.7 2	511.051 9	**1.00E-10**
C04	4.25E-06	**4.17E-06**	4.26E-06	4.27E-06	30378. 79	1195.3 67	421.332	4.22E-06
C05	**0.001916**	**0.001916**	**0.001916**	**0.001916**	520	518.32 89	503.618	**0.001916**
C06	0.0013 02	**0.0012 98**	**0.0012 98**	**0.0012 98**	698.36 52	618.10 48	600.226 2	0.0012 99
C07	0.0002 28	0.0002 27	0.0002 27	0.0002 26	2044.8 92	761.77 48	701.748 3	**0.0002 24**
C08	0.0003 45	0.0003 44	0.0003 44	**0.0003 43**	2147.2 8	1557.3 67	843.434 3	**0.0003 43**
C09	0.0003 3	0.0003 34	0.0003 33	**0.0003 32**	2255.3 6	1657.8 35	942.756 9	0.0003 33
C10	2.17E-05	2.19E-05	2.19E-05	2.18E-05	34128. 95	22425. 71	1997.69	**2.16E-05**
C11	2.19E-05	**2.17E-05**	**2.17E-05**	2.18E-05	33964. 55	22397. 62	2114.09 1	**2.17E-05**
C12	0.0007 34	0.0007 35	0.0007 33	**0.0007 32**	1228.0 11	1241.8 08	1202.00 2	0.0007 34
C13	**0.0007 63**	**0.0007 63**	**0.0007 63**	**0.0007 63**	1305.9 76	1301.0 55	1301.18 8	**0.0007 63**
C14	**0.0004 11**	0.0004 12	0.0004 14	0.0004 13	1775.7 1	1417.8 28	1400.54 3	0.0004 12
C15	2.50E-08	2.52E-08	2.49E-08	2.58E-08	463211 5	1589.5 11	1529.35 1	**2.44E-08**
C16	0.0006 04	0.0006 04	0.0006 04	0.0006 04	1642.5 58	1642.5 65	1633.59 1	**0.0006**
C17	**6.08E-11**	6.27E-11	6.46E-11	6.51E-11	312456 59	972217 .5	141272. 7	6.25E-11
C18	7.19E-12	7.38E-12	**7.18E-12**	7.31E-12	1.64E+10	145683 28	109015 19	7.33E-12
C19	1.02E-11	**1.01E-11**	1.04E-11	**1.01E-11**	3.72E+09	31531. 58	4856.10 3	**1.01E-11**
C20	6.27E-18	**6.13E-18**	6.15E-18	6.31E-18	1.57E+15	471768 .5	3461.32 9	6.17E-18
C21	1.33E-11	1.30E-11	1.34E-11	1.34E-11	2.16E+09	755298 .5	106452. 7	**1.27E-11**
C22	8.61E-18	8.51E-18	**8.22E-18**	8.94E-18	4.65E+12	45227. 98	2312.38 2	8.37E-18
C23	4.36E-05	4.29E-05	4.27E-05	4.31E-05	3614.3 2	2864.0 19	2427.50 5	**4.23E-05**
C24	5.36E-05	5.35E-05	5.37E-05	5.37E-05	13879. 79	10012. 92	3038.90 1	**5.32E-05**
C25	0.0001 69	**0.0001 68**	0.0001 69	0.0001 69	3201.8 34	3272.1 64	2719.24 8	**0.0001 68**
C26	7.61E-05	7.58E-05	7.66E-05	7.61E-05	7477.1 68	4168.6 48	2831.63 5	**7.52E-05**
C27	4.39E-05	4.47E-05	4.40E-05	**4.33E-05**	5538.1 55	5652.6 08	3297.41 9	4.34E-05
C28	2.77E-05	2.77E-05	**2.73E-05**	2.77E-05	12625. 29	20133. 18	3338.48	2.75E-05
C29	**1.01E-11**	1.02E-11	1.04E-11	1.03E-11	2.28E+10	190893 55	464794 0	1.02E-11
C30	1.09E-17	1.10E-17	1.06E-17	1.11E-17	1.09E-17	161513 5	49862.9 3	**1.05E-17**
Total Count	**7**	**10**	**7**	**10**	**0**	**0**	**9**	**17**

TABLE 7. Comparison of mean values for EOSA with ABC, WOA, BOA PSO, DE, GA, and HGSO metaheuristic algorithms using the CEC functions over 20 runs and 100 population size.

	ABC	WOA	BOA	PSO	DE	GA	HGSO	EOSA
CEC 01	5.33E+09	2.78E-11	**2.75E-11**	2.78E-11	3.83E+09	705177 86	225719 97	**2.75E-11**
CEC 02	2.06E+11	**2.45E-12**	2.48E-12	2.49E-12	1.97E+11	5.72E+09	3.43E+08	2.48E-12
CEC 03	265756 .9	1.02E-10	1.02E-10	1.03E-10	229059 .3	19263. 3	5501.9 43	**1.01E-10**
CEC 04	8.86E+10	**3.70E-12**	3.73E-12	3.73E-12	1.27E+11	1.78E+08	1.21E+08	3.71E-12
CEC 05	20.143 94	0.0457 11	0.0457 04	0.0457 04	20.042 48	18.593 28	0.5382 32	**0.0456 69**
CEC 06	18.672 83	4.56E-05	0.0058 99	0.0058 73	2.5641 57	118.74	0.2360 27	**0.0057 43**
CEC 07	53.297 14	0.0097 74	**0.0097 01**	0.0097 43	54.294 49	2.5507 41	0.1217	0.0097 35
CEC 08	206645 .9	**2.40E-06**	2.45E-06	2.44E-06	201318 .7	6854.1 59	555.41 32	2.43E-06
CEC 09	19317. 58	**2.91E-05**	2.92E-05	2.94E-05	17770. 2	743.72 32	52.636 96	2.94E-05
CEC 10	2.6371 36	8.98E-06	0.0061 14	0.0061 48	**0.0057 29**	44.056 48	0.0001 68	0.0058 68
CEC 11	1068.9 86	0.0004 73	0.0004 86	0.0004 86	1073.1 52	39.110 5	3.1610 78	**0.0004 83**
CEC 12	211335	**2.43E-06**	**2.40E-06**	2.41E-06	207142 .2	5722.9 81	404.09 77	2.42E-06
CEC 13	6.38E+16	2.40E-18	2.44E-18	**2.39E-18**	1.53E+17	5.95E+13	4.04E+13	2.37E-18
CEC 14	48.557 28	0.0198 27	0.0197 73	0.0198 08	49.021 63	46.347 48	1.2150 93	**0.0197 95**
C01	5.17E+09	2.81E-11	2.77E-11	**2.76E-11**	3.85E+09	687089 17	353876 99	**2.76E-11**
C02	2.06E+11	**2.45E-12**	2.46E-12	2.46E-12	2.07E+11	5.63E+09	4.76E+08	2.48E-12
C03	264273 .1	1.01E-10	1.02E-10	1.02E-10	229257 .4	18720. 24	7385.1 35	**1.00E-10**
C04	66642. 05	**4.17E-06**	4.26E-06	4.27E-06	60684. 81	1629.6 58	601.29 72	4.22E-06
C05	519.14 11	**0.001916**	**0.001916**	**0.001916**	520.04 47	518.65 85	503.88 36	**0.001916**
C06	710.69 3	**0.0012 98**	**0.0012 98**	**0.0012 98**	713.73 33	620.36 31	600.71 76	0.0012 99
C07	2558.9 92	0.0002 27	0.0002 27	0.0002 26	2521.2 63	776.65 96	707.61 67	**0.0002 24**
C08	1817.5 85	0.0003 44	0.0003 44	**0.0003 43**	2229.9 91	1581.7 96	858.14 52	**0.0003 43**
C09	1918.4 45	0.0003 34	0.0003 33	**0.0003 32**	2357.7 54	1683.9 17	954.23 28	0.0003 33
C10	23020	2.19E-05	2.19E-05	2.18E-05	34606. 45	23387. 87	2446.7 36	**2.16E-05**
C11	22642. 04	**2.17E-05**	**2.17E-05**	2.18E-05	34535. 5	23444. 82	2538.9 94	**2.17E-05**
C12	1200.5 47	0.0007 35	0.0007 33	**0.0007 32**	1231.8 39	1244.1 55	1203.9 38	0.0007 34
C13	1304.7 38	**0.0007 63**	**0.0007 63**	**0.0007 63**	1307.0 94	1301.1 52	1301.3 02	**0.0007 63**
C14	1915.7 52	**0.0004 12**	0.0004 14	0.0004 13	1909.9 53	1422.2 89	1401.6 56	**0.0004 12**
C15	399122 7	2.52E-08	2.49E-08	2.58E-08	825097 6	5798.6 19	2590.2 3	**2.44E-08**
C16	1631.4 47	0.0006 04	0.0006 04	0.0006 04	1642.9 66	1642.5 67	1635.0 98	**0.0006**
C17	6.52E+08	6.27E-11	6.46E-11	6.51E-11	2.35E+08	119450 24	629309 4	**6.25E-11**
C18	3.41E+10	7.38E-12	**7.18E-12**	7.31E-12	3.17E+10	3.32E+08	1.25E+08	7.34E-12
C19	1.01E+10	**1.01E-11**	1.04E-11	**1.01E-11**	1.07E+10	262624 24	143989 54	**1.01E-11**
C20	3.09E+15	**6.13E-18**	6.15E-18	6.31E-18	9.07E+15	3.96E+12	8.23E+12	6.17E-18
C21	6.69E+09	1.30E-11	1.34E-11	1.34E-11	5.96E+09	291934 44	179065 16	**1.27E-11**
C22	5.71E+14	8.51E-18	**8.22E-18**	8.94E-18	7.38E+14	7.24E+11	5.42E+12	8.37E-18
C23	5648.3 71	4.29E-05	4.27E-05	4.31E-05	4936.6	2915.3 59	2442.9 2	**4.23E-05**
C24	10545. 33	5.35E-05	5.37E-05	5.37E-05	14160. 14	10305. 09	3177.7 54	**5.32E-05**
C25	3470.6 54	**0.0001 68**	0.0001 69	0.0001 69	3362.3 74	3294.2 78	2742.1 84	**0.0001 68**
C26	8109.3 53	7.58E-05	7.66E-05	7.61E-05	8290.1 79	4250.1 87	2857.9 24	**7.52E-05**
C27	7333.6 35	4.47E-05	4.40E-05	**4.33E-05**	6777.9 38	5851.1 02	3486.6 84	4.34E-05
C28	17319. 94	2.77E-05	**2.73E-05**	2.77E-05	14334	20220. 09	3719.3 04	2.75E-05
C29	2.68E+10	**1.02E-11**	1.04E-11	1.03E-11	3.07E+10	2.13E+08	938326 03	**1.02E-11**
C30	3.29E+14	1.10E-17	1.06E-17	1.11E-17	1.09E-17	6.25E+11	8.25E+11	**1.05E-17**
AVG	1.54E+15	**1.89E-03**	2.15E-03	2.16E-03	3.71E+15	1.47E+12	1.25E+12	2.15E-03

CEC_F10-12, CEC_F14, C1, C3, C8-9, C11-12, C14-15, C17-20, C22, C25, and C29-30 functions, whereas it recorded an overall good performance for CEC05-06, CEC13, C4, C7, C10, C21, C23-24, and C26-28.

TABLE 8. Comparison of standard deviation values for EOSA with ABC, WOA, BOA PSO, DE, GA, and HGSO metaheuristic algorithms using the CEC functions over 20 runs and 100 population size.

	ABC	WOA	BOA	PSO	DE	GA	HGSO	EOSA
CEC 01	8.44E+08	**3.39E-27**	4.36E-27	3.88E-27	2.13E+09	3.1E+08	2.43E+08	1.46 E-15
CEC 02	1.3E+10	4.64E-28	3.03E-28	**2.83E-28**	3.92E+10	7.53E+09	4.41E+09	9.11 E-17
CEC 03	124317.8	**1.36E-26**	2.07E-26	2.13E-26	80504.07	23773.8	94634.65	3.19 E-14
CEC 04	9.64E+09	5.65E-28	6.26E-28	7.88E-28	7.2E+09	2.26E+09	1.59E+09	**1.35 E-16**
CEC 05	0.957905	**5.90E-18**	4.16E-18	**5.90E-18**	0.146082	0.531952	2.67635	9.25 E-07
CEC 06	28.93125	0.000486	1.08E-18	**6.94E-19**	9.64147	2.394845	2.535846	8.98 E-06
CEC 07	3.287874	**9.54E-19**	1.04E-18	1.13E-18	9.165075	1.842338	1.222283	5.86 E-07
CEC 08	13463.5	**3.18E-22**	3.81E-22	**3.18E-22**	44270.92	7394.998	5414.126	7.16 E-11
CEC 09	1125.511	**4.91E-21**	**4.91E-21**	4.57E-21	2475.684	812.3379	525.8539	7.61 E-09
CEC 10	8.081229	0.000189	9.54E-19	1.04E-18	0.079924	3.439061	0.003118	**1.80 E-05**
CEC 11	67.00617	7.59E-20	**6.78E-20**	8.13E-20	203.6223	38.53338	22.77025	2.51 E-08
CEC 12	13017.52	2.75E-22	**2.12E-22**	3.18E-22	40966.34	7434.063	4920.48	7.17 E-11
CEC 13	1.56E+16	**3.08E-34**	**3.08E-34**	3.27E-34	16	1.16E+15	7.79E+14	6.53 E-22
CEC 14	2.18245	2.26E-18	**2.60E-18**	2.26E-18	0.160205	0.52921	7.091001	1.36 E-06
C01	9.38E+08	5.65E-27	4.04E-27	**3.72E-27**	2.18E+09	3.19E+08	3.53E+08	1.54 E-15
C02	1.27E+10	3.84E-28	3.43E-28	**2.63E-28**	3.91E+10	7.35E+09	5.18E+09	5.13 E-17
C03	97428.44	**1.42E-26**	**1.42E-26**	**1.42E-26**	85524.47	21249.39	92545.77	3.41 E-14
C04	7311.002	6.56E-22	**5.08E-22**	7.41E-22	25164.55	2379.42	1832.528	1.73 E-09
C05	23.24212	3.79E-19	2.93E-19	**2.49E-19**	0.144908	0.504494	1.673248	7.25 E-10
C06	31.98589	1.52E-19	**1.30E-19**	1.63E-19	10.84148	6.159877	4.5231	8.59 E-09
C07	141.8918	4.07E-20	3.39E-20	**3.25E-20**	329.3684	66.00672	52.04919	7.57 E-09
C08	156.4804	3.79E-20	**3.52E-20**	**3.52E-20**	78.9143	44.58426	95.90718	4.04 E-08
C09	158.7789	4.34E-20	**3.25E-20**	4.88E-20	82.38662	45.19802	86.11494	1.49 E-08
C10	3268.026	2.20E-21	2.54E-21	**1.86E-21**	423.4968	1633.723	2997.41	2.54 E-09
C11	3252.009	4.24E-21	**3.05E-21**	3.90E-21	560.3687	1641.823	2811.823	1.13 E-09
C12	54.434002	**1.03E-19**	9.76E-20	**1.03E-19**	4.051227	3.398669	5.949288	6.74 E-08
C13	58.4083	**8.67E-20**	1.03E-19	1.14E-19	0.786165	0.315701	0.26207	9.74 E-10
C14	88.99855	7.05E-20	**4.61E-20**	6.51E-20	91.06467	19.00835	10.87745	2.92 E-09
C15	989141.6	4.63E-24	**2.98E-24**	4.14E-24	244208.5	78999.93	20600.54	3.48 E-11
C16	73.05729	8.13E-20	**4.34E-20**	8.67E-20	0.325375	0.037071	1.887207	2.35 E-07
C17	2E+08	**9.37E-27**	1.62E-26	9.69E-27	3.7E+08	62967751	65905146	8.98 E-15
C18	4.48E+09	1.62E-27	1.29E-27	**1.21E-27**	1.26E+10	1.79E+09	1.26E+09	1.91 E-15
C19	2.5E+09	**1.13E-27**	1.37E-27	1.37E-27	6.61E+09	3.95E+08	2.2E+08	1.68 E-15
C20	2.2E+15	1.00E-33	1.16E-33	**9.63E-34**	7.23E+15	8.15E+13	1.33E+14	2.07 E-21
C21	1.71E+09	**1.94E-27**	2.58E-27	2.02E-27	4.08E+09	2.93E+08	2.81E+08	1.70 E-15
C22	6.8E+14	1.16E-33	1.39E-33	**8.47E-34**	1.44E+15	1.49E+13	1.03E+14	2.69 E-21
C23	505.2743	6.10E-21	**5.76E-21**	6.10E-21	1319.089	159.8902	144.9136	5.58 E-09
C24	1142.609	8.13E-21	8.13E-21	**6.10E-21**	294.4141	523.598	873.6135	7.63 E-09
C25	172.4484	**2.17E-20**	2.57E-20	2.71E-20	175.9663	26.39227	92.35979	1.41 E-08
C26	447.5749	8.81E-21	**7.45E-21**	8.81E-21	609.1627	214.4841	201.5054	1.23 E-08
C27	555.7132	6.44E-21	5.08E-21	**4.74E-21**	1223.351	231.5779	574.4783	9.68 E-09
C28	1546.707	**3.73E-21**	5.76E-21	4.07E-21	1742.996	532.2733	2264.442	2.74 E-09
C29	2.8E+09	1.86E-27	1.62E-27	**1.37E-27**	6.16E+09	1.15E+09	9.03E+08	1.54 E-15
C30	4.95E+14	1.54E-33	1.62E-33	**1.39E-33**	1.05E-21	1.24E+13	1.7E+13	3.53 E-21
AVG	4.31E+14	1.54E-05	**2.45E-19**	2.74E-19	1.97E+14	2.89E+13	2.34E+13	6.90 E-07

TABLE 9. Comparison of worst values for EOSA with ABC, WOA, BOA PSO, DE, GA, and HGSO metaheuristic algorithms using the CEC functions over 20 runs and 100 population size.

	ABC	WOA	BOA	PSO	DE	GA	HGSO	EOSA
CEC 01	1.12E+10	2.78E-11	**2.75E-11**	2.78E-11	1.04E+10	5.59E+09	4.16E+09	**2.75E-11**
CEC 02	2.59E+11	**2.45E-12**	2.48E-12	2.49E-12	2.56E+11	1.38E+11	8.22E+10	**2.48E-12**
CEC 03	2687918	**1.02E-10**	**1.02E-10**	1.03E-10	758263.8	210260.8	194948 0	**1.02E-10**
CEC 04	1.36E+11	**3.70E-12**	3.73E-12	3.73E-12	1.33E+11	4.71E+10	2.92E+10	3.71E-12
CEC 05	21.52029	0.045711	0.045704	0.045704	21.17672	21.51461	20.26211	**0.045668**
CEC 06	131.4872	0.005909	0.005899	0.005873	90.86362	129.6001	44.31723	**0.005825**
CEC 07	67.33259	0.009774	**0.009701**	0.009743	66.20478	35.20829	21.53974	0.009748
CEC 08	262494.3	**2.40E-06**	2.45E-06	2.44E-06	262963.7	136488.5	89507.12	2.43E-06
CEC 09	23595.96	**2.91E-05**	2.92E-05	2.94E-05	23589.75	14244.41	8958.105	2.95E-05
CEC 10	63.28519	**0.004209**	0.006114	0.006148	1.718978	64.79516	0.068601	0.006093
CEC 11	1327.766	**0.000473**	0.000486	0.000486	1347.94	717.9246	422.4248	0.000484
CEC 12	262772.6	2.43E-06	**2.40E-06**	2.41E-06	262112.5	134429.3	88298.64	2.42E-06
CEC 13	1.54E+17	2.40E-18	2.44E-18	2.39E-18	1.53E+17	2.58E+16	1.71E+16	**2.38E-18**
CEC 14	49.52848	0.019827	**0.019773**	0.019808	49.48733	49.33462	49.38548	0.019824
C01	1.15E+10	2.81E-11	**2.77E-11**	2.76E-11	1.11E+10	5.67E+09	6.19E+09	**2.77E-11**
C02	2.58E+11	**2.45E-12**	2.46E-12	2.46E-12	2.59E+11	1.33E+11	1.01E+11	2.48E-12
C03	2103257	**1.01E-10**	1.02E-10	1.02E-10	917649.9	250629.4	190897 7	**1.01E-10**
C04	105914.3	**4.17E-06**	4.26E-06	4.27E-06	105781.7	46273.45	30893.76	4.26E-06
C05	521.5171	**0.001916**	**0.001916**	**0.001916**	521.1812	521.5299	519.7741	**0.001916**
C06	730.2922	**0.001298**	**0.001298**	**0.001298**	730.5756	696.6942	670.6866	0.001299
C07	3079.45	0.000227	0.000227	0.000226	3029.567	1923.113	1559.444	**0.000224**
C08	2406.95	0.000344	0.000344	**0.000343**	2400.625	2074.284	1962.092	**0.000343**
C09	2506.628	0.000334	0.000333	**0.000332**	2506.397	2177.013	2044.832	0.000334
C10	36158.47	2.19E-05	2.19E-05	2.18E-05	36205.08	34710.49	33711.3	**2.17E-05**
C11	36240.38	**2.17E-05**	**2.17E-05**	2.18E-05	36030.53	34829.66	33964.02	**2.17E-05**
C12	1264.28	0.000735	0.000733	**0.000732**	1254.941	1261.787	1254.078	0.000734
C13	1308.329	**0.000763**	**0.000763**	**0.000763**	1308.369	1306.122	1304.812	**0.000763**
C14	2070.763	**0.000412**	0.000414	0.000413	2042.449	1748.141	1589.229	**0.000412**
C15	104904.78	2.52E-08	**2.49E-08**	2.58E-08	109146.94	174813.5	444906.5	2.51E-08
C16	1643.261	**0.000604**	**0.000604**	**0.000604**	1643.697	1643.287	1643.891	**0.000604**
C17	2.25E+09	**6.27E-11**	6.46E-11	6.51E-11	2.25E+09	1.17E+09	1.18E+09	**6.27E-11**
C18	5.88E+10	7.38E-12	**7.18E-12**	7.31E-12	6.01E+10	3.2E+10	2.18E+10	7.35E-12
C19	2.73E+10	**1.01E-11**	1.04E-11	**1.01E-11**	2.44E+10	8.51E+09	4.48E+09	**1.01E-11**
C20	2.05E+16	**6.13E-18**	6.15E-18	6.31E-18	2.19E+16	1.81E+15	2.51E+15	6.21E-18
C21	1.81E+10	1.30E-11	1.34E-11	1.34E-11	1.79E+10	6.09E+09	5.93E+09	**1.27E-11**
C22	7.01E+15	8.51E-18	**8.22E-18**	8.94E-18	6.17E+15	3.32E+14	2.19E+15	8.43E-18
C23	8511.456	4.29E-05	4.27E-05	4.31E-05	8622.02	5437.081	4831.978	**4.24E-05**
C24	15051.17	5.35E-05	5.37E-05	5.37E-05	15014.57	14393.9	13420.02	**5.34E-05**
C25	3955.273	**0.000168**	0.000169	0.000169	3923.978	3536.828	3525.063	**0.000168**
C26	9478.512	7.58E-05	7.66E-05	7.61E-05	9528.773	7271.734	6069.047	**7.55E-05**
C27	10435.93	4.47E-05	4.40E-05	**4.33E-05**	10466.53	7279.321	8191.818	4.35E-05
C28	23982.63	2.77E-05	**2.73E-05**	2.77E-05	22936.38	24198.69	23592.56	2.76E-05
C29	4.24E+10	**1.02E-11**	1.04E-11	1.03E-11	4.14E+10	2.11E+10	1.53E+10	**1.02E-11**
C30	5.34E+15	1.10E-17	**1.06E-17**	1.11E-17	1.09E-17	2.77E+14	3.76E+14	**1.06E-17**
AVG	4.25E+15	**2.11E-03**	2.15E-03	2.16E-03	4.12E+15	6.41E+14	5.04E+14	2.15E-03

Mean values for EOSA, ABC, WOA, BOA, and PSO demonstrated a very close outcome with no clear leading algorithm in the cases of CEC01-04, CEC06, CEC08-10, CEC12, CEC14, C1, C4, C8-9, C11-12, C14, C18-20, C22,

TABLE 10. Comparison of median values for EOSA with ABC, WOA, BOA PSO, DE, GA, and HGSO metaheuristic algorithms using the CEC functions over 20 runs and 100 population size.

	ABC	WOA	BOA	PSO	DE	GA	HGSO	EOSA
CEC 01	4.97E+09	2.78E-11	**2.75E-11**	2.78E-11	3.16E+09	174050 89	7.98E-53	**2.75E-11**
CEC 02	2.02E+11	**2.45E-12**	2.48E-12	2.49E-12	1.95E+11	4.44E+09	1.11E-54	2.48E-12
CEC 03	251561	1.02E-10	1.02E-10	1.03E-10	211966 .5	12804. 61	3.54E-59	**1.01E-10**
CEC 04	8.5E+10	**3.70E-12**	3.73E-12	3.73E-12	1.3E+11	535928 3	98.872 37	3.71E-12
CEC 05	20.024 51	0.0457 11	0.0457 04	0.0457 04	20.000 25	18.404 64	6.22E-16	**0.0456 69**
CEC 06	4.2127 8	**0**	0.0058 99	0.0058 73	0.0003 95	118.33 58	**0**	0.0057 42
CEC 07	52.420 71	0.0097 74	**0.0097 01**	0.0097 43	54.263 27	2.2517 02	0	0.0097 35
CEC 08	202970 .3	**2.40E-06**	2.45E-06	2.44E-06	193545 .9	5656.1 78	0	2.43E-06
CEC 09	19028. 6	**2.91E-05**	2.92E-05	2.94E-05	17241. 84	608.21 15	0.0012 73	2.94E-05
CEC 10	0.1559 3	**0**	0.0061 14	0.0061 48	3.57E-06	42.455 88	**0**	0.0058 66
CEC 11	1050.1 16	**0.0004 73**	0.0004 86	0.0004 86	1037.3 4	32.656 25	1.3055 09	0.0004 83
CEC 12	208014 .3	2.43E-06	**2.40E-06**	2.41E-06	203259 .8	4502.9 93	0.4999 63	2.42E-06
CEC 13	5.70E+16	2.40E-18	2.44E-18	2.39E-18	1.53E+17	1.44E+09	43.795 2	**2.37E-18**
CEC 14	48.562 86	0.0198 27	**0.0197 73**	0.0198 08	48.987 68	46.164 93	0	0.0197 95
C01	4.81E+09	2.81E-11	2.77E-11	2.76E-11	3.07E+09	158002 57	253928 8	**2.76E-11**
C02	2.02E+11	**2.45E-12**	2.46E-12	2.46E-12	2.07E+11	4.48E+09	833101 10	2.48E-12
C03	251898 .3	1.01E-10	1.02E-10	1.02E-10	207464 .7	12286. 88	569.22 32	**1.00E-10**
C04	63680. 08	**4.17E-06**	4.26E-06	4.27E-06	53106. 6	1289.3 25	421.53 14	4.22E-06
C05	520.02 64	**0.0019 16**	**0.0019 16**	**0.0019 16**	520	518.44 32	503.61 96	**0.0019 16**
C06	710.55 76	**0.0012 98**	**0.0012 98**	**0.0012 98**	713.89 83	619.00 61	600.23 45	0.0012 99
C07	2526.2 15	0.0002 27	0.0002 27	0.0002 26	2476.4 66	766.65 83	701.77 71	**0.0002 24**
C08	1756.3 55	0.0003 44	0.0003 44	**0.0003 43**	2205.9 42	1567.9 59	844.69 82	**0.0003 43**
C09	1857.3 48	0.0003 34	0.0003 33	**0.0003 32**	2351.1 15	1671.5 62	944.64 68	0.0003 33
C10	21565. 89	2.19E-05	2.19E-05	2.18E-05	34546. 78	22826. 18	2051.6 25	**2.16E-05**
C11	21202. 09	**2.17E-05**	**2.17E-05**	2.18E-05	34411. 18	23005. 81	2162.1 77	**2.17E-05**
C12	1200.0 98	0.0007 35	0.0007 33	**0.0007 32**	1230.8 66	1243.1 01	1202.8 01	0.0007 34
C13	1307.2 77	**0.0007 63**	**0.0007 63**	**0.0007 63**	1306.9 91	1301.1 2	1301.2 57	**0.0007 63**
C14	1908.9 45	**0.0004 12**	0.0004 14	0.0004 13	1894.8 14	1419.2 33	1400.5 75	**0.0004 12**
C15	357456 0	2.52E-08	2.49E-08	2.58E-08	864324 3	1592.3	1529.5 79	**2.44E-08**
C16	1633.8 13	0.0006 04	0.0006 04	0.0006 04	1643.0 18	1642.5 65	1634.6 36	**0.0006**
C17	5.76E+08	6.27E-11	6.46E-11	6.51E-11	923633 16	287031 1	249692 .2	**6.25E-11**
C18	3.25E+10	7.38E-12	**7.18E-12**	7.31E-12	2.81E+10	502563 34	146498 20	7.33E-12
C19	9.14E+09	**1.01E-11**	1.04E-11	**1.01E-11**	8.3E+09	218340 .8	6335.7 23	**1.01E-11**
C20	2.34E+15	**6.13E-18**	6.15E-18	6.31E-18	7.03E+15	117164 50	3736.1 84	6.17E-18
C21	6.05E+09	1.30E-11	1.34E-11	1.34E-11	4.54E+09	243831 2	166857 .2	**1.27E-11**
C22	3.46E+14	8.51E-18	**8.22E-18**	8.94E-18	8.57E+13	194574 2	2390.8 39	8.37E-18
C23	5473.7 4	4.29E-05	4.27E-05	4.31E-05	4454.5 7	2878.7 11	2428.2 72	**4.23E-05**
C24	10065. 35	5.35E-05	5.37E-05	5.37E-05	14042. 95	10141. 77	3059.5 51	**5.32E-05**
C25	3444.5 07	**0.0001 68**	0.0001 69	0.0001 69	3299.4 47	3287.1 33	2726.6 05	**0.0001 68**
C26	8002.9 61	7.58E-05	7.66E-05	7.61E-05	8208.7 12	4185.8 72	2835.7 85	**7.52E-05**
C27	7171.6 5	4.47E-05	4.40E-05	**4.33E-05**	6358.8 64	5809.8 33	3329.5 76	4.34E-05
C28	16715. 8	2.77E-05	**2.73E-05**	2.77E-05	13773. 55	20133. 18	3352.7 17	2.75E-05
C29	2.57E+10	**1.02E-11**	1.04E-11	1.03E-11	3E+10	476194 82	573478 9	**1.02E-11**
C30	1.85E+14	1.10E-17	1.06E-17	1.11E-17	1.09E-17	255824 27	76134. 32	**1.05E-17**
AVG	1.36E+15	**1.88E-03**	2.15E-03	2.16E-03	3.65E+15	2.40E+08	2.43E+06	2.15E-03

C25, and C27-29 functions. But for the CEC05, CEC07, CEC11, CEC13, C3, C7, C10, C15-17, C21, C23-24, C26, and C30 functions, the mean values of EOSA outperformed all related algorithms including ABC, DE, GA, and HGSO, while it lagged for the C2 and C6 functions. We also observed that the same values were obtained for EOSA, WOA, BOA, and PSO for the C5 and C13 functions.

In Figure 8, the curves of the convergence of EOSA on the solutions of CEC benchmark functions are shown. We graphed the outcome of the optimized solutions over 1, 50, 100, 200, 300, 400 and 500 epochs. To demonstrate this, we plotted the curves of CEC_F1-14 and C_F1. The outcome of the pattern of the curves showed that the solutions were well optimized over the epochs. The convergence of the EOSA algorithm based on the best fit compared with the best fits of ABC, WOA, BOA, PSO, DE, GA, and HGSO we plotted as shown in Figure 9. All the curves representing each optimization algorithm showed a descent from high to low based on their best values. Although the curves of ABC and occasionally that of GA were often seen overshooting in values compared with others, we confirm that this is unrelated to the significantly large values obtained by these algorithms (ABC and GA, with DE and HGSO inclusive). EOSA, WOA, BOA, and PSO curves appear to lie low, though with marginal descent continuously overshadowed by ABC and GA.

D. COMPUTATIONAL TIME COST FOR EOSA AND RELATED ALGORITHMS ON BENCHMARK AND CEC FUNCTIONS

The computational time required for running the optimization algorithms discussed in previous subsections was recorded and reported. We took an average of the computation time for all the forty-seven (47) standard benchmark functions and the forty-four (44) CEC-based functions. The outcome of these averages for ABC, WOA, BOA, PSO, DE, GA, and HGSO compared to EOSA are listed in Table 11. We discovered that the computational requirement of EOSA reports a minimal CPU time compared with other algorithms.

The computational requirement for executing all the algorithms on the standard benchmark functions was graphed as shown in Figure 10. We randomly selected some of these functions: F1, F2, F3, F4, F5, F7, F8, F14, F20, F27, F33, F37, F17, F26, D45, and F44. Although EOSA consumed the lowest CPU time in all cases, we noticed that the discrepancies in the case of F27, F33, F2, and F14 were quite marginal. On the other hand, EOSA's computational requirements in the case of F1, F4, and F8 were significantly low compared with related optimization algorithms.

Similarly, in Figure 11, the computational requirement for CEC-based functions was illustrated for some selected functions namely CEC01, CEC02, CEC03, CEC04, CEC05, CEC06, CEC07, CEC10, CEC11, CEC12, C1, C9, C25, and C30. In all cases, the CPU time for training EOSA was lower than other related algorithms. While EOSA showed less CPU time, GA and ABC were more demanding for this same computational resource. We note that the unusual computational time accounted for in some of the algorithms might

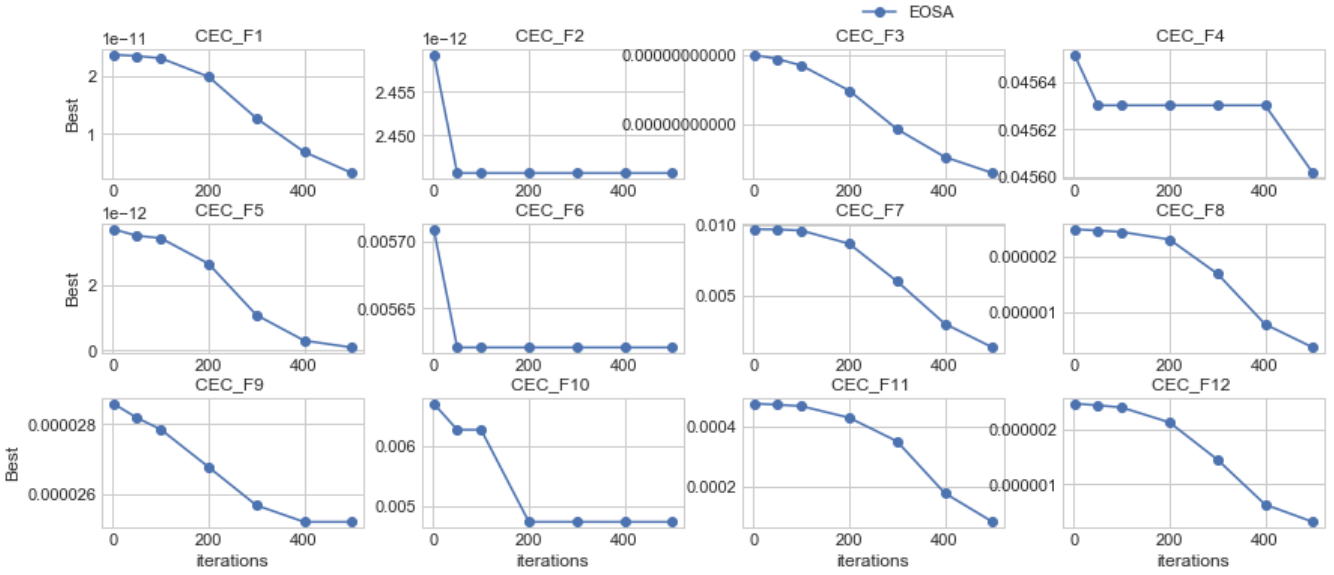


FIGURE 8. Convergent curves of EOSA on some selected CEC benchmark functions over 1, 50, 100, 200, 300, 400 and 500 epochs.

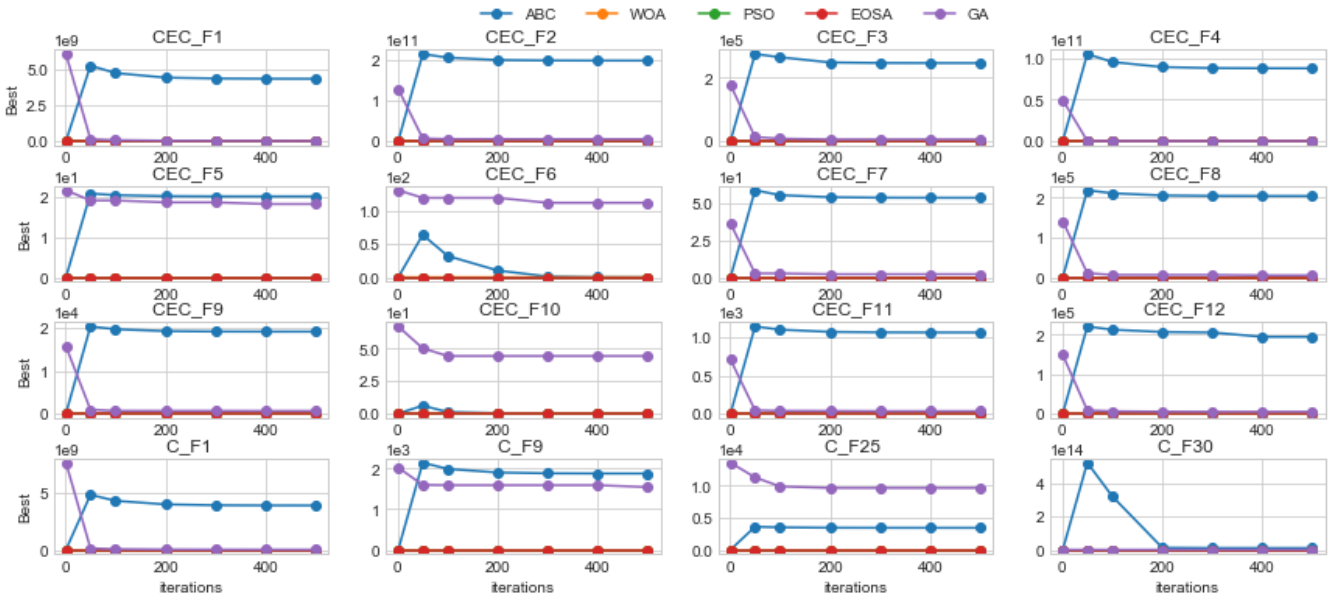


FIGURE 9. Convergent curves of EOSA and related optimization algorithms on some selected CEC functions.

not be unconnected with occasions when several algorithms experimented on the same system.

E. COMPARING THE PERFORMANCE OF EOSA WITH SIMILAR METHODS USING STATISTICAL TEST

Using the results obtained from the forty-seven (47) standard benchmark functions, we validated the performance gain of EOSA against those of ABC, WOA, BOA, PSO, DE, GA, and HGSO using statistical analysis. The Friedman mean rank test was carried out to achieve this, and the result obtained is shown in Table 12. The results showed that the proposed EOSA method ranked best above all other methods by yielding a mean rank of 1.60. The PSO, WOA, and BOA trail after it while HGSO, GA, ABC, and DE follow in that order.

The test statistics (χ^2) result for the Friedman test revealed an overall statistically significant difference between the mean ranks of the eight (8) methods, namely: EOSA, ABC,

TABLE 11. Average computational requirements of ABC, WOA, BOA PSO, QSO, DE, GA, and HGSO for forty-seven (47) benchmarks and forty-four (44) CEC for 500 runs and 100 population size.

	ABC	WOA	BOA	PSO	EOSA	DE	GA	HGSO
Benchmark functions	450.9394	357.6453	418.0029	353.3408	132.1047	441.237	365.8788	442.8186
CEC	450.9394	357.6453	418.0029	353.3408	132.1047	441.237	365.8788	442.8186

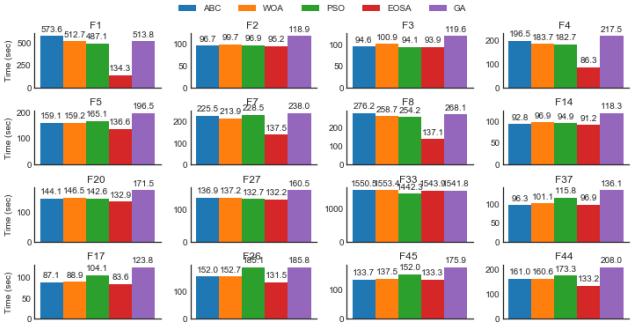


FIGURE 10. A graphical illustration of computational time required for the execution of EOSA compared with ABC, WOA, PSO, and GA for the standard benchmark functions.

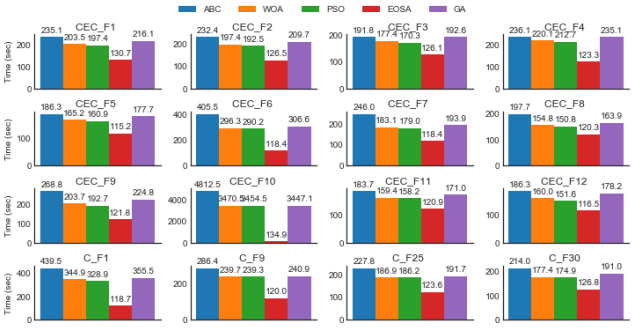


FIGURE 11. A graphical illustration of computational time required for the execution of EOSA compared with ABC, WOA, PSO, and GA for the CEC functions.

TABLE 12. Friedman mean ranks test for EOSA compared with similar optimization algorithms.

Algorithm	Mean rank
ABC	7.10
WOA	2.90
BOA	2.90
PSO	2.79
EOSA	1.60
DE	7.57
GA	6.12
HGSO	5.02

WOA, BOA, PSO, GA, DE and HGSO. The test statistics (χ^2) value of 249.847380 was obtained along with degrees of freedom (df) of 7 and significance level (Asymptotic Significance) of 0.001. We discovered a statistically significant difference in performance of the eight (8) methods compared based on the values of $\chi^2(7) = 249.847380$, $p = 0.001$. The existence of this significant difference then necessitated the need for a Wilcoxon signed-rank test. Each of the methods (optimization algorithm) was uniquely combined with EOSA to determine where the significance lay. Running the test, the results in Table 13 show the post hoc output of the Wilcoxon signed-rank tests. The post hoc analysis confirms that there was a statistically significant reduction in perceived effort in the ABC-EOSA ($Z = -5.602$, $p = 0.001$), WOA-EOSA ($Z = -3.635$, $p = 0.001$), BOA-EOSA ($Z = -4.277$, $p = 0.001$), PSO-EOSA ($Z = -3.532$, $p = 0.001$), DE-EOSA ($Z = -5.613$, $p = 0.001$), GA-EOSA ($Z = -5.64$, $p = 0.001$), and HGSO-EOSA ($Z = -5.415$, $p = 0.001$).

TABLE 13. Wilcoxon Post hoc test of EOSA with each of the selected optimization methods.

	ABC - EOSA	WOA - EOSA	BOA - EOSA	PSO - EOSA	DE - EOSA	GA - EOSA	HGSO - EOSA
Z	-5.602[b]	-3.635[b]	-4.277[b]	-3.532[b]	-5.613[b]	-5.645[b]	-5.415[b]
Asymp. Sig. (2-tailed)	.000	.000	.000	.000	.000	.000	.000
a. Wilcoxon Signed Ranks Test							
b. Based on negative ranks.							

In summary, based on the outcome of the exhaustive experimentation done in this study, EOSA has shown to be a search algorithm capable of finding better solutions in a tight competition with state-of-the-art optimization algorithms. Also, the proposed algorithm demonstrated that it can find far better solutions with fewer computational requirements when compared with ABC, WOA, BOA, PSO, GA, DE, and HGSO methods.

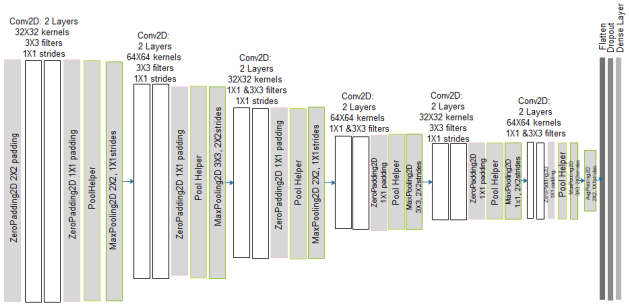


FIGURE 12. CNN model for the architecture with i1z-2c1z1s-2c1z1s-2c1z1s-2c1z1s-2c1z1s-2c1z1s-1f1drp1d.

F. APPLICATION TO MEDICAL IMAGE CLASSIFICATION PROBLEM

In this section, we present the performance of the proposed EOSA algorithm in addressing the complex problem of selecting the optimal combination of hyperparameters of convolutional neural (CNN) architecture. The resulting optimized CNN architecture is then applied to feature detection and classification of digital mammographic

TABLE 14. Performance comparison of CNN, CNN-GA, CNN-WOA, and CNN-EOSA in terms of accuracy and loss.

Epoch	CNN		CNN-GA		CNN-WOA		CNN-EOSA	
	Acc	**loss**	**Acc**	**loss**	**Acc**	**loss**	**Acc**	**loss**
1	0.55	644.72	0.85	1247.6	0.75	1249.7	0.87	1280
2	0.58	153.92	0.87	1180.7	0.77	1127.3	0.86	1156
3	0.74	362.74	0.73	1047.0	0.83	1014.9	0.88	1060
4	0.78	55.82	0.67	901.02	0.81	913.52	0.89	953
5	0.66	20.18	0.76	841.39	0.75	819.94	0.89	4894

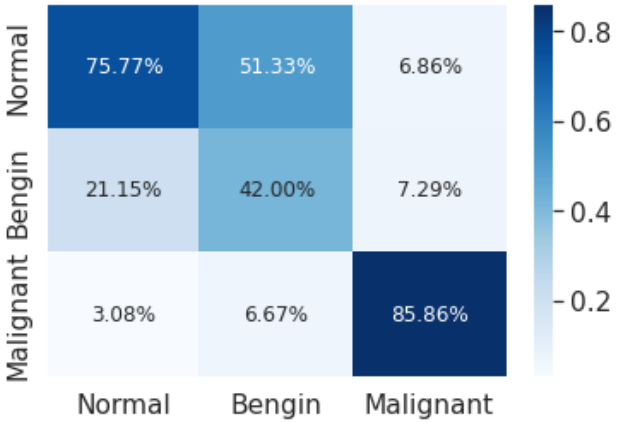
images to characterize breast cancer abnormalities. The architecture of the CNN model applied to this task is shown in Figure 12.

Table 14 shows a comparative analysis of the performance for hybridization of CNN and three optimization algorithms CNN-GA, CNN-WOA, and CNN-EOSA. We applied the GA, WOA and EOSA optimization algorithms to optimize the hyperparameters of the CNN architecture. The optimization algorithms and the resulting optimized CNN architecture were trained for five (5) epochs and compared performance. Meanwhile, the traditional CNN architecture with no optimization applied to it was also trained for the same number of epochs. Results obtained showed that the un-optimized CNN architecture showed the least accuracy compared with an optimized version of the same architecture. Comparing the accuracies of CNN-GA, CNN-WOA, and CNN-EOSA, we found that CNN-EOSA demonstrated superior performance. Also, we captured the loss values obtained for all the variations of the CNN architecture, as shown in the table listing.

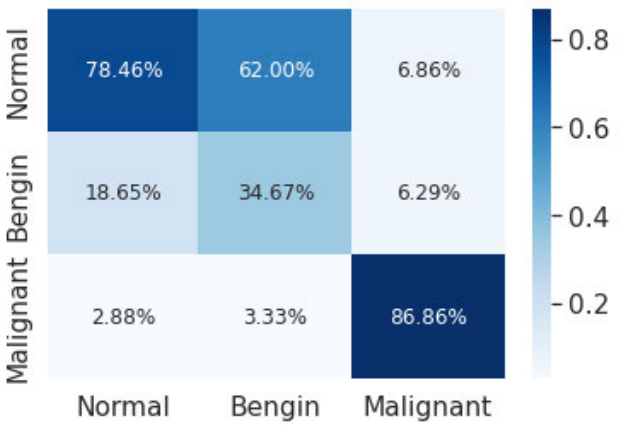
TABLE 15. Comparison of performance of CNN, CNN-GA, CNN-WOA and CNN-EOSA in terms of classification accuracy, precision, recall, and F1 score.

S/n	Algorithm	Accuracy	Precision	Recall	F1 score
1	CNN	0.86	0.73	0.86	0.79
2	CNN-GA	0.96	0.92	0.96	0.94
3	CNN-WOA	0.89	0.92	0.89	0.90
4	CNN-EOSA	0.96	0.92	0.96	0.94

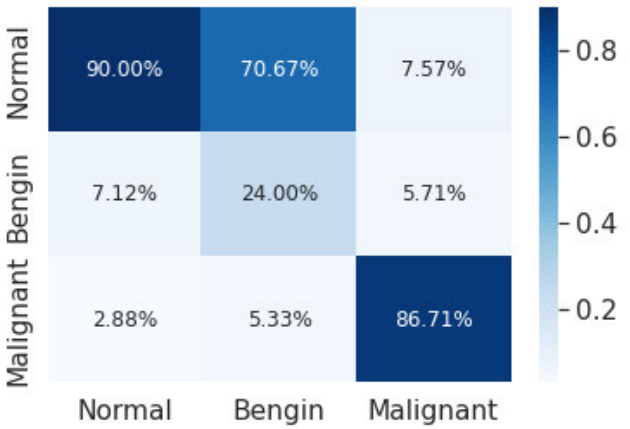
Furthermore, the trained models obtained from training CNN, CNN-GA, CNN-WOA, and CNN-EOSA were applied to test data, and performance was also compared using the metrics: accuracy, precision, recall, and F1 score. The prediction achieved using the trained model is listed in Table 15. The result again confirmed that optimizing the hyperparameters of CNN architecture using a metaheuristic algorithm is essential. However, more interesting is the performance comparison of the application of the algorithm to optimize the traditional CNN architecture. We observed that the hybrid of CNN-EOSA outperformed CNN-GA and CNN-WOA when comparison was made using the metrics mentioned earlier.



(a) CNN-GA



(b) CNN-WOA



(c) CNN-EOSA

FIGURE 13. An illustration for confusion matrix for CNN-GA, CNN-WOA and CNN_EOSA hybrid algorithms.

This further confirms the EOSA algorithm's applicability to effectively address the problem of medical image classification with particular mention to breast cancer classification from digital mammography samples.

In addition to the classification problem of breast cancer, we also investigated the performance of the

EOSA-CNN algorithm on the lung cancer dataset. The experimental results obtained are outlined in Table 16, where the accuracy, Cohens Kappa, specificity, sensitivity, precision, recall, F1-score and balanced accuracy are computed and reported. CNN-EOSA algorithm outperforms the other hybrid models and the traditional CNN in most of the metrics used for the evaluation. Again, this confirms that CNN-EOSA successfully detected the features of each class and correctly classified them in an excellent performance accuracy rating.

TABLE 16. The overall and per class performance of the CNN-GA, CNN-WOA, and CNN-EOSA hybrid algorithms and as compared with the basic CNN architecture.

Measure/ Methods	CNN-GA	CNN-WOA	CNN -EOSA	CNN
Accuracy	0.82	0.82	0.87	0.80
Cohens kappa	0.63	0.63	0.70	0.60
Precision	0.84	0.82	0.83	0.81
Recall	0.77	0.77	0.82	0.75
F1 score	0.8	0.79	0.82	0.78
Specificity	0.73	0.75	0.98	0.70
Sensitivity	0.57	0.41	0.38	0.53

For instance, the best values obtained for accuracy for CNN-GA, CNN-WOA, and CNN-EOSA are 0.81, 0.81, and 0.82, respectively. We see that in all these tests, the CNN-EOSA algorithm yielded a better performance when compared with other hybrid algorithms. In addition to the CNN-EOSA surpassing other hybrids, we also noticed that it outperformed the basic CNN architecture by an increase of 0.06.

Figure 13 shows the confusion matrix plot for all hybrid algorithms with respect to all the class labels observed in the dataset. The classification accuracy of all classes is indicated for each plot of the confusion matrix to give an accurate report on their performances. Taking the case of CNN-EOSA as an example, we see that 90% of all cases with *normal* label were correctly identified and over 86% of cases labelled as *malignant* were correctly identified by the proposed hybrid algorithm.

The performance of the EOSA metaheuristic algorithm on the standard benchmark functions, IEEE-CEC constraint function, and its application to the challenge of image classification in the detection of breast cancer are all strong, unbiased indicators of the viability of the newly proposed metaheuristics algorithm.

VI. CONCLUSION

This paper has presented a novel optimization algorithm, EOSA, based on the propagation model of the deadly Ebola virus and its associated disease. The study has shown how the bio-inspired algorithm derived its efficiency from the dynamic mechanism of moving individuals in the population through the susceptible, infected, quarantined, hospitalized, recovered, and dead sub-populations. The study presented an improved version of the propagation model of the Ebola virus disease, which was further translated into a mathematical model. The resulting model was applied to the design of the novel EOSA metaheuristic algorithm. We applied EOSA to two sets of benchmark functions consisting of forty-seven (47) classical and over thirty (30) constrained IEEE CEC-benchmark functions. The outcome of extensive experimentation to determine the algorithm's performance showed that it provides performance on a par with other population-based methods. This performance is seen as competitive with other state-of-the-art related methods from the literature. Although the EOSA metaheuristic algorithm did not show superior performance in all cases, a significant outcome confirms it is very potent in handling optimization problems.

Moreover, considering the no-free lunch theorem, we safely conclude that the optimization fits into the body of recognized and viable optimization algorithms in the literature. A more interesting outcome of the proposed algorithm is the computational demand required for its performance. The experimentation showed that the CPU time for the completion of the algorithm was substantially lower than some state-of-the-art optimization algorithms. Also, we applied the resulting algorithm to the real-world problem of classification of abnormalities in medical (mammography) images to detect breast cancer. EOSA was used to optimize the selection process of obtaining the best combination of hyperparameters of CNN architectures used for the image classification problem. Performance comparison of CNN-EOSA with other similar hybrids confirms the EOSA method's effective applicability. As future work, this study intends to investigate different strategies capable of maximizing a more excellent balance between the exploration and exploitation phase of the algorithm. Also, the constraint of the new algorithm might be overcome using a hybridization solution with other optimization algorithms, demonstrating characteristics of eliminating the constraint.

DATA AVAILABILITY

The authors confirm that the data supporting the findings of this study are available within the article.

FUNDING STATEMENT

This research received no specific grant from any public funding agency.

DECLARATION OF COMPETING INTEREST

The authors declare that they have no known competing financial interests or personal relationships that could have influenced the work reported in this paper.

ACKNOWLEDGMENT

The authors wish to state that a previous version of this paper is presented as a preprint and available online as: Oyelade,

O. N., & Ezugwu, A. E. (2021). Ebola Optimization Search Algorithm (EOSA): A new metaheuristic algorithm based on the propagation model of Ebola virus disease. arXiv preprint arXiv:2106.01416. https://arxiv.org/abs/2106.01416

REFERENCES

[1] S. Gumusova, M. Sunbul, and H. Leblebicioglu, "Ebola virus disease and the veterinary perspective," *Ann. Clin. Microbiol. Antimicrobials*, vol. 14, no. 1, p. 30, Dec. 2015.

[2] M. T. Osterholm, K. A. Moore, and N. S. Kelley, "Transmission of Ebola viruses: What we know and what we do not know," *mBio*, vol. 6, no. 2, 2015, Art. no. e00137.

[3] A. Kadanali, "An overview of ebola virus disease," *Northern Clinics Istanbul*, vol. 2, no. 1, pp. 81–86, 2016.

[4] CDC. (Jan. 14, 2021). *Ebola (Ebola Virus Disease)*. Accessed: Feb. 19, 2021. [Online]. Available: https://www.cdc.gov/vhf/ebola/transmission/index.html

[5] C. Tanade, N. Pate, E. Paljug, R. A. Hoffman, and M. D. Wang, "Hybrid modeling of Ebola propagation," in *Proc. IEEE Int. Symp. Bioinf. Bioeng.*, Oct. 2019, pp. 204–210.

[6] T. Agent. (2020). *Ebola Virus*. Accessed: Feb. 19, 2021. [Online]. Available: https://www.bcm.edu/departments/molecular-virology-and-microbiology/emerging-infections-and-biodefense/specific-agents/ebola-virus

[7] WHO. (Feb. 10, 2020). *Ebola Virus Disease*. Accessed: Feb. 19, 2021. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/ebola-virus-disease

[8] P. Marrow, "Nature-inspired computing technology and Applications," *BT Technol. J.*, vol. 18, pp. 13–23, Jan. 2000.

[9] M. Wang, H. Chen, B. Yang, X. Zhao, L. Hue, and Z. Cai, "Toward an optimal kernel extreme learning machine using a chaotic moth-flame optimization strategy with applications in medical diagnoses," *Neurocomputing*, vol. 267, pp. 69–84, Dec. 2017.

[10] N. A. H. Siddique, "Nature inspired computing: An overview and some future directions," *Cognit. Comput.*, vol. 8, vol. 7, no. 6, pp. 706–714, 2015, doi: 10.1007/s12559-015-9370-8.

[11] A. E. Ezugwu, A. K. Shukla, R. Nath, A. A. Akinyelu, J. O. Agushaka, H. Chiroma, and P. K. Muhuri, "Metaheuristics: A comprehensive overview and classification along with bibliometric analysis," *Artif. Intell. Rev.*, vol. 54, no. 6, pp. 4237–4316, Aug. 2021.

[12] A. E. Ezugwu and D. Prayogo, "Symbiotic organisms search algorithm: Theory, recent advances and applications," *Expert Syst. Appl.*, vol. 119, pp. 184–209, Apr. 2019.

[13] A. E. Ezugwu, O. J. Adeleke, A. A. Akinyelu, and S. Viriri, "A conceptual comparison of several Metaheuristic algorithms on continuous optimisation problems," *Neural Comput. Appl.*, vol. 32, no. 10, pp. 6207–6251, May 2020.

[14] M. E. C. Bento, D. Dotta, R. Kuiava, and R. A. Ramos, "A procedure to design fault-tolerant wide-area damping controllers," *IEEE Access*, vol. 6, pp. 23383–23405, 2018.

[15] J. Xiong, X. Liu, X. Zhu, H. Zhu, and H. Q. Liand Zhang, "Semi-supervised fuzzy c-means clustering optimized by simulated annealing and genetic algorithm for fault diagnosis of bearings," *IEEE Access*, vol. 8, pp. 181976–181987, 2020.

[16] D. I. Arkhipov, D. Wu, T. Wu, and A. C. Regan, "A parallel genetic algorithm framework for transportation planning and logistics management," *IEEE Access*, vol. 8, pp. 106506–106515, 2020.

[17] S. Sennan, S. Ramasubbareddy, S. Balasubramaniyam, A. Nayyar, M. Abouhawwash, and N. Hikal, "T2FL-PSO: Type-2 fuzzy logic-based particle swarm optimization algorithm used to maximize the lifetime of Internet of Things," *IEEE Access*, vol. 9, pp. 63966–63979, 2020.

[18] A. E. Ezugwu, "Advanced discrete firefly algorithm with adaptive mutation-based neighborhood search for scheduling unrelated parallel machines with sequence-dependent setup times," *Int. J. Intell. Syst.*, early access, pp. 1–42, Nov. 2021, doi: 10.1002/int.22733.

[19] A. E. Ezugwu, V. Pillay, D. Hirasen, K. Sivanarain, and M. Govender, "A comparative study of meta-heuristic optimization algorithms for 0–1 knapsack problem: Some initial results," *IEEE Access*, vol. 7, pp. 43979–44001, 2019.

[20] A. E. Ovre and J. Agushaka, "Influence of initializing Krill Herd algorithm with low-discrepancy sequences," *IEEE Access*, vol. 4, pp. 210886–210909, 2020.

[21] A. Darwish, "Bio-inspired computing: Algorithms review, deep analysis, and the scope of applications," *Future Comput. Informat. J.*, vol. 3, no. 2, pp. 231–246, Dec. 2018.

[22] S. Sivanandam and S. Deepa, *Introduction to Genetic Algorithms*. Berlin, Germany: Springer, 2008.

[23] S. Salcedo-Sanz, J. Del Ser, I. Landa-Torres, S. Gil-López, and J. A. Portilla-Figueras, "The coral reefs optimization algorithm: A novel Metaheuristic for efficiently solving optimization problems," *Sci. World J.*, vol. 2014, pp. 1–15, Jan. 2014.

[24] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," Dept. Comput. Eng., Erciyes Univ., Kayseri, Turkey, Tech. Rep.-tr06, 2005.

[25] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Proc. 6th Int. Symp. Micro Mach. Hum. Sci.*, 1955, pp. 39–43.

[26] S. Arora and S. Singh, "Butterfly optimization algorithm: A novel approach for global optimization," *Soft Comput.*, vol. 23, no. 3, pp. 715–734, 2019.

[27] S. C. Chu, P. W. Tsai, and J. S. Pan, "Cat swarm optimization," in *Proc. Pacific Rim Int. Conf. Artif. Intell.* Berlin, Germany: Springer, 2006, pp. 854–858.

[28] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, Jan. 2014.

[29] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Adv. Eng. Softw.*, vol. 95, pp. 51–67, Jun. 2016.

[30] M. Mahmood and B. Al-Khateeb, "The blue monkey: A new nature inspired metaheuristic optimization algorithm," *Periodicals Eng. Natural Sci.*, vol. 7, no. 3, pp. 1054–1066, 2019.

[31] L. Abualigah, A. Diabat, S. Mirjalili, M. Abd Elaziz, and A. H. Gandomi, "The arithmetic optimization algorithm," *Comput. Methods Appl. Mech. Eng.*, vol. 376, Apr. 2021, Art. no. 113609.

[32] J. O. Agushaka and A. E. Ezugwu, "Evaluation of several initialization methods on arithmetic optimization algorithm performance," *J. Intell. Syst.*, vol. 31, no. 1, pp. 70–94, Dec. 2021.

[33] J. O. Agushaka and A. E. Ezugwu, "Advanced arithmetic optimization algorithm for solving mechanical engineering design problems," *PLoS ONE*, vol. 16, no. 8, Aug. 2021, Art. no. e0255703.

[34] L. Abualigah, D. Yousri, M. Abd Elaziz, A. A. Ewees, M. A. Al-qaness, and A. H. Gandomi, "Aquila optimizer: A novel meta-heuristic optimization algorithm," *Comput. Ind. Eng.*, vol. 157, Jul. 2021, Art. no. 107250.

[35] L. Abualigaha, M. A. Elaziz, P. Sumari, Z. WooGeem, and A. H. Gandomi, "Reptile Search Algorithm (RSA): A nature-inspired meta-heuristic optimizer," *Expert Syst. Appl.*, vol. 1, Apr. 2022, Art. no. 116158.

[36] A. Kaur, S. Jain, and S. Goel, "Sandpiper optimization algorithm: A novel approach for solving real-life engineering problems," *Int. J. Speech Technol.*, vol. 50, no. 2, pp. 582–619, Feb. 2020.

[37] P. Game, V. Vaze, and M. Emmanuel, "Bio-inspired optimization: Meta-heuristic algorithms for optimization," in *Nat. Conf. Emerg. Trends, Challenges Opportunities Data Mining Inf. Secur.*, 2020, pp. 1–5.

[38] M. Crepinsek, S.-H. Liu, and M. Mernik, "Exploration and exploitation in evolutionary algorithms: A survey," *ACM Comput. Surv.*, vol. 45, no. 3, pp. 1–33, Jul. 2012.

[39] E. Cuevas, A. Echavarría, and M. A. Ramírez-Ortegón, "An optimization algorithm inspired by the states of matter that improves the balance between exploration and exploitation," *Appl. Intell.*, vol. 40, no. 2, pp. 256–272, 2014.

[40] O. N. Oyelade and A. E. Ezugwu, "Ebola Optimization Search Algorithm (EOSA): A new metaheuristic algorithm based on the propagation model of Ebola virus disease," Cornell Univ., New York, NY, USA, Tech. Rep., 2021. [Online]. Available: https://arxiv.org/abs/2106.01416

[41] M. Jamil and X.-S. Yang, "A literature survey of benchmark functions for," *Int. J. Math. Model. Numer. Optim.*, vol. 4, no. 2, pp. 150–194, 2013.

[42] R. Cheng, M. Li, Y. Tian, X. Zhang, S. Yang, Y. Jin, and X. Yao, "Benchmark functions for the CEC'2018 competition on many-objective optimization," Univ. Birmingham, Birmingham, U.K., Tech. Rep. TR-CEC2018-MaOO-Competition, 2018. [Online]. Available: http://hdl.handle.net/2086/15157

[43] S. Rewar and D. Mirdha, "Transmission of Ebola virus disease: An overview," *Ann. Global Health*, vol. 80, no. 6, pp. 444–451, 2014.

[44] L. M. Mobula, N. MacDermott, C. Hoggart, K. Brantly, W. Plyler, J. Brown, B. Kauffeldt, D. Eisenhut, L. A. Cooper, and J. Fankhauser, "Clinical manifestations and modes of death among patients with ebola virus disease, monrovia, liberia, 2014," *Amer. J. Tropical Med. Hygiene*, vol. 98, no. 4, pp. 1186–1193, Apr. 2018.

[45] T. Berge, J. M.-S. Lubuma, G. M. Moremedi, N. Morris, and R. Kondera-Shava, "A simple mathematical model for ebola in Africa," *J. Biol. Dyn.*, vol. 11, no. 1, pp. 42–74, Jan. 2017.

[46] N. N. Yet, "Modeling Ebola virus infection," *Model. Control Infectious Diseases Host, MATLAB R.*, vol. 6, pp. 85–103, Oct. 2019, doi: 10.1016/B978-0-12-813052-0.00016-6.

[47] A. Rachah and D. F. M. Torres, "Mathematical modelling, simulation, and optimal control of the 2014 Ebola outbreak in West Africa," *Discerete Dyn. Nature Soc.*, vol. 2015, Feb. 2015, Art. no. 842792.

[48] E. Okyere, J. D.-G. Ankamah, A. K. Hunkpe, and D. Mensah, "Deterministic epidemic models for Ebola infection with time-dependent controls," *Discerete Dyn. Nature Soc.*, vol. 2020, Jul. 2020, Art. no. 2823816.

[49] P. M. Oliveira, E. S. Pires, J. Boaventura-Cunha, and T. Martins, "Review of nature and biologically inspired metaheuristics for greenhouse environment control," *Trans. Inst. Meas. Control*, vol. 42, no. 12, pp. 2338–2358, 2020, doi: 10.1177/0142331220909010.

[50] J. Fan, Q. Hu, and Z. Tang, "Predicting vacant parking space availability: An SVR method with fruit fly optimisation," *IET Intell. Transport. Syst.*, vol. 12, no. 10, pp. 1414–1420, Dec. 2018.

[51] M. Zhou, F. Lin, Q. Hu, Z. Tang, and C. Jin, "AI-enabled diagnosis of spontaneous rupture of ovarian endometriomas: A PSO enhanced random forest approach," *IEEE Access*, vol. 10, pp. 132253–132264, 2020.

[52] P. Liu, Q. Hu, K. Jin, and Z. Tang, "Toward the energy-saving optimization of WLAN deployment in real 3-D environment: A hybrid swarm intelligent method," *IEEE Syst. J.*, early access, Apr. 5, 2021, doi: 10.1109/JSYST.2021.3065434.

[53] O. N. Oyelade and A. E. Ezugwu, "A bioinspired neural architecture search based convolutional neural network for breast cancer detection using histopathology images," *Sci. Rep.*, vol. 11, Oct. 2021, Art. no. 19940.

[54] T. Achary, S. Pillay, S. M. Pillai, M. Mqadi, E. Genders, and A. E. Ezugwu, "A performance study of meta-heuristic approaches for quadratic assignment problem," *Concurrency Comput., Pract. Exper.*, vol. 33, no. 17, pp. 1–29, Sep. 2021.

[55] C. Munien and A. E. Ezugwu, "Metaheuristic algorithms for one-dimensional bin-packing problems: A survey of recent advances and applications," *J. Intell. Syst.*, vol. 30, no. 1, pp. 636–663, Apr. 2021.

[56] C. Munien, S. Mahabeer, E. Dzitiro, S. Singh, S. Zungu, and A. E.-S. Ezugwu, "Metaheuristic approaches for one-dimensional bin packing problem: A comparative performance study," *IEEE Access*, vol. 8, pp. 227438–227465, 2020.

[57] A. E. Ezugwu, "Enhanced symbiotic organisms search algorithm for unrelated parallel machines manufacturing scheduling with setup times," *Knowl. Based Syst.*, vol. 172, no. 15, pp. 15–32, May 2019.

[58] G. Dhiman, "ESA: A hybrid bio-inspired metaheuristic optimization approach for engineering problems," *Eng. Comput.*, vol. 37, no. 3, pp. 323–353, 2019.

[59] F. Martínez-Álvarez, G. Asencio-Cortés, J. F. Torres, D. Gutiérrez-Avilés, L. Melgar-García, R. Pérez-Chacón, C. Rubio-Escudero, J. C. Riquelme, and A. Troncoso, "Coronavirus optimization algorithm: A bioinspired Metaheuristic based on the COVID-19 propagation model," *Big Data*, vol. 8, no. 4, pp. 308–322, Aug. 2020, doi: 10.1089/big.2020.0051.

[60] M. Al-Betar, Z. Alyasseri, M. Awadallah, and I. A. Doush, "Coronavirus herd immunity optimizer (CHIO)," *Neural Comput. Appl.*, vol. 33, pp. 5011–5042, Aug. 2020.

[61] G. G. Wang, S. Deb, and L. D. S. Coelho, "Earthworm optimisation algorithm: A bio-inspired Metaheuristic algorithm for global optimisation problems," *Int. J. Bio-Inspired Comput.*, vol. 12, no. 1, p. 1, 2018.

[62] D. Simon, "Biogeography-based optimization," *IEEE Trans. Evol. Comput.*, vol. 12, no. 6, pp. 702–713, Dec. 2008, doi: 10.1109/TEVC.2008.919004.

[63] A. R. Mehrabian and C. Lucas, "A novel numerical optimization algorithm inspired from weed colonization," *Ecol. Inform.*, vol. 1, no. 4, pp. 355–366, Dec. 2006.

[64] S. Zhang, Y. Zhou, and Q. A. Luo, "A complex-valued encoding satin bowerbird optimization algorithm for global optimization," *Evol. Syst.*, vol. 12, no. 1, pp. 191–205, 2019.

[65] D. G. B. Amali and M. Dinakaran, "Wildebeest herd optimization: A new global optimization algorithm inspired by wildebeest herding behaviour," *J. Intell. Fuzzy Syst.*, vol. 37, no. 6, pp. 8063–8076, Dec. 2019.

[66] K. P. N. Jayasena, L. Li, M. Abd Elaziz, and S. Xiong, "Multi-objective energy efficient resource allocation using virus colony search (VCS) algorithm," in *Proc. IEEE 20th Int. Conf. High Perform. Comput. Communications; IEEE 16th Int. Conf. Smart City; IEEE 4th Int. Conf. Data Sci. Syst. (HPCC/SmartCity/DSS)*, Jun. 2018, pp. 766–773.

[67] S. Li, H. Chen, M. Wang, A. A. Heidari, and S. Mirjalili, "Slime mould algorithm: A new method for stochastic optimization," *Future Gener. Comput. Syst.*, vol. 111, pp. 300–323, Oct. 2020.

[68] S. M. Moghadas, T. N. Vilches, K. Zhang, and C. R. Wells, "The impact of vaccination on COVID-19 outbreaks in the United States," Tech. Rep., 2020, doi: 10.1101/2020.11.27.20240051.

[69] BBC. (Feb. 3, 2021). *COVID-19: Study Showing Oxford Vaccine Slows Virus Spread 'Superb'- Hancock*. Accessed: Feb. 24, 2021. [Online]. Available: https://www.bbc.com/news/U.K.-55913913

[70] L. Mathebula, D. E. Ndwandwe, E. Pienaar, and C. S. Wiysonge, "Effects of vaccines in protecting against Ebola virus disease: Protocol for a systematic review," *BMJ Open*, vol. 9, no. 7, 2019, Art. no. e029617.

[71] R. Potluri, A. Kumar, V. Maheshwari, C. Smith, V. Oriol Mathieu, K. Luhn, B. Callendret, and H. Bhandari, "Impact of prophylactic vaccination strategies on ebola virus transmission: A modeling analysis," *PLoS ONE*, vol. 15, no. 4, Apr. 2020, Art. no. e0230406, doi: 10.1371/journal.pone.0230406.

[72] WHO. (Jan. 16, 2016). *Interim Advice on the Sexual Transmission of the Ebola Virus Disease*. Accessed: Feb. 24, 2021. [Online]. Available: https://www.who.int/reproductivehealth/topics/rtis/ebola-virus-semen/en/

[73] UNCHC. (Aug. 2, 2017). *Ebola Detected in Semen of Survivors Two Years After Infection: Researchers Suggest Updating WHO Guidelines and Exploring Aging*. Accessed: Feb. 24, 2021. [Online]. Available: https://www.sciencedaily.com/releases/2017/08/170802152532.htm

[74] A. E. Thorson, "Persistence of ebola virus in semen among ebola virus disease survivors in sierra leone: A cohort study of frequency, duration, and risk factors," *PLOS Med.*, vol. 18, no. 2, Feb. 2021, Art. no. e1003273, doi: 10.1371/journal.pmed.1003273.

[75] X.-S. Yang, "Firefly algorithm, stochastic test functions and design optimisation," 2010, *arXiv:1003.1409*.

[76] M. Molga and C. Smutnicki. (2020). *Test Functions for Optimization Needs*. [Online]. Available: https://www.robertmarks.org/Classes/ENGR5358/Papers/functions.pdf

[77] J. G. Digalakis and K. G. Margaritis, "On benchmarking functions for genetic algorithms," *Int. J. Comput. Math.*, vol. 77, no. 4, pp. 481–506, Jan. 2001.

[78] K. Hussain, M. N. B. Salleh, S. Cheng, and R. Naseem, "Common benchmark functions for metaheuristic evaluation: A review," *Int. J. Inf. Vis.*, vol. 4, pp. 218–223, Oct. 2017.

[79] N. H. Awad, M. Z. Ali, J. Liang, B. Qu, and P. N. Suganthan, "Problem definitions and evaluation criteria for the CEC 2017 special session and competition on single objective real parameter numerical optimization," Nanyang Technol. Univ., Singap., Tech. Rep. 61780053, 2016.

[80] A. Sau, "A simulation study on hypothetical Ebola virus transmission in India using spatiotemporal epidemiological modeler (STEM): A way towards precision public health," *J. Environ. Public Health*, vol. 2017, Feb. 2017, Art. no. 7602301.

OLAIDE NATHANIEL OYELADE is currently a Postdoctoral Fellow with the School of Mathematics, Statistics, and Computer Science, University of KwaZulu-Natal, Durban, South Africa. His main research interests include knowledge representation and reasoning algorithm design, deep learning, and artificial intelligence (AI), with a specific interest in computational intelligence.

ABSALOM EL-SHAMIR EZUGWU received the B.Sc. degree in mathematics with computer science and the M.Sc. and Ph.D. degrees in computer science from Ahmadu Bello University, Zaria, Nigeria. He is currently a Senior Lecturer with the School of Mathematics, Statistics, and Computer Science, University of KwaZulu-Natal, Durban, South Africa. He has published articles relevant to his research interest in internationally refereed journals and edited books, conference proceedings, and local journals. His main research interests include parallel algorithms design in the cloud and grid computing environments, artificial intelligence with a specific interest in computational intelligence, and metaheuristic solutions to real-world global optimization problems. He is a member of IAENG and ORSSA.

TEHNAN I. A. MOHAMED received the B.Sc. degree (Hons.) in computer science from the University of Gezira, Sudan. She is currently pursuing the master's degree in computer science with the University of Kwazulu-Natal, South Africa. Her research interests include applied artificial intelligence, specifically in using machine learning models for medical image analysis.

LAITH ABUALIGAH received the first degree in computer information system and the master's degree in computer science from Al-Albayt University, Jordan, in 2011 and 2014, respectively, and the Ph.D. degree from the School of Computer Science, Universiti Sains Malaysia (USM), Malaysia, in 2018. He is currently an Assistant Professor with the Computer Science Department, Amman Arab University, Jordan. He is also a Distinguished Researcher at the School of Computer Science, USM. According to the report published by Stanford University, in 2020, he is one of the 2% influential scholars, which depicts the 100,000 top scientists in the world. He has published more than 80 journal articles and books, which collectively have been cited more than 3100 times (H-index = 27). His main research interests include arithmetic optimization algorithm (AOA), bio-inspired computing, nature-inspired computing, swarm intelligence, artificial intelligence, meta-heuristic modeling, optimization algorithms, evolutionary computations, information retrieval, text clustering, feature selection, combinatorial problems, optimization, advanced machine learning, big data, and natural language processing. He serves as an Associate Editor for the *Cluster Computing* (Springer) and the *Soft Computing* (Springer).