IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 16, 2022, accepted March 3, 2022, date of publication March 10, 2022, date of current version March 21, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3158357

Philippine Eagle Optimization Algorithm

ERIKA ANTONETTE T. ENRIQUEZ[1,2], **RENIER G. MENDOZA**[1], **AND ARRIANNE CRYSTAL T. VELASCO**[1,2]

[1]Institute of Mathematics, University of the Philippines Diliman, Quezon City 1101, Philippines
[2]Natural Sciences Research Institute, University of the Philippines Diliman, Quezon City 1101, Philippines

Corresponding author: Arrianne Crystal T. Velasco (acvelasco@math.upd.edu.ph)

This work was supported by the Natural Sciences Research Institute, University of the Philippines Diliman, under Research Grant MAT-22-1-01.

ABSTRACT We propose the Philippine Eagle Optimization Algorithm (PEOA), which is a meta-heuristic and population-based search algorithm inspired by the territorial hunting behavior of the Philippine Eagle. From an initial random population of eagles in a given search space, the best eagle is selected and undergoes a local food search using the interior point method as its means of exploitation. The population is then divided into three subpopulations, and each subpopulation is assigned an operator which aids in the exploration. Once the respective operators are applied, the new eagles with improved function values replace the older ones. The best eagle of the population is then updated and conducts a local food search again. These steps are done iteratively, and the food searched by the final best eagle is the optimal solution of the search space. PEOA is tested on 20 optimization test functions with different modality, separability, and dimension properties. The performance of PEOA is compared to 13 other optimization algorithms. To further validate the effectiveness of PEOA, it is also applied to image reconstruction in electrical impedance tomography and parameter identification in a neutral delay differential equation model. Numerical results show that PEOA can obtain accurate solutions to various functions and problems. PEOA proves to be the most computationally inexpensive algorithm relative to the others examined, while also helping promote the critically endangered Philippine Eagle.

INDEX TERMS Electrical impedance tomography, metaheuristic search, nature-inspired algorithm, neutral delay differential equation, numerical optimization, Philippine Eagle.

I. INTRODUCTION

A. METAHEURISTIC ALGORITHMS

Numerical optimization is the study of finding solutions using mathematical tools to achieve objectives in the most efficient way [1]. Finding solutions to optimization problems is usually very challenging, so various algorithms have been created to tackle different kinds of problems.

In particular, metaheuristic search algorithms have been used because of their trial-and-error approach in finding solutions, which have many advantages over traditional and purely deterministic methods [1], [2]. These advantages can be seen when dealing with functions that have some discontinuity, design optimization problems that have highly nonlinear functions or constraints, or stochastic problems where uncertainty and noise exist [1]–[3]. In these cases, techniques using a trade-off between randomization and local search, such as metaheuristic algorithms, are preferred [4].

A state-of-the-art metaheuristic algorithm is the Genetic Algorithm (GA) [5], which is based on Darwinian evolution and natural selection of biological systems. The problem-solving strategy of GA is to use genetic operators, namely crossover and recombination, mutation, and selection.

One further development to GA is the Differential Evolution (DE) [6], which is a vector-based, derivative-free evolutionary algorithm. Unlike GA, DE treats solutions as real-number strings, and operations are carried out over each component of the solution vectors.

More improved variants of these algorithms have also been developed recently, such as those that use adaptive parameter control, an external archive, and combinations of multiple operators and methods.

For example, the Improved Multi-Operator Differential Evolution (IMODE) [17] has been proposed, which uses

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

TABLE 1. Summary of some nature-inspired optimization algorithms, including their inspiration source from nature, algorithmic key features, and the year when they were proposed.

Nature-Inspired Algorithm	Inspiration Source	Key Features	Year
Firefly Algorithm [7]	flashing patterns and behavior of tropical fireflies	A given firefly will be attracted to other fireflies based on brightness, which can simply be proportional to the objective function value. Attractiveness and brightness both decrease as the distance between fireflies increases. For any two given fireflies, the less bright one will move towards the brighter one.	2007
Cuckoo Search Algorithm [8]	brood parasitism of cuckoo species	Cuckoos are obtained randomly via Levy flights, and each cuckoo lays an egg in a randomly chosen host nest. The host bird can discover the egg laid by a cuckoo under a certain probability. In this case, the host bird can either get rid of the egg laid by the cuckoo or simply abandon the nest and build a completely new nest.	2009
Bat Algorithm [9]	echolocation behavior of microbats	Bats use echolocation to sense distance. They fly randomly with a certain velocity and at some location per iteration. They can automatically tune the frequency or wavelength of their emitted pulses and adjust the pulse emission rate depending on their target's proximity. Loudness varies from a large positive number to a minimum value.	2010
Flower Pollination Algorithm [10]	flower pollination process of flowering plants	Biotic and cross-pollination are parts of the global pollination process, where pollen-carrying pollinators move via Levy flights. For local pollination, abiotic pollination and self-pollination are used. Pollinators can develop flower constancy or reproduction probability. A switch probability controls the process of local and global pollination.	2012
Moth Flame Optimization Algorithm [11]	navigation method of moths called transverse orientation	Moths fly towards a flame. When the light source is near, moths fly and spiral closer to the flames. When the light source is far away, moths may fly in a straight line over a long distance while maintaining a fixed angle with the moon. A swarm of moths and a group of flames are fixed.	2015
Whale Optimization Algorithm [12]	bubble-net attacking mechanism of humpback whales	Whales use a bubble-net feeding method as part of their foraging behavior. The essential mechanisms of this method are the shrinking encircling mechanism and the spiral updating position, which are performed randomly with a 50% probability. For the exploration mechanism, whales search for prey randomly.	2016
Butterfly Optimization Algorithm [13]	foraging strategy and mating behavior of butterflies	Butterflies emit fragrances that enable them to be attracted to each other. They utilize their sense of smell to determine the location of a mating partner. Each butterfly can move randomly or fly towards the butterfly that emits the most fragrance. The stimulus intensity of a butterfly is affected by the landscape of the objective function.	2019
Wingsuit Flying Search [14]	intention of wingsuit fliers to land at the lowest possible point of the Earth's surface	The flier sees a rough picture of the Earth's surface when she/he starts flying. The flier's view toward different areas corresponds to a population of points in the search space. Though the flier cannot see where the minimum is, the flier can roughly see the valleys, so large portions of terrain can be immediately eliminated.	2020
Tunicate Swarm Algorithm [15]	jet-propulsion and swarm behaviors of tunicates during navigation and foraging	Two behaviors of tunicates are employed to find the food source, namely jet propulsion and swarm intelligence. A tunicate should avoid conflicts between search agents, move towards the best search agent, and remain close to this best search agent. Swarm behavior updates the positions of other search agents based on the best optimal solution.	2020
Chimp Optimization Algorithm [16]	individual intelligence and sexual motivation of chimps in their group hunting	Four types of chimps entitled attacker, barrier, chaser, and driver are employed for simulating chimps' diverse intelligence. The driver, barrier, and chaser undertake the responsibility of exploration whereas the attackers have the leadership of exploitation. Four main steps of hunting, i.e. driving, chasing, blocking, and attacking, are implemented.	2020
Improved Multi-Operator Differential Evolution [17]	benefits of multiple differential evolution operators	The initial population is divided into several subpopulations, each of which is evolved using its own DE variant. Subpopulation sizes are updated based on the quality of solutions and the diversity of each subpopulation. At the end of every generation, all subpopulations are gathered and redivided based on new subpopulation sizes.	2020

multiple DE operators, with more emphasis placed on the best-performing operator. IMODE also uses adaptation mechanisms to determine parameter values and randomly chooses between binomial and exponential crossover. IMODE has proven successful as an optimization algorithm, especially since it ranked first in the CEC 2020 Competition on Single Objective Bound Constrained Numerical Optimization.

Many other metaheuristic algorithms have been developed, not only because of their capability of solving optimization problems, but also due to their wide range of recent real-world applications [18]–[40].

Two essential components of metaheuristic algorithms are exploitation and exploration. Exploitation focuses the search in a local region, whereas exploration expands the search on a global scale [1], [2]. A proper balance between these two components is crucial for the overall efficiency of metaheuristic algorithms.

B. NATURE-INSPIRED ALGORITHMS

Metaheuristic algorithms are mostly nature-inspired, deriving from the beauty and order that natural elements possess [4]. For instance, animals and plants naturally develop strategies to ensure their survival through time. The abundance and success of these strategies have led to the creation of many nature-inspired metaheuristics [41]. Specifically, flying movements, foraging behavior, and hunting techniques of animals are some of the inspirations of nature-inspired metaheuristics [42].

Another aspect of nature that has also been a basis for many algorithms is swarm intelligence, which concerns the behavior of self-organizing systems, the members of which evolve and interact to achieve optimality [43]. Thus, many algorithms are also swam-intelligence-based, such as the Particle Swarm Optimization (PSO) [44].

The main inspiration of PSO is the flocking behavior of birds. In PSO, each particle in a given swarm represents a candidate solution to the optimization problem. Each particle is then updated based on its own local best position and the position of the global best particle.

More recent SI-based algorithms have further been developed, including the Firefly Algorithm, which is based on the flashing patterns and behavior of tropic fireflies [7], and the Cuckoo Search Algorithm, which is inspired by the brood parasitism of cuckoo species [8], [45]. Additionally, we have the Bat Algorithm, which is derived from the echolocation behavior of microbats [9], and the Flower Pollination Algorithm, which is based on the flower pollination process of flowering plants [10].

Even more nature-inspired algorithms have been created over recent years, such as Moth Flame Optimization Algorithm [11], Whale Optimization Algorithm [12], Butterfly Optimization Algorithm [13], Chimp Optimization Algorithm [16], Wingsuit Flying Search [14], and Tunicate Swarm Algorithm [15].

Table 1 presents a summary of the nature-inspired algorithms mentioned above, along with their inspiration sources, key features, and year.

With the increasing number of nature-inspired algorithms, various benchmarking tests have been developed to examine their performance [46]. These include testing the algorithms on different types of functions [47], [48], and checking the number of objective function evaluations they use [49].

The No-Free-Lunch Theorem for Optimization states that if algorithm A performs better than algorithm B for some optimization functions, then B will outperform A for other functions [1], [41]. In other words, there is no metaheuristic best suited for all existing optimization problems. Given this, the research area on metaheuristic algorithms is active and steadily progressing. New metaheuristics and nature-inspired algorithms are constantly being studied to determine what specific types of optimization problems these algorithms could solve the best [41], [42].

C. PHILIPPINE EAGLE (PITHECOPHAGA JEFFERYI)

In this study, we develop an optimization algorithm based on the hunting behavior of the Philippine Eagle (Pithecophaga jefferyi), the national bird of the Philippines, shown in Figure 1.

FIGURE 1. A photo of the Philippine Eagle (*Pithecophaga jefferyi*). This is owned by Sinisa Djordje Majetic and was obtained from [50]. This was confirmed to be licensed under the terms of the cc-by-sa-2.0.

Tagged as the "Haribon" or bird king, the Philippine Eagle is among the rarest and most powerful birds in the world whose species is endemic only to four islands of the Philippine archipelago, namely Luzon, Samar, Leyte, and Mindanao [51]. It is commonly known as the Monkey-Eating Eagle, but it can also prey on other vertebrates apart from monkeys, including mammals, reptiles, and other birds [52].

Unfortunately, it is now classified as critically endangered as it is continually being threatened by hunting and loss of habitat [53].

According to [54], the hunting behavior of the Philippine Eagle follows a three-part sequence, where it first perches and calls as a preparatory stage, then does the capture of prey by dropping from its perch, and finally circles back up to return to its starting point. Figure 2 shows a representation of this three-part hunting sequence.

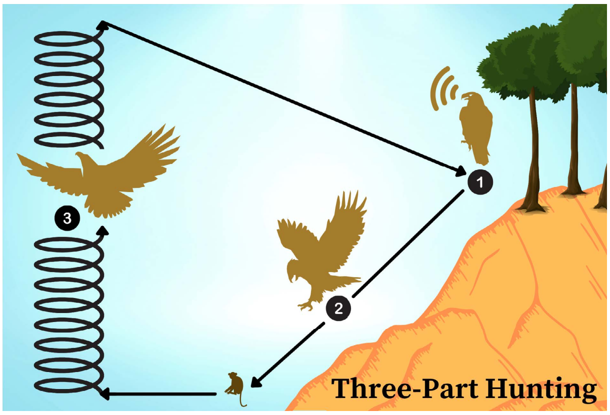


FIGURE 2. Representation of the three-part hunting sequence of the Philippine Eagle [54]. Part 1 is the preparatory period where it perches and calls, Part 2 is the act of hunting by dropping from its perch, and Part 3 is its return to its starting point in a circular upwards motion. This is designed using Canva (https://www.canva.com).

It can thus be observed that Philippine Eagles are highly territorial during hunting, besides also being known to be loyal to their nest sites [51].

Furthermore, the Philippine Eagle can hunt both singly and in pairs [54], but they generally make a more successful hunt when done in pairs. A particular strategy is for one eagle to distract the prey while the other captures this prey from behind. It is additionally noted that a bulk of the Philippine Eagles' time is spent at perch because it is from perch that they watch their surroundings and look out for prey.

Meanwhile, the Philippine Eagle's flight behavior generally follows differing patterns, where they either glide in a straight line from a higher to a lower elevation or make a sequence of short glides and large sweeping circles [54], [55].

D. INSPIRATION SOURCES AND LIMITATIONS

The national bird of the Philippines, the Philippine Eagle, has particular hunting, flying, and foraging behaviors, which had thus inspired the proposed Philippine Eagle Optimization Algorithm (PEOA). The main characteristics of the Philippine Eagle that we incorporate into PEOA are the following:

- It is a highly territorial bird when hunting and is loyal to its nest site.

- Its pair hunt strategy is more successful than hunting alone.
- It has differing flight patterns, varying between straight glides and large circles.
- It watches its surroundings and looks out for prey at perch.

The pair-hunt strategy, differing flight patterns, and perching behavior of the Philippine Eagle are the sources of inspiration for the three global operators of PEOA.

On the other hand, its territorial hunting behavior is modeled using the intensive local search of the algorithm, such that the best eagle searches for food only within its local territory.

The adaptive reduction of the population size within PEOA is likewise due to the territorial behavior of the Philippine Eagle, in the sense that eagles fight for their survival in the given region for every passing generation. Thus, the defeated eagles would just fly out of the domain and live elsewhere, reducing the population of eagles that stay in the region.

We clarify that PEOA was conceptualized out of inspiration from the Philippine Eagle, but we do not intend to attribute the whole process of the algorithm solely to this inspiration. Several nature-inspired algorithms in the literature only derive from selected characteristics of their source of inspiration [41], [42].

Furthermore, besides finding direct relationships between the Philippine Eagle and our proposed algorithm, we also seek to strengthen the algorithmic design of PEOA so it could perform efficiently on different kinds of optimization problems. This way, PEOA could be comparable with recent algorithms and can be tested on specific applications.

E. MOTIVATION OF PEOA

The motivation of the paper is to create a multi-operator algorithm with a fast and intensive local search, ultimately inspired by the Philippine Eagle.

In the algorithmic sense, the goal is to propose an algorithm with stronger exploration and exploitation techniques. The use of three different global operators, each one assigned to adaptive subpopulations based on the progress of the algorithm, is an effective global search strategy. Combining this with the consistent local search of the best solutions via the interior point method results in fast yet accurate convergence. We thus intend to create a more effective algorithm that can compete with those that are currently existing.

We also aim to associate specific characteristics of the Philippine Eagle to important aspects of an optimization algorithm (e.g. exploration and exploitation). We hope that highlighting the special traits of the Philippine Eagle can help promote the conservation of this critically-endangered national bird.

F. CONTRIBUTION HIGHLIGHTS

We propose the Philippine Eagle Optimization Algorithm (PEOA), a novel, metaheuristic, nature-inspired, and SI-based optimization algorithm inspired by the distinctive characteristics of the Philippine Eagle.

PEOA has three different global operators: the Movement Operator, the Mutation I Operator, and the Mutation II Operator. The features of each operator are the following:

- The Movement Operator considers eagle proximity, wherein eagles close to each other swarm around the same local solutions. One of these local solutions is possibly the global solution.
- The Mutation I Operator uses the concept of Lévy flights, which helps in the search within unknown, large-scale spaces.
- The Mutation II Operator determines the overall picture of the search performance by considering the current mean location of all the eagles.

These three operators are added to contribute to the exploration mechanism of PEOA. They make PEOA more competitive not only against classical algorithms but also with other modern algorithms.

PEOA conducts an intensive local search in each iteration. In particular, food search is done regularly in a specific territory of the best eagle, that is, the eagle with the least function value in a minimization problem. The interior-point method, a deterministic algorithm, is used here. This helps the exploitation capacity of PEOA.

PEOA uses an adaptive reduction of population size, that is, the population size of eagles linearly reduces depending on the current number of function evaluations. This complements both the exploration and exploitation techniques of PEOA. With more eagles at the beginning of the process, the three operators guide the eagles in exploring the better locations of the space. Then, the worst eagles are regularly removed as a survival-of-the-fittest kind of mechanism. Thus, in the latter stages of the process, the best eagles can use more function evaluations in their local food searches.

PEOA is evaluated on a varied set of 20 benchmark functions with different modality, separability, and dimension properties. The results are compared to a set of 13 metaheuristics, nature-inspired, or swarm-intelligence-based algorithms, which contain both classical and modern algorithms.

Given the No-Free-Lunch Theorem, we also explore the specific real-world optimization problems where PEOA can be best and suitably applied. For this paper, the algorithm is used in two applications: solving the inverse conductivity problem of electrical impedance tomography and estimating the parameters of a pendulum-mass-spring-damper system that involves neutral delay differential equations.

Finally, in creating PEOA and proving that it has excellent results, we aspire to give the critically endangered Philippine Eagle much more exposure and possibly help initiate further conservation efforts for the national bird.

G. PAPER ORGANIZATION

The remainder of this paper is organized as follows. Section II provides a detailed description of the proposed PEOA and its components, including the pseudocode and a flowchart. Section III discusses the experimental results and performance comparison of PEOA with other algorithms in solving optimization test functions. Section IV presents the results of PEOA upon application to a real-world optimization problem. Finally, Section V gives the conclusion and recommendations for future research.

II. PHILIPPINE EAGLE OPTIMIZATION ALGORITHM

In this section, we provide a detailed discussion of PEOA. First, we thoroughly explain its three main phases: 1) the Initialization Phase, which is conducted once for the initial generation of eagles, 2) the Local Phase, and 3) the Global Phase, which are the phases performed in every eagle generation. Then, we explain the adaptive mechanisms used by PEOA for its parameters.

A. INITIALIZATION PHASE

Given a bound-constrained minimization problem, i.e., an objective function f to be minimized, a search space having $X_{\min}$ and $X_{\max}$ as its lower and upper bounds, respectively, and a corresponding dimension D, PEOA starts with an initial population of eagles X. Each row of X, given by X_i, represents the ith eagle and is generated as follows:

$$X_i = X_{\min} + [X_{\max} - X_{\min}] \cdot \texttt{lhs}, \quad (1)$$

for $i = 1, 2, \ldots, S_0$, where S_0 is the initial population size of eagles. Here, $X_{\min}$ and $X_{\max}$ are $1 \times D$ vectors and "$\cdot$" is used as a symbol for scalar multiplication. Throughout the paper, we will use this notation for scalar products.

Moreover, `lhs` is a number obtained from a matrix containing a Latin hypercube sample of S_0 rows and D columns. We use this sampling technique so that the initial eagles are randomly generated while being more or less uniformly distributed over each dimension [56].

The function values of the eagles are then obtained and sorted. Because we are considering a minimization problem, the eagle with the least function value is selected as the best eagle of the initial population. We denote this best eagle as $X^\star$.

B. LOCAL PHASE

The best eagle obtained in the previous phase then conducts a local food search within its territory. We denote the best food that it will search as $Y^\star$. The territory has lower bound $Y_{\min}$ and upper bound $Y_{\max}$, which are dependent on a scalar radius Y_{size}. The radius and bounds of the territory are obtained as

$$Y_{\text{size}} = \max[\rho \cdot \min(X_{\max} - X_{\min}), 1], \quad (2)$$

$$Y_{\min} = X^\star - Y_{\text{size}} \cdot \vec{1}, \quad Y_{\max} = X^\star + Y_{\text{size}} \cdot \vec{1}, \quad (3)$$

where $\vec{1}$ is a vector of all ones having D entries.

The max operator is used in Equation (2) to ensure a reasonably large territory where the best eagle can search food, even in cases when a small search space is given. In Equation (2), we set the value of ρ to 0.04. The discussion on how the value of this parameter is chosen can be found in Section III.

We note that if the bounds of the territory are beyond the search space bounds, then the bounds are truncated within the limits of the search space.

The method that the best eagle uses to search for food is the interior point method, where $X^\star$ is taken to be the initial point, $Y_{\min}$ and $Y_{\max}$ are the range bounds, and an initially defined parameter called S_{loc} is assigned as the maximum function evaluations in this phase.

The basis for using this method is the technique proposed in the United Multi-Operator Evolutionary Algorithms-II (UMOEAs-II), which has claimed that the interior point method can increase exploitation ability [57].

Once the best eagle obtains its best food, the Global Phase is conducted, generating a new population of eagles. This new population will again be sorted using their function values, and its new best eagle will likewise be selected to conduct another local food search.

In other words, each generation of eagles has a best eagle that searches locally for food. Therefore, PEOA heavily capitalizes on exploitation to intensify the speed of the optimization process.

On the other hand, for the inspiration source, the territorial behavior of the Philippine Eagle can also be pictured through this local exploitation technique. Figure 3 shows a representation of the territorial local search of the Philippine Eagle.

FIGURE 3. Representation of the territorial local search of the Philippine Eagle, which is the exploitation technique of the proposed PEOA. After a single view of all the valleys between mountains, the Philippine Eagle flies to the lowest valley, or the one that contains the best food. It will then stay and circle about this valley, trying to be as close as possible until it can finally consume the food. This is designed using Canva (https://www.canva.com).

We further note that whenever two consecutive generations select the same $X^\star$, the initial point taken for the interior point method of the latter generation is the $Y^\star$ of the former generation.

C. GLOBAL PHASE

After the Local Phase, the eagle population is divided into three subpopulations, the members of which are dependent on a probability vector, denoted by P. The specific details on how the vector P is obtained can be found in Subsection II-E. Each subpopulation is then assigned an operator, which makes the eagles either move from their original positions or be replaced by new eagles using mutation. After the application of the respective operators, the newly created eagles are referred to as the eagle offspring, denoted by X_{new}. Similar to X, X_{new} has S_0 rows and D columns.

Note that a selection process is carried out here, such that the eagle offspring with improved function values are the only ones that will proceed to the next generation of eagles.

Furthermore, a parameter, called the scaling factor and denoted by F, is used in each operator. This parameter follows a success-history-based parameter adaptation and will be explained in detail in Subsection II-E.

We now thoroughly discuss the three operators, namely 1) the Movement Operator, 2) the Mutation I Operator, and 3) the Mutation II Operator.

Let S denote the size of the whole eagle population of the current generation, and S_1, S_2, S_3 denote the sizes of the subpopulations assigned to the three operators, respectively. Therefore, we have $S = S_1 + S_2 + S_3$. Note that all considered eagles in each operator are of size D.

1) MOVEMENT OPERATOR

For $i = 1, 2, \ldots, S_1$, the Movement Operator is given by

$$(X_{\text{new}})_i = X_i + F_i \cdot (X^\star - X_i + X_{r_1} - X_{\text{arc}} + e^{-d^2} \cdot (X_{\text{near}} - X_i)), \qquad (4)$$

where X_{r_1} is a randomly selected eagle from the current population that is different from $X^\star$. A representation of this operator is shown in Figure 4.

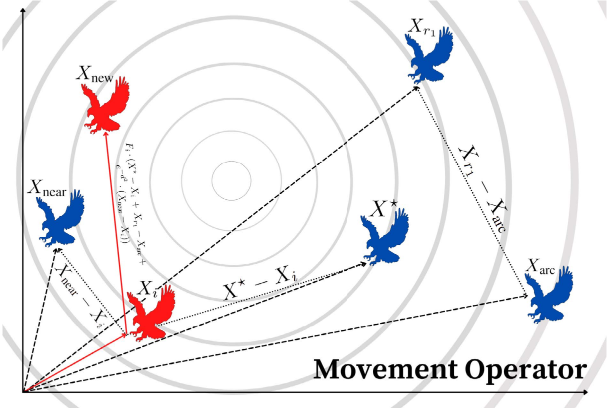


FIGURE 4. **Representation of the movement operator of the proposed PEOA in Equation (4). Five eagles are involved to transform X_i to X_{new}, which are X_i, X_{near}, X^*, X_{r_1}, and X_{arc}. Note that the gray circles are contour plots and the smallest circle in the middle represents the neighborhood containing the global optimum point. Observe that X^* is the best eagle and X_{near} is the closest eagle to X_i. Using the combined position vectors of the five eagles, X_i moves to a better position and becomes X_{new}. This is designed using Canva (https://www.canva.com).**

Also, X_{arc} is another randomly chosen eagle, different from both $X^\star$ and X_{r_1}, taken from the union of the current population and an external archive of eagles.

Finally, X_{near} is the eagle from the current population having the least Euclidean distance d to X_i.

The first part of the Movement Operator is based on an operator used in the Adaptive Differential Evolution Algorithm (JADE), referred to as "DE/current-to-$p$best/1 with archive." This operator has a good searching ability and can also prevent the algorithm from getting trapped in a local minimum due to a bias towards promising directions [58].

The external archive contains the eagles that were not successfully chosen to proceed to the next generations. This archive, also based on JADE, can add more diversity to the eagle population. We note that the archive has a finite size, obtained by multiplying a predefined archive rate A with the initial eagle population size S_0. Randomly selected archive elements are removed if the archive exceeds its predefined size.

A novel feature of the Movement Operator is the addition of a term that considers neighboring eagle proximity. This was included to model the pair hunt strategy of the Philippine Eagle, as the movement of an eagle is dependent on the position of the eagle closest to it.

On the other hand, this term also enhances the efficiency of PEOA because it can make the subpopulation further divide into subgroups, each swarming around different local solutions. One of these local solutions could be the global best solution, so this feature is useful particularly when solving multimodal problems.

2) MUTATION I OPERATOR

For $i = 1, 2, \ldots, S_2$, the Mutation I Operator is given by

$$(X_{\text{new}})_i = F_i \cdot (X_{r_1} + X^\star - X_{r_2}) + S \cdot L(D), \qquad (5)$$

where X_{r_1} and X_{r_2} are distinct eagles that are randomly selected from the current population and must be both different from $X^\star$.

Meanwhile, S is a random vector of size $1 \times D$ having values inside (0, 1). The Lévy flight function, denoted by L, is defined as

$$L(D) = \frac{0.01\, u\sigma}{|v|^{1/\beta}}, \quad \sigma = \left(\frac{\Gamma(1+\beta)\sin(\frac{\pi\beta}{2})}{\beta\Gamma(\frac{1+\beta}{2})2^{\frac{\beta-1}{2}}} \right)^{\frac{1}{\beta}}, \qquad (6)$$

where u and v are values drawn from normal distributions. Also, the parameter β is a default constant set to 1.5, and $\Gamma(x)$ is the Gamma function.

The first part of the Mutation I Operator is based on an operator used in UMOEAs-II [57], called the "DE weighted-rand-to-ϕbest." However, a modification was made, which is the addition of a Lévy flight term. This was done to model the differing flight patterns of the Philippine Eagle mathematically.

Lévy flights are random walks whose step sizes are drawn from a Lévy distribution [4]. They are commonly used to demonstrate the irregular flight behavior of many animals and insects, which exhibit a Lévy-flight-style, intermittent flight pattern [59]. For a more detailed discussion on Lévy flights, we refer the reader to [4] and [60].

3) MUTATION II OPERATOR

For $i = 1, 2, \ldots, S_3$, the Mutation II Operator is given by

$$(X_{\text{new}})_i = F_i \cdot (\hat{X} + X^\star - X_{\text{mean}}), \tag{7}$$

where X_{mean} is the average of all eagles in the current population and $\hat{X}$ is a newly generated random eagle inside the search space.

The Mutation II Operator is similar to one of the operators used in the Harris Hawks Optimization Algorithm (HHO) [61]. This operator not only strengthens the exploration capacity of the algorithm but also models the perching characteristic of the Philippine Eagle. In particular, the addition of X_{mean} depicts how an eagle gets a general picture of the search space, then consequently flies in consideration of the positions of other eagles.

D. ITERATIVE PROCESS OF LOCAL PHASE AND GLOBAL PHASE

Once the operators have been applied to their corresponding subpopulations, the eagle offsprings with improved function values replace their corresponding parent eagles, thus generating a new eagle population.

In the case when some eagles have moved or mutated to locations outside the search space, a resetting scheme is applied based on JADE [58]. The scheme truncates the component of the eagle outside the space bounds within the limits of the space. The function values of these new eagles are sorted once again, and the best eagle of the new population goes back to the Local Phase.

Hence, the Local and Global Phases are carried out iteratively for multiple generations until the given stopping criterion is satisfied. The best food searched by the best eagle at the final generation is the optimal solution of PEOA.

The basic steps of the Philippine Eagle Optimization Algorithm are summarized in the pseudocode shown in Algorithm 1. In addition, a flowchart for PEOA is also provided in Figure 5.

E. ADAPTATION SCHEMES OF PARAMETERS

To further improve the performance of PEOA, the algorithm uses adaptation schemes to control certain parameters. These parameters are the eagle population size S, the probability vector P, and the scaling factor F.

We note that these adaptation schemes were derived from selected papers on evolutionary algorithms and differential evolution: IMODE [17], UMOEAs-II [57], JADE [58], and the Success-History Based Adaptive Differential Evolution with Linear Population Size Reduction Algorithm (L-SHADE) [62]. These papers were chosen because of their proven success as optimization algorithms.

Algorithm 1 Philippine Eagle Optimization Algorithm

```
Input: $f, X_{\min}, X_{\max}, D$
Output: $x^*, f^*$
 1: Define $N_{\max}$, $S_0$, and $S_{\text{loc}}$.
 2: Set $K \leftarrow 0$, $N \leftarrow 0$, and for each $i = 1, 2, 3$, $P_i \leftarrow \frac{1}{3}$.
    {Initialization Phase}
 3: Generate initial population of eagles $X$ of size $S_0$ using eqnarray (1).
 4: Sort $X$ based on function value to determine $X^\star$ and update $N$.
    {Local Phase}
 5: Search $Y^\star$ via interior point method using Equations (2) and (3) with maximum evaluations $S_{\text{loc}}$ and update $N$.
 6: while $|f(Y^\star) - f_{\text{true}}| \geq 10^{-8}$ or $N \leq N_{\max}$ do
 7:     Set $K = K + 1$.
 8:     Update $S$ via linear population size reduction using eqnarray (8).
 9:     Divide eagle population into subpopulations using $P$.
        {Global Phase}
10:     Generate new population of eagles $X_{\text{new}}$ using Equations (4), (5), and (7) via the corresponding operators assigned to the subpopulations.
11:     Sort $X_{\text{new}}$ based on function value to obtain the new $X^\star$ and update $N$.
        {Local Phase}
12:     Repeat the Local Phase (Step 5) with the updated $X^\star$.
13:     Update $P$ based on the improvement rate of each operator using Equations (9) and (10).
14: end while
15: return $x^* = Y^\star$ and $f^* = f(Y^\star)$
```

We discuss how S, P, and F are determined based on the papers mentioned. For a more in-depth analysis of the behavior of these parameters, we refer the reader to [17], [57], [58], and [62].

1) LINEAR POPULATION SIZE REDUCTION

After every generation, a linear reduction of the entire eagle population size S is carried out as

$$S = \left\llbracket S_0 + (S_{\min} - S_0) \cdot \frac{N}{N_{\max}} \right\rrbracket, \tag{8}$$

where S_0 is the initial population size of eagles, N is the current number of function evaluations, and $N_{\max}$ is the maximum number of function evaluations.

Moreover, $S_{\min}$ is the minimum possible population size. For PEOA, we set $S_{\min} = 5$, since the Movement Operator requires at least five eagles. The worst eagles of the population, i.e. the eagles with the highest function values, are removed to meet the required population size.

Derived from L-SHADE [62], this mechanism can maintain diversity during the earlier generations, then enhance the exploitation ability in the later ones.

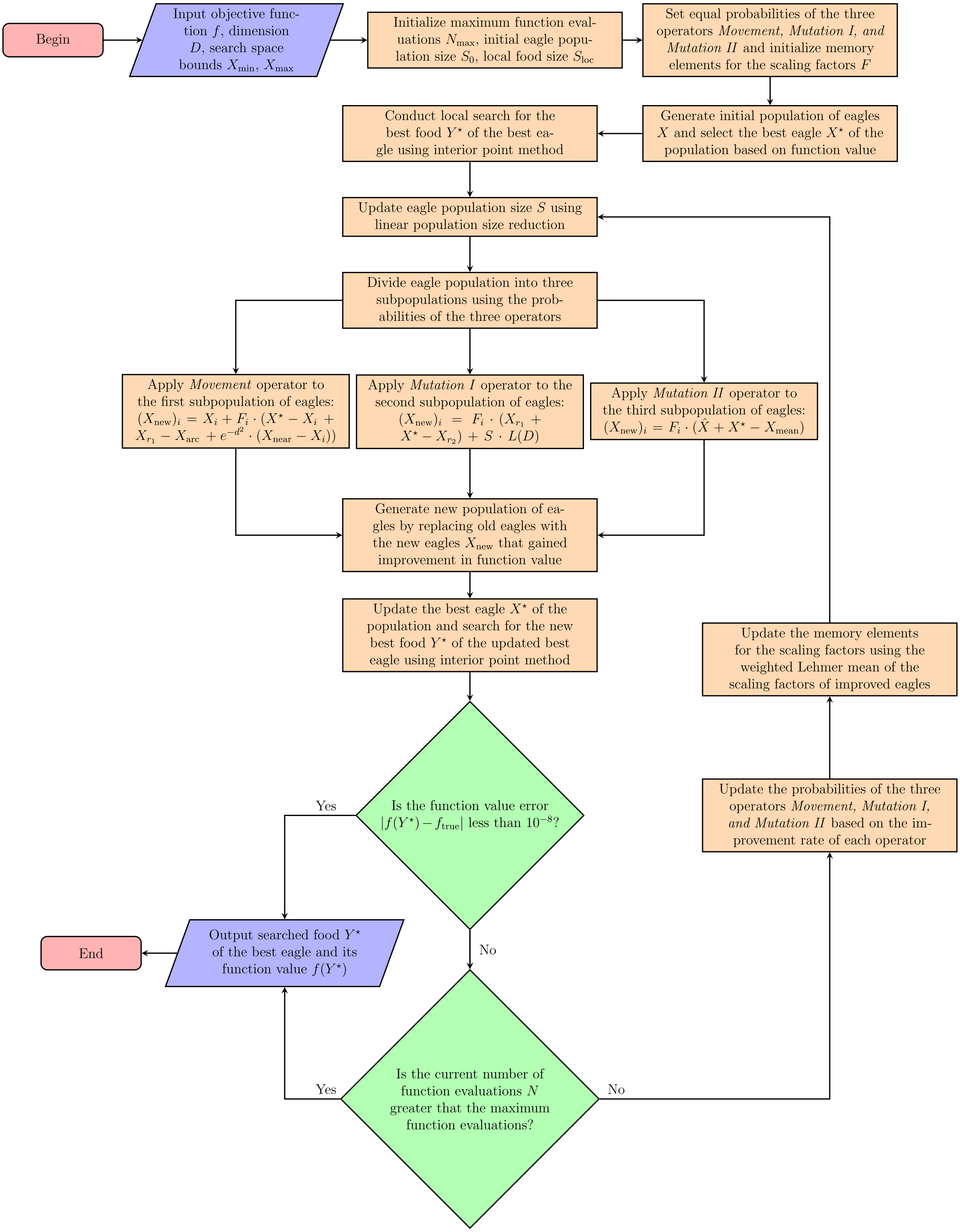

Begin
Input objective function f, dimension D, search space bounds $X_{\min}$, $X_{\max}$
Initialize maximum function evaluations $N_{\max}$, initial eagle population size S_0, local food size S_{loc}
Set equal probabilities of the three operators *Movement, Mutation I, and Mutation II* and initialize memory elements for the scaling factors F
Generate initial population of eagles X and select the best eagle $X^\star$ of the population based on function value
Conduct local search for the best food $Y^\star$ of the best eagle using interior point method
Update eagle population size S using linear population size reduction
Divide eagle population into three subpopulations using the probabilities of the three operators
Apply *Movement* operator to the first subpopulation of eagles: $(X_{\text{new}})_i = X_i + F_i \cdot (X^\star - X_i + X_{r_1} - X_{\text{arc}} + e^{-d^2} \cdot (X_{\text{near}} - X_i))$
Apply *Mutation I* operator to the second subpopulation of eagles: $(X_{\text{new}})_i = F_i \cdot (X_{r_1} + X^\star - X_{r_2}) + S \cdot L(D)$
Apply *Mutation II* operator to the third subpopulation of eagles: $(X_{\text{new}})_i = F_i \cdot (\hat{X} + X^\star - X_{\text{mean}})$
Generate new population of eagles by replacing old eagles with the new eagles X_{new} that gained improvement in function value
Update the best eagle $X^\star$ of the population and search for the new best food $Y^\star$ of the updated best eagle using interior point method
Is the function value error $|f(Y^\star) - f_{\text{true}}|$ less than 10^{-8}?
Yes
No
Is the current number of function evaluations N greater that the maximum function evaluations?
Yes
No
Output searched food $Y^\star$ of the best eagle and its function value $f(Y^\star)$
End
Update the probabilities of the three operators *Movement, Mutation I, and Mutation II* based on the improvement rate of each operator
Update the memory elements for the scaling factors using the weighted Lehmer mean of the scaling factors of improved eagles

FIGURE 5. Flowchart summarizing the steps of the Philippine Eagle optimization algorithm.

2) IMPROVEMENT RATES FOR THE SUBPOPULATION SIZES

The probability vector, denoted by P, has three entries. These entries contain probability values that guide the assignment of eagles into the subpopulations.

Initially, the values are all set to $\frac{1}{3}$. For each eagle in the initial population, a random number j between 0 and 1 is obtained, and if $j \leq \frac{1}{3}$, then it will be evolved using the Movement Operator. Otherwise, if $\frac{1}{3} < j \leq \frac{2}{3}$, then this eagle will be evolved using the Mutation I Operator. Else, it will be evolved using the Mutation II Operator.

Afterward, the probabilities are modified depending on the improvement rates of the operators. If S_i is the subpopulation size corresponding to operator i, for $i = 1, 2, 3$, then the improvement rate R_i is calculated as

$$R_i = \frac{\sum_{z=1}^{S_i} \max(0, f_{\text{old},z} - f_{\text{new},z})}{\sum_{z=1}^{S_i} f_{\text{old},z}}, \tag{9}$$

where $f_{\text{old},z}$ and $f_{\text{new},z}$ are the function values of the current eagle and its corresponding eagle offspring, respectively.

Then, the probability value P_i corresponding to operator i is updated as

$$P_i = \max\left[0.1, \min\left(0.9, \frac{R_i}{R_1 + R_2 + R_3}\right)\right]. \tag{10}$$

Derived from UMOEAs-II [57], this mechanism highlights the best-performing operator per generation, giving it more control of the optimization process. Meanwhile, the underperforming operators are given a chance to improve in the next generations.

3) ADAPTIVE CONTROL OF THE SCALING FACTOR

During the Global Phase, every eagle is associated with a scaling factor F_i. This scaling factor is generated according to a Cauchy distribution with mean μ_{F_i} and variance 0.1.

If $F_i \geq 1$, then it is truncated to be 1, and if $F_i \leq 0$, then it is regenerated. The mean values μ_{F_i} come from a particular memory, which has a predefined memory size H. The values in the memory are all initially set to 0.2.

We note that the constant values used in this scheme, namely the variance of 0.1 and the initial values of the memory given by 0.2, are the values chosen by IMODE [17]. Therefore, for consistency, we retain these values for PEOA.

Then, a memory element is updated whenever a generation has at least one eagle offspring with an improved function value. In this case, the scaling factors corresponding to the improved eagle offspring are recorded in a vector G.

The update is done using the weighted Lehmer mean as

$$\text{mean}_{WL}(G) = \frac{\sum_{k=1}^{|G|} w_k F_k^2}{\sum_{k=1}^{|G|} w_k F_k}, \quad w_k = \frac{\Delta f_k}{\sum_{\ell=1}^{|G|} \Delta f_\ell}, \tag{11}$$

where F_k is the kth scaling factor contained in G, and Δf_k is the change in function value of the kth eagle offspring.

Based on JADE [58], the Cauchy distribution is more capable of diversifying the scaling factors compared to a normal distribution. Also, the weighted Lehmer mean is more effective than the usual arithmetic mean because the former can generate larger scaling factors. This improves the progress rate of PEOA.

III. EXPERIMENTAL RESULTS AND DISCUSSION

We subject PEOA to benchmark tests to assess its performance in this section. In particular, we apply PEOA to optimization test functions and compare PEOA with other recent optimization algorithms. We also describe the parameter settings of PEOA and the experimental setup of our comparative analysis.

A. BENCHMARK OPTIMIZATION TEST FUNCTIONS

We apply PEOA on 20 optimization test functions having varied combinations of properties among modality, separability, and dimension. We first explain what these properties mean and how they contribute to the difficulty of an optimization problem.

A function with only one local optimum is called unimodal, whereas it is called multimodal if it has two or more local optima [48]. One aspect of a well-designed exploration process in an algorithm is the capacity to escape from any local yet nonglobal optimum. Unimodality, on the contrary, examines the exploitation capability of an algorithm [63].

Separable and nonseparable functions formulate another classification of functions. A function of n variables is called separable if it can be written as a sum of n functions of just one variable, that is, its variables are independent of each other [64]. On the other hand, a function is called nonseparable if its variables show interrelation among themselves and are thus not independent. It is relatively easier to solve separable functions because they can be decomposed into independent subfunctions, each one of which can be optimized independently [48].

Finally, the dimension, that is, the number of variables a function has, also dictates the difficulty of an optimization problem. As the dimension increases, the search space enlarges exponentially, thus making it more challenging for an algorithm to find the optimal solution [65].

Therefore, we divide our experimentation into four different types of functions, namely five unimodal and separable functions, five multimodal and separable functions, five unimodal and nonseparable functions, and five multimodal and nonseparable functions, obtained from [48] and [66].

For each of these 20 functions, we use dimensions of 2, 5, 10, and 20, thus giving 80 experiments in total. Therefore, we have chosen an extensive test suite that accommodates a wide variety of function properties.

Table 2 presents the functions used in our experiments, along with their corresponding search range, true optimal function value, and true optimal solution.

B. PARAMETER SETTINGS OF PEOA

The adopted parameter values of PEOA in our experiments are shown in Table 4.

TABLE 2. Optimization test functions having varied combinations of types and dimensions which were applied to the Philippine Eagle optimization algorithm and 13 other examined algorithms.

Type	Name	Function	Dimension D	Range	f_{true}	Solution x_{true}
Unimodal & Separable	Powell Sum	$f_1(\mathbf{x}) = \sum_{i=1}^{D} \lvert x_i \rvert^{i+1}$	2, 5, 10, 20	$[-1, 1]$	0	$(0, \ldots, 0)$
	Schwefel 2.20	$f_2(\mathbf{x}) = \sum_{i=1}^{D} \lvert x_i \rvert$	2, 5, 10, 20	$[-100, 100]$	0	$(0, \ldots, 0)$
	Schwefel 2.21	$f_3(\mathbf{x}) = \max_{i=1,\ldots,D} \lvert x_i \rvert$	2, 5, 10, 20	$[-100, 100]$	0	$(0, \ldots, 0)$
	Sphere	$f_4(\mathbf{x}) = \sum_{i=1}^{D} x_i^2$	2, 5, 10, 20	$[-5.12, 5.12]$	0	$(0, \ldots, 0)$
	Sum Squares	$f_5(\mathbf{x}) = \sum_{i=1}^{D} i x_i^2$	2, 5, 10, 20	$[-10, 10]$	0	$(0, \ldots, 0)$
Multimodal & Separable	Alpine 1	$f_6(\mathbf{x}) = \sum_{i=1}^{D} \lvert x_i \sin(x_i) + 0.1 x_i \rvert$	2, 5, 10, 20	$[0, 10]$	0	$(0, \ldots, 0)$
	Wavy	$f_7(\mathbf{x}) = 1 - \frac{1}{D} \sum_{i=1}^{D} \cos(10 x_i) \exp\left(-\frac{1}{2} x_i^2\right)$	2, 5, 10, 20	$[-\pi, \pi]$	0	$(0, \ldots, 0)$
	Qing	$f_8(\mathbf{x}) = \sum_{i=1}^{D} (x_i^2 - i)^2$	2, 5, 10, 20	$[-500, 500]$	0	$(\pm 1, \ldots, \pm\sqrt{D})$
	Rastrigin	$f_9(\mathbf{x}) = 10D + \sum_{i=1}^{D} \left[x_i^2 - 10\cos(2\pi x_i)\right]$	2, 5, 10, 20	$[-5.12, 5.12]$	0	$(0, \ldots, 0)$
	Xin-She Yang 1	$f_{10}(\mathbf{x}) = \sum_{i=1}^{D} \texttt{rand}_i \lvert x_i \rvert^i$	2, 5, 10, 20	$[-5, 5]$	0	$(0, \ldots, 0)$
Unimodal & Nonseparable	Brown	$f_{11}(\mathbf{x}) = \sum_{i=1}^{D-1} \left[(x_i^2)^{(x_{i+1}^2+1)} + (x_{i+1}^2)^{(x_i^2+1)}\right]$	2, 5, 10, 20	$[-1, 4]$	0	$(0, \ldots, 0)$
	Rosenbrock	$f_{12}(\mathbf{x}) = \sum_{i=1}^{D-1} \left[100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2\right]$	2, 5, 10, 20	$[-5, 10]$	0	$(1, \ldots, 1)$
	Schwefel 2.22	$f_{13}(\mathbf{x}) = \sum_{i=1}^{D} \lvert x_i \rvert + \prod_{i=1}^{D} \lvert x_i \rvert$	2, 5, 10, 20	$[-100, 100]$	0	$(0, \ldots, 0)$
	Xin-She Yang 3	$f_{14}(\mathbf{x}) = \exp\left(-\sum_{i=1}^{D} \left(\frac{x_i}{15}\right)^{10}\right) - 2\exp\left(-\sum_{i=1}^{D} x_i^2\right) \prod_{i=1}^{D} \cos^2(x_i)$	2, 5, 10, 20	$[-2\pi, 2\pi]$	-1	$(0, \ldots, 0)$
	Zakharov	$f_{15}(\mathbf{x}) = \sum_{i=1}^{D} x_i^2 + \left(\sum_{i=1}^{D} 0.5 i x_i\right)^2 + \left(\sum_{i=1}^{D} 0.5 i x_i\right)^4$	2, 5, 10, 20	$[-5, 10]$	0	$(0, \ldots, 0)$
Multimodal & Nonseparable	Ackley	$f_{16}(\mathbf{x}) = -20 \cdot \exp\left(-0.2\sqrt{\frac{1}{D} \sum_{i=1}^{D} x_i^2}\right) - \exp\left(\frac{1}{D} \sum_{i=1}^{D} \cos(2\pi x_i)\right) + 20 + \exp(1)$	2, 5, 10, 20	$[-32.768, 32.768]$	0	$(0, \ldots, 0)$
	Periodic	$f_7(\mathbf{x}) = 1 + \sum_{i=1}^{D} \sin^2(x_i) - 0.1\exp\left(-\sum_{i=1}^{D} x_i^2\right)$	2, 5, 10, 20	$[-10, 10]$	0.9	$(0, \ldots, 0)$
	Griewank	$f_{18}(\mathbf{x}) = 1 + \sum_{i=1}^{D} \frac{x_i^2}{4000} - \prod_{i=1}^{D} \cos\left(\frac{x_i}{\sqrt{i}}\right)$	2, 5, 10, 20	$[-100, 100]$	0	$(0, \ldots, 0)$
	Salomon	$f_{19}(\mathbf{x}) = 1 - \cos\left(2\pi\sqrt{\sum_{i=1}^{D} x_i^2}\right) + 0.1\sqrt{\sum_{i=1}^{D} x_i^2}$	2, 5, 10, 20	$[-100, 100]$	0	$(0, \ldots, 0)$
	Xin-She Yang 4	$f_{20}(\mathbf{x}) = \left(\sum_{i=1}^{D} \sin^2(x_i) - \exp\left(-\sum_{i=1}^{D} x_i^2\right)\right) \exp\left(-\sum_{i=1}^{D} \sin^2(\sqrt{\lvert x_i \rvert})\right)$	2, 5, 10, 20	$[-10, 10]$	-1	$(0, \ldots, 0)$

We recall that the constant value of 0.04 is used in Equation (2) for the cluster size factor ρ. This parameter controls the cluster size of each local food search. To provide an analysis of ρ, experiments were done to determine the best value of this constant such that PEOA could give the most optimal results.

Recall from Section II-B that Equation (2) is

$$Y_{\text{size}} = \max[\rho \cdot \min(X_{\max} - X_{\min}), 1].$$

Different values for the parameter ρ were considered. For each value of ρ, PEOA was tested on the 20 test functions given in Table 2. For this simulation, we set the dimension to 5 and run the algorithm 20 times.

The results obtained by PEOA for this experiment are summarized in Table 3. Observe that the value of ρ that gave the best average result (highlighted in green) is 0.04.

To illustrate how the population size S is updated per generation, PEOA was implemented once for the Xin-She Yang 1 function with dimension 2. After 21 generations, PEOA attained an optimal function value of 6.7459E-09.

The population sizes obtained from this experiment are

$$S = \begin{bmatrix} 80 & 79 & 79 & 78 & 78 & 77 & 77 \\ 76 & 76 & 76 & 75 & 75 & 74 & 74 \\ 73 & 73 & 73 & 72 & 72 & 71 & 71 \end{bmatrix}.$$

We thus see that the population sizes decrease linearly. Recall from Equation (8) that the slope of this decrease is $\frac{S_{\min} - S_0}{N_{\max}}$, where $S_{\min}$ is the minimum population size, S_0 is the initial population size, and $N_{\max}$ is the maximum number of function evaluations.

We take the minimum eagle population size $S_{\min} = 5$ since the Movement Operator in Section $II - C1$ requires at least

TABLE 3. Parameter tuning of cluster size factor ρ which determines the cluster size per local search. Optimal function values obtained by the Philippine Eagle optimization algorithm using different values for ρ. Experiments were done on each test function with dimension 5, and results were averaged over 20 independent runs for each function. The value of 0.04 gave the best result (highlighted in green).

parameter ρ	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.1
Ackley, D = 5	1.7782E-09	9.7759E-10	4.1470E-10	1.0584E-09	3.2541E-10	2.8472E-10	4.6380E-10	3.8248E-10	9.9210E-10	4.3085E-10
Wavy, D = 5	2.0982E-14	2.7235E-14	9.6250E-15	2.1256E-14	9.3207E-15	9.5119E-15	2.1869E-14	1.2354E-14	2.8362E-14	1.9498E-14
Salomon, D = 5	4.9488E-11	1.8624E-11	1.5483E-11	1.3563E-11	4.9937E-03	4.9937E-03	1.4981E-02	9.9873E-03	1.1997E-11	1.2657E-11
Xin-She Yang 4, D = 5	1.8607E-09	1.7819E-09	1.8065E-09	2.1454E-09	2.7005E-09	1.8437E-09	1.1687E-09	3.2604E-09	1.0773E-09	1.6042E-09
Alpine 1, D = 5	0	0	0	0	0	0	0	0	0	0
Periodic, D = 5	3.2145E-14	1.2688E-14	2.1868E-14	4.7343E-15	1.4275E-14	2.6242E-14	4.4572E-14	2.4870E-14	2.2104E-14	1.0323E-14
Qing, D = 5	2.3845E-14	1.5176E-14	1.9453E-14	1.9354E-14	1.4227E-14	1.6687E-14	1.9944E-14	2.5450E-14	3.3501E-10	1.9552E-10
Xin-She Yang 1, D = 5	1.6708E-07	2.2784E-07	3.4547E-07	1.5467E-07	1.0378E-07	1.5530E-07	8.7675E-07	2.6181E-07	2.4350E-07	1.9822E-06
Griewank, D = 5	2.6104E-13	3.2783E-13	2.5521E-13	4.0403E-13	5.3205E-13	4.0645E-13	4.0323E-13	1.6342E-13	4.8389E-13	1.8385E-13
Sum Squares, D = 5	6.1803E-15	5.0441E-15	9.9334E-15	7.5859E-15	8.1303E-15	5.4330E-15	1.1270E-14	6.7452E-13	1.9735E-15	4.0433E-14
Schwefel 2.20, D = 5	3.1603E-09	1.5984E-09	2.0958E-09	1.9398E-09	2.0064E-09	1.9515E-09	1.9019E-09	1.5489E-09	1.6727E-09	2.0413E-09
Powell Sum, D = 5	3.5203E-09	3.4654E-09	3.4013E-09	3.3918E-09	4.5497E-09	4.1739E-09	2.7353E-09	2.9486E-09	3.2139E-09	3.1862E-09
Zakharov, D = 5	4.0617E-14	1.3912E-14	3.2947E-14	1.4049E-14	1.5183E-14	2.1480E-14	1.6598E-14	9.3915E-15	1.0069E-14	8.0972E-14
Xin-She Yang 3, D = 5	2.3057E-13	2.4066E-14	1.3899E-14	1.2321E-14	2.6746E-14	5.4959E-15	3.7886E-14	6.7816E-13	8.0520E-15	8.8837E-15
Schwefel 2.22, D = 5	1.5485E-09	2.4281E-09	3.6965E-09	2.0066E-09	1.2923E-09	2.1754E-09	2.2947E-09	2.0411E-09	1.8680E-09	2.3485E-09
Schwefel 2.21, D = 5	4.6529E-10	5.2274E-10	4.9110E-10	2.7100E-10	1.2535E-09	1.3283E-10	5.4769E-10	1.3827E-09	5.9170E-10	3.7210E-10
Brown, D = 5	8.1158E-13	2.5861E-13	3.6792E-11	1.1512E-13	1.7539E-13	1.3452E-13	2.4183E-13	1.0823E-13	4.4943E-13	2.9509E-13
Rastrigin, D = 5	0	0	0	0	0	0	0	0	0	0
Sphere, D = 5	5.6480E-12	8.7120E-14	1.5976E-12	4.2641E-13	6.2517E-13	3.4748E-12	1.2214E-13	1.7405E-12	3.7974E-14	7.0133E-13
Rosenbrock, D = 5	5.3620E-11	5.8565E-11	5.4455E-11	1.2323E-10	5.6634E-11	5.6250E-11	5.3570E-11	5.4670E-11	5.4045E-11	4.4657E-11
average function value	8.9761E-09	1.1934E-08	1.7874E-08	8.2810E-09	2.4969E-04	2.4969E-04	7.4909E-04	4.9938E-04	1.2666E-08	9.9623E-08

TABLE 4. Parameter values adopted for the Philippine Eagle optimization algorithm in the experimentations with 13 other examined optimization algorithms.

Parameter Name	Symbol	Description	Value	References
function	f	benchmark optimization test function with different modality, separability, and dimension properties	Table II	[48], [66]
dimension	D	number of variables a function has	2, 5, 10, 20	CEC 2020 Comp. on SO-BCO [67]
minimum eagle population size	$S_{\min}$	minimum number of surviving eagles throughout the optimization process	5	Equation (8)
cluster size factor	ρ	constant factor that determines the cluster size of each local food search	0.04	Table III
maximum function evaluations	$N_{\max}$	maximum number of function evaluations throughout the optimization process	$10000 \cdot D$	CEC 2020 Comp. on SO-BCO [67]
scaling factor	F_i	factor associated with each eagle i during the Global Phase which is obtained using a Cauchy distribution	$\text{Cauchy}(\mu_{F_i}, 0.1)$	JADE [58], UMOEAs-II [57], L-SHADE [62]
archive rate	A	number multiplied to the initial eagle population size which determines the size of the external archive of recorded eagles	2.6	IMODE [17]
memory size	H	size of the memory containing mean values μ_{F_i} which are used to determine the scaling factor F_i	$20 \cdot D$	IMODE [17]

five eagles. The choice for the values of the dimension D and the maximum function evaluations $N_{\max}$ was based on the CEC 2020 Special Session and Competition on Single Objective Bound Constrained Numerical Optimization (CEC 2020 Comp. on SO-BCO) [67].

For the analysis of the other parameters adopted by PEOA, such as the scaling factors F_i in Equations (4), (5), and (7), the archive rate A from Section II-C1, and the memory size H from Section II-E3, we refer the reader to IMODE [17], UMOEAs-II [57], JADE [58], and L-SHADE [62].

Lastly, the value for the initial eagle population size S_0 is $20 \cdot D^2$ while the value of the local food size S_{loc} is $10 \cdot D^2$. These are the general default settings for PEOA.

C. EXPERIMENTAL SETUP OF COMPARATIVE ANALYSIS

In solving the test functions, we compare the performance of PEOA to a set of metaheuristic algorithms, swarm intelligence algorithms, and nature-inspired heuristics.

Specifically, the 13 selected algorithms for comparison are Genetic Algorithm [5], Particle Swarm Optimization [44], Flower Pollination Algorithm [10], [68], Bat Algorithm

TABLE 5. Parameter settings and initial values of the 13 examined optimization algorithms for comparison and evaluation of the Philippine Eagle optimization algorithm.

Algorithm	Parameter	Value
GA	type	real coded
	selection	stochastic uniform
	recombination	scattered
	mutation	Gaussian
	dimension (D)	2, 5, 10, 20
	population size (S)	50 if $D \leq 5$, 200 o/w
	maximum number of function evaluations ($N_{\max}$)	$10000 \cdot D$
	maximum number of generations	$\text{round}\left(\frac{N_{\max}}{S}\right)$
PSO	self-adjustment weight	1.49
	social adjustment weight	1.49
	minimum adaptive neighborhood size	0.25
	inertia range	$[0.1, 1.1]$
	dimension (D)	2, 5, 10, 20
	population size (S)	$\min(100, 10 \cdot D)$
	maximum number of function evaluations ($N_{\max}$)	$10000 \cdot D$
	maximum number of iterations	$\text{round}\left(\frac{N_{\max}}{S}\right)$
FPA	probability switch	0.8
	dimension (D)	2, 5, 10, 20
	population size (S)	20
	maximum number of function evaluations ($N_{\max}$)	$10000 \cdot D$
	maximum number of iterations	$\text{round}\left(\frac{N_{\max}}{S}\right)$
BA	loudness	0.5
	pulse rate	0.5
	frequency range	$[0, 2]$
	dimension (D)	2, 5, 10, 20
	population size	20
	maximum number of function evaluations	$10000 \cdot D$
CS	discovery rate of alien eggs	0.25
	dimension (D)	2, 5, 10, 20
	population size	25
	maximum number of function evaluations	$10000 \cdot D$
FA	randomization parameter	0.5
	minimum attractiveness value	0.2
	light absorption coefficient	1
	dimension (D)	2, 5, 10, 20
	population size (S)	20
	maximum number of function evaluations ($N_{\max}$)	$10000 \cdot D$
	maximum number of iterations	$\text{round}\left(\frac{N_{\max}}{S}\right)$
WOA	convergence constant	$[2, 0]$
	spiral factor	1
	dimension (D)	2, 5, 10, 20
	population size (S)	30
	maximum number of function evaluations ($N_{\max}$)	$10000 \cdot D$
	maximum number of iterations	$\text{round}\left(\frac{N_{\max}}{S}\right)$
MFO	convergence constant	$[-1, -2]$
	spiral factor	1
	dimension (D)	2, 5, 10, 20
	population size (S)	30
	maximum number of function evaluations ($N_{\max}$)	$10000 \cdot D$
	maximum number of iterations	$\text{round}\left(\frac{N_{\max}}{S}\right)$
BOA	dimension (D)	2, 5, 10, 20
	population size (S)	50
	maximum number of function evaluations ($N_{\max}$)	$10000 \cdot D$
	maximum number of iterations	$\text{round}\left(\frac{N_{\max}}{S}\right)$
IMODE	dimension (D)	2, 5, 10, 20
	initial population size	$6 \cdot D^2$
	minimum number of individuals	4
	maximum number of function evaluations ($N_{\max}$)	$10000 \cdot D$
	current number of function evaluations (N)	$[0, N_{\max}]$
	minimum function evaluations to begin local search	$0.85 \cdot N_{\max}$
	local search probability	0.1
	maximum local search evaluations	$\text{ceil}(0.02 \cdot N_{\max}, N_{\max} - N)$
	archive rate	2.6
	memory size	$20 \cdot D$
	scaling factor	see L-SHADE [62]
	crossover rate	see L-SHADE [62]
WFS	flier's velocity	random between 10 and 100
	dimension (D)	2, 5, 10, 20
	total number of points at each iteration	$1000 \cdot D$
	maximum number of function evaluations	$10000 \cdot D$
	maximum number of iterations	10
CHOA	dimension (D)	2, 5, 10, 20
	population size	50
	maximum number of function evaluations	$10000 \cdot D$
	maximum number of iterations	$200 \cdot D$
TSA	dimension (D)	2, 5, 10, 20
	population size	30
	maximum number of function evaluations	$10000 \cdot D$
	maximum number of iterations	$500 \cdot D$

[9], [69], Cuckoo Search Algorithm [8], [70], Firefly Algorithm [7], [71], Whale Optimization Algorithm [12], [72], Moth Flame Optimization Algorithm [11], [73], Butterfly Optimization Algorithm [13], [74], Improved Multi-Operator Differential Evolution [17], Chimp Optimization Algorithm [16], [75], Wingsuit Flying Search [14], [76], and Tunicate Swarm Algorithm [15], [77].

Our experimental setup is based on the experimental settings recommended by the CEC 2020 Special Session and Competition on Single Objective Bound Constrained Numerical Optimization (CEC 2020 Comp. on SO-BCO) [67]. These settings ensure the efficiency and fairness of the comparison of competing algorithms.

The features of our experimental setup are the following:

- Default values of the parameters of each selected algorithm are used, as shown in Table 5.
- The total number of independent runs for each algorithm (per test function) is 30.
- The maximum number of function evaluations for all algorithms is $10000 \cdot D$, where $D = 2, 5, 10, 20$ is the dimension.
- We emphasize that the maximum number of evaluations is the chosen parameter to be kept constant for all the algorithms in our experiments. On the other hand, the population size and the maximum number of iterations may vary per algorithm depending on their default parameters.
- For the termination criteria, an algorithm is terminated once it reaches the maximum number of function evaluations or if its function value error, or the distance between its obtained optimal value and the true optimal value, is lesser than 10^{-8}.
- Function value errors less than 10^{-8} are treated as zero.
- Four performance indicators for function value errors are used, namely the best, worst, mean, and standard deviation (Stdev) of the results over 30 runs of each algorithm (per test function). These statistical measurements reflect the success rates of the algorithms because they show the algorithm's capability of accurately finding a solution while maintaining low variation in the results.
- Wilcoxon's rank-sum test [78] is used as a nonparametric statistical tool to further justify any significant difference between PEOA and the other examined algorithms. Examples of the application of this test can be found in [79]–[81]. We use the MATLAB function `ranksum` to find the p-values at a significance level of 5%. A p-value less than the significance level indicates a rejection of the null hypothesis.
- Average number of function evaluations upon reaching the termination criteria are also recorded per algorithm.
- Standard benchmark test functions are chosen from [48] and [66], as shown in Table 2. The test suite is chosen to be large enough to include a diverse collection of problems, ranging from unimodal to multimodal, from

TABLE 6. Average, best, and worst function value errors and standard deviations over 30 independent runs obtained by the Philippine Eagle optimization algorithm compared to the 13 other examined algorithms for *5 different unimodal and separable functions* of dimension 20.

Function		PEOA	GA	PSO	FPA	BA	CS	FA	WOA	MFO	BOA	CHOA	WFS	TSA	IMODE
Powell Sum, D = 20	Mean	0	4.96E-05	0	0	0	0	0	0	0	0	0	1.02E-08	0	0
	Best	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Worst	0	6.97E-04	0	0	0	0	0	0	0	0	0	1.55E-08	0	0
	Stdev	0	1.64E-04	0	0	0	0	0	0	0	0	0	0	0	0
Schwefel 2.20, D = 20	Mean	0	2.27E-04	0	1.33E-04	3.11E+02	0	5.38E-02	0	1.67E+01	0	0	1.50E-01	0	1.35E-06
	Best	0	8.85E-05	0	5.98E-06	1.57E+02	0	3.92E-02	0	0	0	0	1.08E-01	0	5.44E-07
	Worst	2.51E-07	3.96E-04	0	7.87E-04	5.18E+02	0	7.25E-02	0	1.00E+02	0	0	2.12E-01	0	3.13E-06
	Stdev	4.56E-08	7.29E-05	0	1.49E-04	7.25E+01	0	7.17E-03	0	3.79E+01	0	0	2.62E-02	0	6.95E-07
Schwefel 2.21, D = 20	Mean	0	1.09E+00	1.27E-05	9.63E+00	4.03E+01	2.05E-02	7.79E-03	1.34E+00	3.59E+01	0	0	2.70E-01	0	1.93E-06
	Best	0	3.42E-01	1.41E-08	3.76E+00	2.20E+01	1.99E-05	5.08E-03	0	6.10E+00	0	0	1.20E-01	0	3.05E-07
	Worst	0	2.32E+00	3.78E-04	2.02E+01	5.42E+01	2.63E-01	1.01E-02	2.07E+01	6.95E+01	0	0	5.85E-01	0	7.01E-06
	Stdev	0	4.97E-01	6.91E-05	4.22E+00	8.94E+00	5.06E-02	1.21E-03	4.38E+00	1.52E+01	0	0	1.01E-01	0	1.60E-06
Sphere, D = 20	Mean	0	8.61E-08	0	7.00E-07	4.10E-06	0	5.34E-07	0	0	0	0	4.09E-06	0	0
	Best	0	1.48E-08	0	0	2.66E-06	0	2.09E-07	0	0	0	0	1.50E-06	0	0
	Worst	0	3.26E-07	0	4.05E-06	5.08E-06	0	7.17E-07	0	0	0	0	8.50E-06	0	0
	Stdev	0	6.42E-08	0	1.13E-06	6.56E-07	0	9.43E-08	0	0	0	0	1.79E-06	0	0
Sum Squares, D = 20	Mean	0	1.52E-07	0	2.07E-05	4.46E-05	0	2.30E-05	0	1.47E+02	0	0	2.28E-04	0	0
	Best	0	5.63E-08	0	5.85E-08	2.71E-05	0	1.61E-05	0	0	0	0	8.10E-05	0	0
	Worst	0	3.65E-07	0	1.10E-04	6.98E-05	0	4.08E-05	0	1.00E+03	0	0	4.42E-04	0	0
	Stdev	0	7.53E-08	0	2.77E-05	8.80E-06	0	5.83E-06	0	2.29E+02	0	0	9.19E-05	0	0

TABLE 7. Average, best, and worst function value errors and standard deviations over 30 independent runs obtained by the Philippine Eagle optimization algorithm compared to the 13 other examined algorithms for *5 different unimodal and nonseparable functions* of dimension 20.

Function		PEOA	GA	PSO	FPA	BA	CS	FA	WOA	MFO	BOA	CHOA	WFS	TSA	IMODE
Brown, D = 20	Mean	0	7.83E-08	0	4.45E-06	7.75E-06	0	2.37E-07	0	4.37E+00	0	0	6.83E-02	0	0
	Best	0	1.07E-08	0	0	4.43E-06	0	1.50E-07	0	0	0	0	1.07E-02	0	0
	Worst	0	2.48E-07	0	8.12E-05	9.11E-06	0	3.68E-07	0	1.40E+01	0	0	2.33E-01	0	0
	Stdev	0	5.63E-08	0	1.47E-05	1.14E-06	0	5.21E-08	0	3.51E+00	0	0	5.14E-02	0	0
Rosenbrock, D = 20	Mean	0	1.06E+01	5.07E+00	1.75E+01	6.39E+00	4.58E-01	1.43E+01	1.44E+01	2.30E+04	1.86E+01	1.86E+01	7.14E+01	2.01E+01	0
	Best	0	1.65E-03	2.75E-03	9.03E-02	3.12E-03	0	1.20E+01	1.39E+01	1.08E+00	1.85E+01	1.73E+01	1.25E+01	1.43E+01	0
	Worst	0	7.09E+01	1.08E+01	1.01E+02	7.51E+01	3.69E+00	1.73E+01	1.51E+01	1.10E+05	1.88E+01	1.90E+01	1.66E+02	8.07E+01	0
	Stdev	0	1.99E+01	2.88E+00	2.53E+01	1.78E+01	9.90E-01	1.34E+00	3.44E-01	3.51E+04	6.98E-02	4.22E-01	4.49E+01	1.15E+01	0
Schwefel 2.22, D = 20	Mean	0	1.95E-04	0	9.97E+01	2.35E+19	1.00E+10	5.37E-02	0	2.60E+02	4.38E+27	0	1.22E-01	0	2.44E-08
	Best	0	7.59E-05	0	2.19E-02	3.35E+09	1.00E+10	4.33E-02	0	0	7.46E+22	0	8.69E-02	0	0
	Worst	0	4.34E-04	1.78E-08	2.62E+02	5.33E+20	1.00E+10	6.42E-02	0	5.00E+02	2.21E+28	0	1.76E-01	0	1.02E-07
	Stdev	0	7.46E-05	0	7.66E+01	9.85E+19	0	4.89E-03	0	1.35E+02	7.31E+27	0	2.24E-02	0	2.15E-08
Xin-She Yang 3, D = 20	Mean	0	2.00E+00	2.00E+00	0	1.00E+00	1.97E+00	2.00E+00	0	2.00E+00	2.00E+00	2.00E+00	2.22E-04	2.00E+00	2.00E+00
	Best	0	2.00E+00	2.00E+00	0	1.00E+00	1.06E+00	2.00E+00	0	2.00E+00	2.00E+00	2.00E+00	7.00E-05	2.00E+00	2.00E+00
	Worst	0	2.00E+00	2.00E+00	1.11E-08	1.00E+00	2.00E+00	2.00E+00	0	2.00E+00	2.00E+00	2.00E+00	4.80E-04	2.00E+00	2.00E+00
	Stdev	0	1.71E-06	0	0	0	1.71E-01	3.08E-04	0	0	0	0	9.89E-05	2.18E-04	0
Zakharov, D = 20	Mean	0	4.54E-06	4.02E+00	1.65E-03	3.25E+03	0	3.00E-06	1.97E+01	1.36E+02	0	0	3.52E+00	0	0
	Best	0	1.21E-07	0	5.71E-05	6.40E-06	0	1.80E-06	2.05E-01	0	0	0	8.53E-01	0	0
	Worst	0	2.97E-05	1.20E+02	6.72E-03	5.67E+04	0	5.11E-06	6.37E+01	3.40E+02	0	0	7.12E+00	0	0
	Stdev	0	7.53E-06	2.20E+01	1.56E-03	1.24E+04	0	7.94E-07	1.67E+01	8.63E+01	0	0	1.69E+00	0	0

separable to nonseparable, and dimensions of 2, 5, 10, and 20.

- The optimization algorithms chosen for comparison include a variety of metaheuristic, SI-based, and nature-inspired algorithms, both classical (GA, PSO, FA, CSA, BA) and more recent ones (FPA, MFO, WOA, BOA, IMODE, CHOA, WFS, TSA). Due to space and time constraints, we only limit our experiments to these 13 algorithms.
- All algorithms are implemented in MATLAB R2020a on a computer with Intel(R) Core(TM) i5-1035G1 CPU @ 1.00 GHz 1.19 GHz, 8.00 GB RAM, and Windows 10 OS.

The source codes of PEOA are available online [82].

D. RESULTS OF PERFORMANCE COMPARISON AND ANALYSIS

For brevity, we only present here the numerical results for functions with dimension $D = 20$. Results for functions with dimensions $D = 2, 5, 10$ can be found in the Appendix.

Tables 6, 8, 7, and 9 provide the average (mean), best, and worst function value errors as well as the standard deviations obtained for functions with dimension $D = 20$ using the different examined algorithms. The cells having a value of 0 are highlighted in green for emphasis.

Figures 6, 7, 8, and 9 present the boxplots for functions with dimension $D = 20$. The boxplots show the function value error $|f_{\text{true}} - f(x^*)|$, where f_{true} is the true function value and x^* is the obtained optimal solution of the corresponding algorithm labeled on the bottom axis. For better illustration purposes, all values less than or equal to 10^{-8} are treated as 10^{-8} in the boxplots. Also, the logarithmic scale is used to accommodate a wide range of values.

Tables 13, 12, 11, and 10 illustrate the p-values from Wilcoxon's rank-sum test with 5% significance level. The p-values greater than or equal to 0.05 are shown in boldface. The "NaN" in these results, also in boldface, indicates that no significant difference between the algorithms can be concluded using Wilcoxon's rank-sum test [16].

Figure 10 shows the average number of function evaluations taken by the different examined algorithms for each

TABLE 8. Average, best, and worst function value errors and standard deviations over 30 independent runs obtained by the Philippine Eagle optimization algorithm compared to the 13 other examined algorithms for *5 different multimodal and separable functions* of dimension 20.

Function		PEOA	GA	PSO	FPA	BA	CS	FA	WOA	MFO	BOA	CHOA	WFS	TSA	IMODE
Alpine 1, D = 20	Mean	0	1.52E-04	0	6.01E-01	5.47E+00	1.85E-01	6.45E-02	0	0	1.53E-05	0	4.24E+00	1.20E+01	0
	Best	0	3.07E-05	0	0	7.89E-01	0	1.04E-04	0	0	0	0	1.44E+00	5.98E+00	0
	Worst	0	3.87E-04	0	1.42E+00	9.65E+00	5.75E-01	2.52E-01	0	0	3.23E-04	0	9.38E+00	1.76E+01	0
	Stdev	0	1.06E-04	0	5.30E-01	3.04E+00	1.84E-01	7.89E-02	0	0	6.10E-05	0	1.84E+00	3.04E+00	0
Wavy, D = 20	Mean	0	8.40E-02	3.95E-01	2.58E-01	5.80E-01	7.54E-02	3.66E-01	0	3.41E-01	5.61E-01	0	2.48E-05	4.31E-01	0
	Best	0	4.44E-02	2.35E-01	1.87E-01	3.19E-01	3.55E-02	2.21E-01	0	1.30E-01	4.94E-01	0	1.26E-05	2.67E-01	0
	Worst	0	1.34E-01	6.55E-01	3.29E-01	8.18E-01	1.30E-01	5.17E-01	0	4.87E-01	6.19E-01	0	5.01E-05	5.88E-01	0
	Stdev	0	2.31E-02	1.25E-01	3.90E-02	1.08E-01	2.03E-02	6.71E-02	0	9.45E-02	3.05E-02	0	7.67E-06	7.30E-02	0
Qing, D = 20	Mean	0	3.47E-07	0	5.47E+01	2.55E+10	8.67E+09	2.92E-01	6.62E-01	0	1.53E+02	8.68E+02	1.70E+03	9.55E+01	0
	Best	0	8.39E-08	0	4.53E-02	1.93E+09	6.45E+00	1.22E-01	5.77E-02	0	8.56E+01	3.41E+02	6.39E+02	7.02E+00	0
	Worst	0	6.72E-07	0	8.01E+02	7.41E+10	1.00E+10	8.77E-01	3.96E+00	0	2.10E+02	1.36E+03	2.44E+03	5.45E+02	0
	Stdev	0	1.16E-07	0	1.66E+02	1.88E+10	3.44E+09	1.80E-01	1.04E+00	0	2.61E+01	2.58E+02	5.28E+02	1.32E+02	0
Rastrigin, D = 20	Mean	0	6.63E-02	2.79E+01	1.83E+01	8.59E+01	9.89E+00	2.53E+01	0	7.84E+01	5.68E+01	2.18E-01	4.04E-03	7.66E+01	0
	Best	0	5.88E-07	8.95E+00	1.30E+01	2.39E+01	3.44E+00	1.09E+01	0	2.98E+01	0	0	1.89E-03	3.92E+01	0
	Worst	0	9.95E-01	5.27E+01	3.63E+01	1.74E+02	1.68E+01	6.77E+01	0	1.76E+02	1.10E+02	3.17E+00	1.08E-02	1.24E+02	0
	Stdev	0	2.52E-01	1.18E+01	4.62E+00	3.86E+01	3.41E+00	1.11E+01	0	3.44E+01	5.02E+01	7.05E-01	2.01E-03	1.96E+01	0
Xin-She Yang 1, D = 20	Mean	6.93E-08	3.70E-05	4.84E-01	8.81E-03	2.00E+06	5.57E-06	1.17E-05	0	8.49E+02	1.34E-06	0	5.40E-05	1.17E-03	3.54E-04
	Best	0	7.58E-08	0	1.13E-05	4.57E+00	0	2.11E-06	0	0	1.33E-08	0	6.63E-07	4.30E-06	7.23E-06
	Worst	1.24E-06	2.48E-04	3.28E+00	1.01E-01	4.27E+07	1.45E-04	3.99E-05	0	1.36E+04	8.60E-06	0	3.19E-04	9.33E-03	4.32E-03
	Stdev	2.41E-07	5.85E-05	1.02E+00	2.11E-02	8.24E+06	2.64E-05	8.35E-06	0	2.73E+03	2.06E-06	0	6.71E-05	2.15E-03	7.83E-04

TABLE 9. Average, best, and worst function value errors and standard deviations over 30 independent runs obtained by the Philippine Eagle optimization algorithm compared to the 13 other examined algorithms for *5 different multimodal and nonseparable functions* of dimension 20.

Function		PEOA	GA	PSO	FPA	BA	CS	FA	WOA	MFO	BOA	CHOA	WFS	TSA	IMODE
Ackley, D = 20	Mean	0	7.07E-05	1.93E-01	2.89E+00	1.71E+01	0	4.03E-03	0	3.02E+00	0	0	9.25E-02	1.39E+00	1.36E-07
	Best	0	3.60E-05	0	1.42E+00	1.49E+01	0	3.22E-03	0	0	0	0	5.85E-02	0	2.93E-08
	Worst	0	1.11E-04	2.45E+00	4.03E+00	1.90E+01	0	4.59E-03	0	1.87E+01	0	0	1.47E-01	3.62E+00	4.59E-07
	Stdev	0	2.09E-05	6.14E-01	7.26E-01	1.06E+00	0	3.71E-04	0	6.50E+00	0	0	2.10E-02	1.63E+00	1.10E-07
Periodic, D = 20	Mean	0	1.00E-01	4.88E-01	1.40E-01	1.00E-01	1.11E-01	1.00E-01	5.19E-02	1.57E+00	1.87E+00	3.25E+00	1.02E-04	9.25E-01	6.81E-02
	Best	0	1.00E-01	1.00E-01	1.08E-01	1.00E-01	1.04E-01	1.00E-01	0	1.00E-01	1.18E+00	2.31E+00	3.00E-05	5.05E-01	0
	Worst	0	1.00E-01	3.53E+00	1.75E-01	1.00E-01	1.16E-01	1.00E-01	1.10E-01	3.65E+00	2.30E+00	3.70E+00	2.00E-04	1.54E+00	1.00E-01
	Stdev	0	5.12E-08	9.89E-01	1.69E-02	5.79E-07	2.86E-03	3.57E-07	5.28E-02	1.01E+00	2.92E-01	3.41E-01	4.14E-05	2.25E-01	4.65E-02
Griewank, D = 20	Mean	0	6.07E-03	2.01E-02	4.56E-02	3.01E+00	0	1.41E-03	2.83E-04	1.33E-01	0	9.58E-03	1.00E-04	6.58E-03	0
	Best	0	1.92E-08	0	3.46E-06	1.16E+00	0	9.85E-06	0	0	0	0	2.84E-05	0	0
	Worst	0	7.89E-02	8.34E-02	1.72E-01	6.07E+00	0	1.48E-02	8.50E-03	3.21E+00	0	7.35E-02	1.83E-04	3.25E-02	0
	Stdev	0	1.70E-02	2.12E-02	3.89E-02	1.03E+00	0	3.75E-03	1.55E-03	5.81E-01	0	1.92E-02	4.32E-05	8.23E-03	0
Salomon, D = 20	Mean	6.66E-03	4.07E-01	4.57E-01	1.52E+00	1.29E+01	2.47E-01	2.00E-01	1.03E-01	1.79E+00	2.79E-01	9.99E-02	2.54E-01	2.10E-01	3.24E-01
	Best	0	3.00E-01	2.00E-01	6.00E-01	8.80E+00	2.00E-01	9.99E-02	0	8.00E-01	2.00E-01	9.99E-02	9.99E-02	9.99E-02	5.02E-08
	Worst	9.99E-02	6.00E-01	1.50E+00	2.60E+00	2.10E+01	4.00E-01	3.00E-01	2.00E-01	6.50E+00	3.00E-01	9.99E-02	4.45E-01	3.00E-01	5.00E-01
	Stdev	2.53E-02	7.85E-02	2.65E-01	4.65E-01	2.77E+00	5.71E-02	2.63E-02	4.90E-02	1.18E+00	3.96E-02	0	8.95E-02	4.81E-02	1.16E-01
Xin-She Yang 4, D = 20	Mean	0	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	8.67E-01	1.00E+00	1.00E+00	1.00E+00	1.39E-02	1.00E+00	1.00E+00
	Best	0	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	0	1.00E+00	1.00E+00	1.00E+00	7.28E-03	1.00E+00	1.00E+00
	Worst	0	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	2.54E-02	1.00E+00	1.00E+00
	Stdev	0	0	1.84E-08	0	0	0	0	3.46E-01	1.32E-08	0	0	4.40E-03	0	0

dimension $D = 2, 5, 10,$ and 20 when the stopping criterion is satisfied. The averages are computed over the 30 independent runs of each test function and the 20 test functions per dimension.

From these results, we see that PEOA obtained the most number of solutions with errors less than 10^{-8} among all the 14 examined algorithms found in Tables 6, 8, 7, and 9. Also, most of the optimal solutions that PEOA found for the different functions of dimension 20 are close to the true optimal solutions. While PEOA did not attain values less than the tolerance for the Xin-She Yang 1, Salomon, and Schwefel 2.21 functions, its obtained values for these functions are still relatively small.

Moreover, the boxplots in Figures 6, 7, 8, and 9 further validate the superior performance of PEOA among the examined algorithms. The boxplots corresponding to PEOA are generally thin and placed at 10^{-8} for almost all functions, indicating that the errors obtained by PEOA are consistently small. In particular, PEOA shows highly competitive results for the Schwefel 2.21, Periodic, Rosenbrock, Xin-She Yang 3, and Xin-She Yang 4 functions.

The p-values in Tables 13, 12, 11, and 10 show that we can find statistically significant differences between the results obtained by PEOA and the other optimization algorithms in almost all the experiments. Out of 1120 p-values, only 78 of them are either NaN or ≥ 0.05, so the obtained p-values of PEOA are good by over 93%. This result confirms that PEOA has performed remarkably better than the other algorithms.

We also see in Figure 10 that for all the different dimensions of functions tested, PEOA used the least average number of function evaluations until the error tolerance of 10^{-8} is reached. PEOA thus fared well in comparison with the other examined algorithms in terms of the speed and cost function value. This computationally inexpensive feature of PEOA can be attributed to its heavy exploitation technique, depicted through its regular and intensive local food search.

IV. APPLICATION TO REAL-WORLD OPTIMIZATION PROBLEMS

In the previous section, we have shown how PEOA is an efficient global optimization algorithm through various benchmark tests. We now present two applications that PEOA has effectively solved.

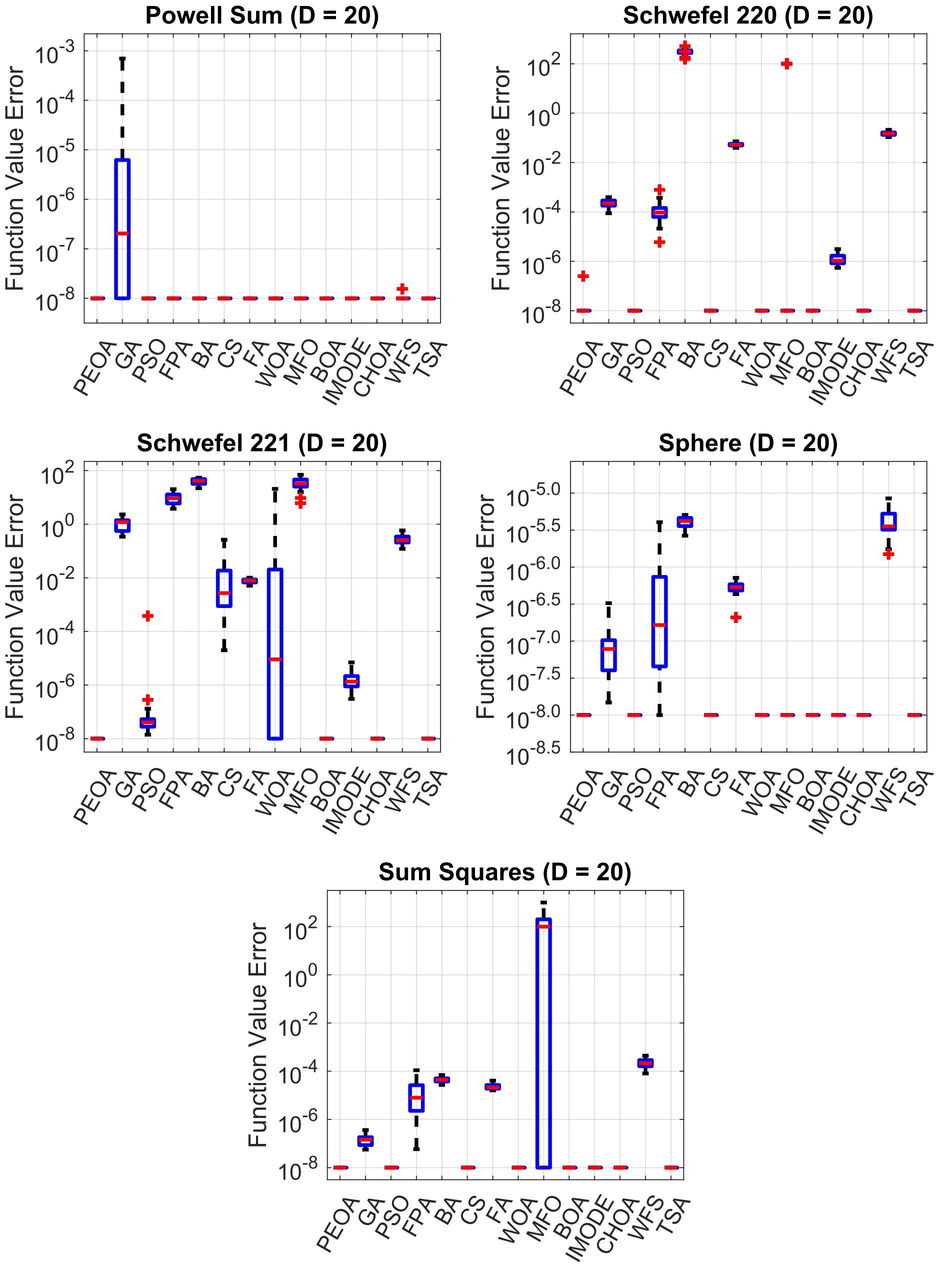


FIGURE 6. Boxplots over 30 independent runs (in logarithmic scale) of the function value errors obtained by the Philippine Eagle optimization algorithm and the 13 other examined algorithms for *5 unimodal and separable functions* with 20 dimensions.

A. ELECTRICAL IMPEDANCE TOMOGRAPHY

Electrical Impedance Tomography (EIT) is a non-invasive imaging technique that reconstructs the conductivity distribution of an object using electric currents. EIT has gained great interest for research due to its affordability, portability, and as a radiation-free imaging technique [83]–[93]. In particular, the main application of EIT is (continuous) lung monitoring in medical imaging [83], [85]. In this work, PEOA is applied to solve the inverse conductivity problem of EIT using the Complete Electrode Model (CEM), which is the most accurate and commonly used model for EIT.

EIT as a mathematical problem is divided into two parts: the forward and the inverse problem. The forward problem is where the data acquisition is made, that is, it computes for the voltages at the electrodes given a current pattern and the conductivity distribution inside the object. Let $\Omega \subset \mathbb{R}^d,\ d = 2, 3$ be bounded with a smooth boundary $\partial\Omega$. Let a set of patches $e_\ell \subset \partial\Omega,\ \ell = 1, 2, \dots, L$, where $L \in \mathbb{N}$, be the mathematical model of disjoint contact electrodes. Denote $I_\ell \in \mathbb{R}$ the current injected on the ℓth electrode and suppose that the current pattern $I = (I_\ell)_\ell$ satisfies the conservation of charge, i.e., $\sum_{\ell=1}^{L} I_\ell = 0$.

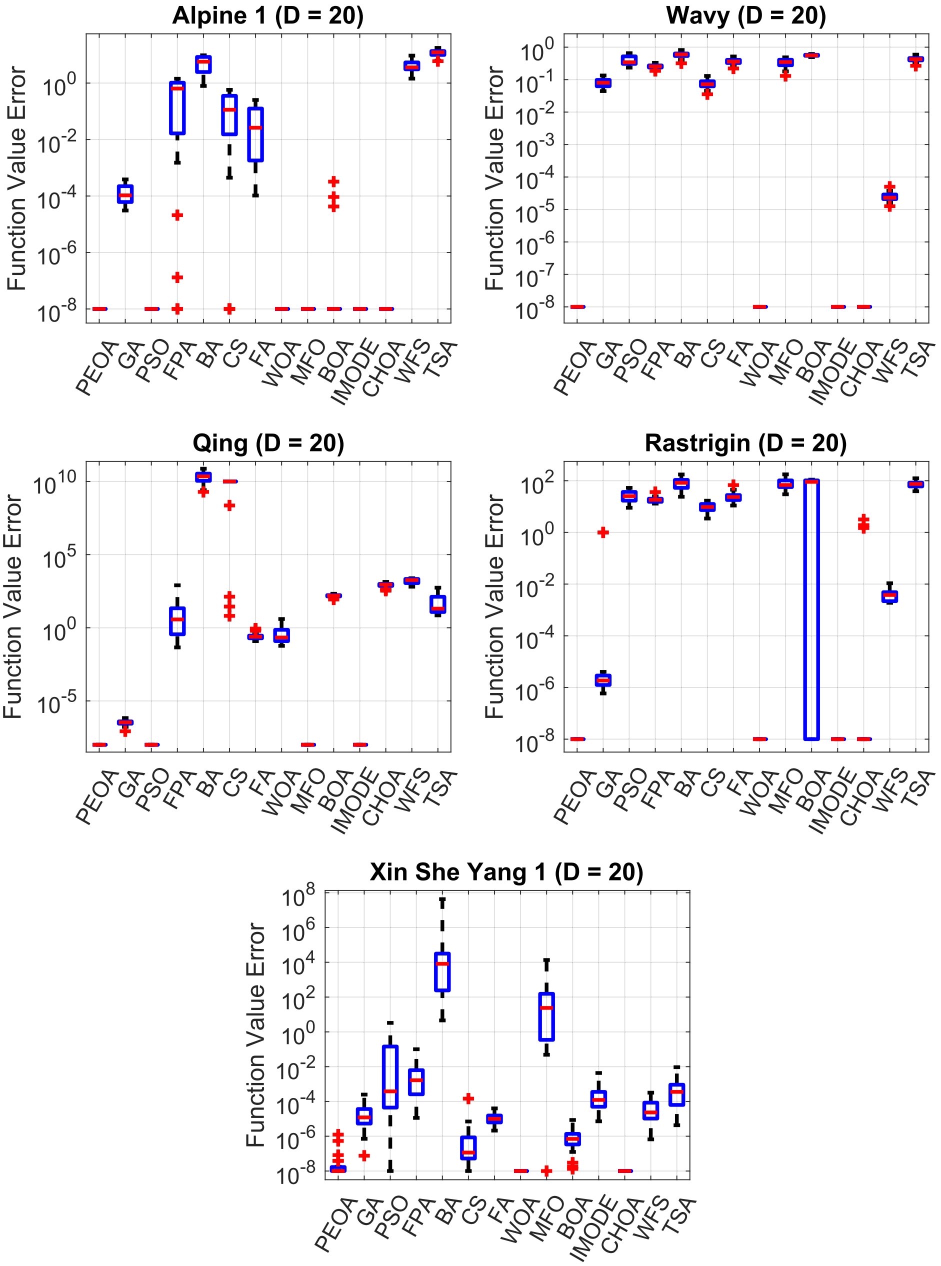


FIGURE 7. Boxplots over 30 independent runs (in logarithmic scale) of the function value errors obtained by the Philippine Eagle optimization algorithm and the 13 other examined algorithms for *5 multimodal and separable functions* with 20 dimensions.

The effective contact impedance is denoted by $Z \in \mathbb{R}^L$, where $Z = (z_\ell)_\ell$, $\ell = 1, \dots, L$ and $z_\ell > z_{\min}$, for some positive constant $z_{\min}$. Moreover, the conductivity distribution $\sigma \in L^\infty(\Omega)$ is assumed to satisfy $0 < \sigma_{\min} \le \sigma(x) \le \sigma_{\max} < +\infty$, for some constants $\sigma_{\min}, \sigma_{\max}$. Let $u \in H^1(\Omega)$ be the potential inside the domain and the measured voltages at the electrodes be $U = (U_\ell)_\ell$ which satisfies the arbitrary choice of ground, that is, $\sum_{\ell=1}^{L} U_\ell = 0$.

The CEM forward problem for EIT is: given current pattern I and conductivity distribution σ, find potentials (u, U) such that

$$\begin{cases} \nabla \cdot (\sigma \nabla u) = 0, & \text{in } \Omega, \qquad (12) \\ u + z_\ell \sigma \partial_{\vec{n}} = U_\ell, & \text{on } e_\ell,\ \ell = 1, 2, \dots, L, \qquad (13) \\ \sigma \dfrac{\partial u}{\partial \vec{n}} = 0, & \text{on } \partial\Omega \setminus \Gamma_e, \qquad (14) \\ \displaystyle\int_{e_\ell} \sigma \frac{\partial u}{\partial \vec{n}}\, ds = I_\ell, & \ell = 1, 2, \dots, L. \qquad (15) \end{cases}$$

To learn more on the background of the equations, see [94], [95]. The existence and uniqueness of the solution of the

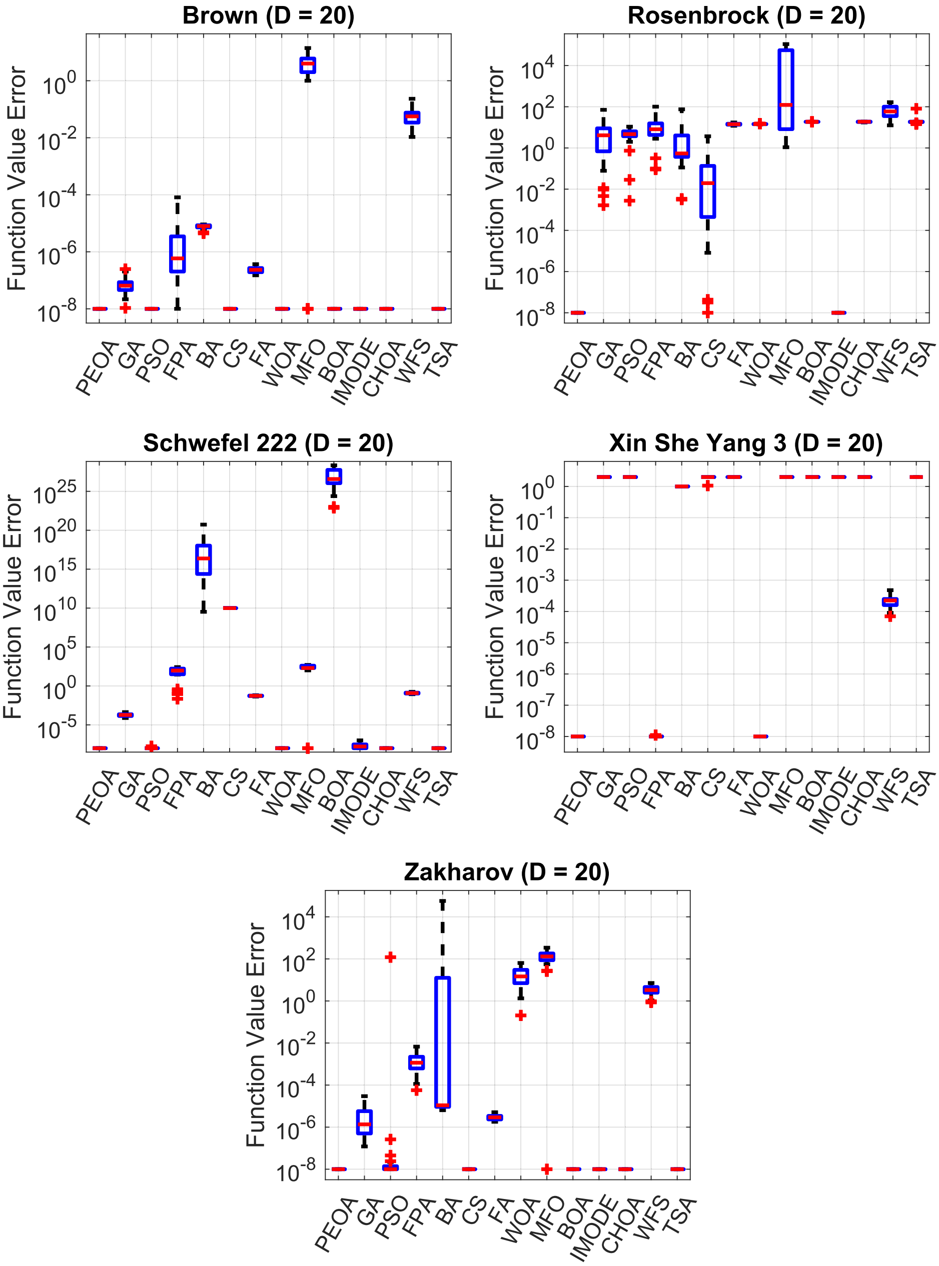


FIGURE 8. Boxplots over 30 independent runs (in logarithmic scale) of the function value errors obtained by the Philippine Eagle optimization algorithm and the 13 other examined algorithms for *5 unimodal and nonseparable functions* with 20 dimensions.

forward problem are proven in [95]. The discussion of the numerical solution and sensitivity analysis of the forward problem can be found in [96].

Meanwhile, the inverse problem reconstructs the conductivity distribution given the voltage measurements on the electrodes. First, we assume that σ is piecewise constant, i.e., $\sigma(x) = \sum_{i=0}^{N} \sigma_i \chi_i(x)$, $x \in \Omega$, where σ_0 is the background conductivity, $\chi_0(\mathbf{x})$ is the characteristic function of the background domain $\Omega_0 = \Omega \setminus \bigcup_{i=1}^{N} \Omega_i$, N corresponds to the number of (possible) inclusions Ω_i ($i = 1, \ldots, N$) in Ω, $\chi_i(x) = 1$ if $x \in \Omega_i$ and 0 otherwise.

Our goal is to retrieve the N inclusions of different conductivities in Ω. More precisely, we want to estimate vectors $P \in \mathbb{R}^m$ and $S \in \mathbb{R}^N$ iteratively. P contains the geometric attributes (e.g., center, side length) of the inclusions Ω_i, $i = 1, \ldots, N$ and S has the respective conductivity σ_i for each inclusion, $i = 1, 2, \ldots, N$ such that the error between the observed voltages and that predicted by the CEM forward

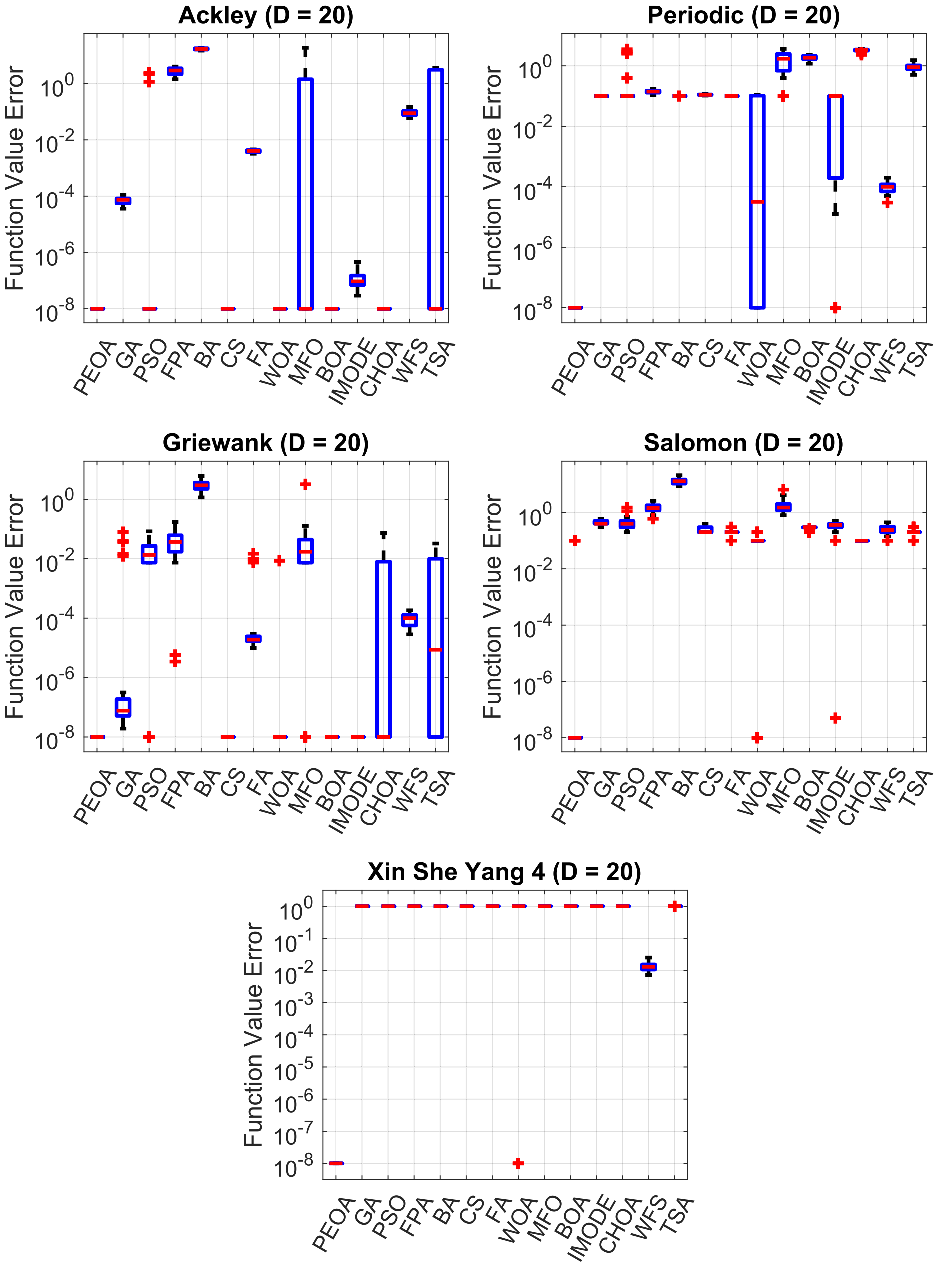


FIGURE 9. Boxplots over 30 independent runs (in logarithmic scale) of the function value errors obtained by the Philippine Eagle optimization algorithm and the 13 other examined algorithms for ***5 multimodal and nonseparable functions*** with 20 dimensions.

problem is minimized. Now, the inverse conductivity problem of EIT can be formulated as an optimization problem with the following objective function:

$$C(P, S) = \|U(P, S) - U_{\text{obs}}\|_2^2.$$

The voltages $U(P, S)$ are determined by solving the CEM forward problem (12) − (13) − (14) − (15) at a fixed conductivity σ and U_{obs} is the observed voltage at the electrodes, and $\|\cdot\|_2$ is the Euclidean norm.

Because of the importance of EIT in various fields, numerous approaches in solving the inverse problem can be found in the literature [97]–[102]. Several meta-heuristic algorithms were applied to the EIT inverse conductivity problem and produced promising results [94], [103]–[105]. We show how PEOA can also effectively solve the EIT inverse problem.

In this paper, we consider a disk domain with one elliptical inclusion. In particular, we aim to find the value of unknowns, that is, σ_e the conductivity of the inclusion, (h, k) the center of the ellipse, and the lengths of the major and minor axes, a and b, respectively. The conductivity σ_0 of the background medium is known and equal to 1.0. We work with synthetic data generated by setting the conductivity of the elliptical

TABLE 10. ***p*-values of Wilcoxon rank-sum statistical test with 5% significance of the Philippine Eagle optimization algorithm compared to the 13 other examined algorithms for *5 different unimodal and separable functions* of dimensions $D = 2, 5, 10, 20$. The *p*-values greater than or equal to 0.05 are shown in boldface. "NaN" in these results, also in boldface, indicates that no significant difference between the algorithms can be concluded using Wilcoxon's rank-sum test.**

Function	D	GA	PSO	FPA	BA	CS	FA	WOA	MFO	BOA	ABC	IMODE	CHOA	WFS	TSA
Powell Sum	2	1.21E-05	6.12E-10	3.02E-11	3.34E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	1.46E-10	5.07E-10	1.69E-09	3.02E-11
	5	2.76E-02	3.02E-11	4.13E-03	8.95E-11	3.67E-03	1.11E-04	8.88E-06	1.25E-04	2.37E-10	1.95E-10	**7.85E-02**	3.18E-03	**4.20E-01**	1.75E-05
	10	1.68E-04	1.44E-06	1.43E-05	5.36E-11	2.96E-05	6.53E-08	8.88E-06	7.74E-06	8.46E-10	3.02E-11	3.99E-04	8.29E-06	2.15E-02	5.09E-06
	20	6.12E-10	**1.71E-01**	6.68E-11	2.96E-11	7.37E-10	3.01E-11	6.12E-10	1.33E-10	3.01E-11	3.02E-11	1.31E-08	4.83E-10	2.49E-06	1.21E-10
Schwefel 2.20	2	3.02E-11	6.06E-11	1.14E-03	3.02E-11	1.52E-03	3.02E-11	7.68E-08	9.77E-04	3.02E-11	3.02E-11	8.31E-03	3.34E-03	3.02E-11	3.01E-07
	5	3.02E-11	1.77E-03	8.88E-10	3.02E-11	5.97E-09	3.02E-11	1.55E-09	1.12E-09	3.02E-11	3.02E-11	8.51E-07	6.01E-08	3.02E-11	3.82E-10
	10	3.02E-11	4.28E-02	3.34E-11	3.02E-11	4.97E-11	3.02E-11	2.61E-10	4.50E-11	4.07E-11	3.02E-11	1.96E-10	1.33E-10	3.02E-11	4.07E-11
	20	3.02E-11	5.21E-09	3.02E-11	3.02E-11	5.56E-10	3.02E-11	5.56E-10	3.42E-10	5.53E-10	3.02E-11	3.02E-11	5.57E-10	3.02E-11	5.57E-10
Schwefel 2.21	2	3.02E-11	6.06E-11	**2.14E-01**	3.02E-11	3.85E-02	3.02E-11	1.03E-02	**9.19E-02**	3.02E-11	3.02E-11	**2.55E-01**	**6.41E-01**	3.02E-11	1.89E-04
	5	3.02E-11	5.09E-06	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	4.50E-11	3.02E-11	3.02E-11	3.02E-11
	10	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.01E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
	20	3.01E-11	3.01E-11	3.01E-11	3.01E-11	3.01E-11	3.01E-11	3.01E-11	3.01E-11	2.96E-11	3.01E-11	3.01E-11	3.00E-11	3.01E-11	3.01E-11
Sphere	2	3.02E-11	1.39E-06	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	1.53E-05
	5	3.02E-11	4.50E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	1.25E-07
	10	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.01E-11	3.01E-11	3.02E-11	3.01E-11	3.01E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	4.50E-11
	20	3.02E-11	3.02E-11	3.02E-11	3.01E-11	3.01E-11	3.02E-11	3.02E-11	3.01E-11	3.01E-11	3.02E-11	2.32E-04	3.02E-11	3.02E-11	5.49E-11
Sum Squares	2	8.10E-10	2.15E-02	3.02E-11	3.02E-11	3.02E-11	3.69E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	7.38E-11	3.02E-11	3.02E-11	3.02E-11
	5	3.02E-11	4.07E-11	3.01E-11	3.00E-11	3.01E-11	3.01E-11	3.01E-11	3.02E-11	3.01E-11	3.02E-11	3.02E-11	3.01E-11	3.02E-11	3.02E-11
	10	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.01E-11	3.02E-11	3.02E-11	3.01E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
	20	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.01E-11	3.01E-11	2.89E-11	3.01E-11	3.02E-11	9.21E-05	3.02E-11	3.02E-11	3.02E-11

TABLE 11. ***p*-values of Wilcoxon rank-sum statistical test with 5% significance of the Philippine Eagle optimization algorithm compared to the 13 other examined algorithms for *5 different multimodal and separable functions* of dimensions $D = 2, 5, 10, 20$. The *p*-values greater than or equal to 0.05 are shown in boldface. "NaN" in these results, also in boldface, indicates that no significant difference between the algorithms can be concluded using Wilcoxon's rank-sum test.**

Function	D	GA	PSO	FPA	BA	CS	FA	WOA	MFO	BOA	ABC	IMODE	CHOA	WFS	TSA
Alpine 1	2	1.95E-09	**NaN**	**NaN**	1.21E-12	**NaN**	**NaN**	**NaN**	**NaN**	**3.34E-01**	**NaN**	**NaN**	**NaN**	1.21E-12	8.87E-07
	5	1.21E-12	**NaN**	4.19E-02	1.21E-12	**NaN**	**NaN**	**NaN**	**NaN**	4.79E-08	**NaN**	**NaN**	**NaN**	1.21E-12	1.21E-12
	10	1.21E-12	**NaN**	3.45E-07	1.21E-12	**1.61E-01**	2.21E-06	**NaN**	**NaN**	5.85E-09	4.57E-12	**NaN**	**NaN**	1.21E-12	1.21E-12
	20	1.21E-12	**3.34E-01**	1.21E-12	1.21E-12	1.93E-10	1.21E-12	**NaN**	4.19E-02	**8.15E-02**	1.21E-12	**NaN**	**NaN**	1.21E-12	1.21E-12
Wavy	2	5.95E-11	1.20E-03	4.43E-11	2.82E-11	5.40E-11	4.43E-11	4.43E-11	3.55E-11	2.97E-11	2.97E-11	1.19E-10	5.41E-11	4.43E-11	2.97E-11
	5	2.77E-11	7.57E-11	3.02E-11	3.01E-11	3.01E-11	2.73E-11	3.01E-11	2.75E-11	3.02E-11	3.02E-11	3.01E-11	3.02E-11	3.02E-11	3.02E-11
	10	2.43E-11	3.00E-11	3.01E-11	3.01E-11	3.01E-11	3.00E-11	3.01E-11	3.01E-11	3.01E-11	3.01E-11	3.01E-11	3.01E-11	3.01E-11	3.01E-11
	20	2.93E-11	2.97E-11	2.97E-11	2.97E-11	2.97E-11	2.97E-11	2.97E-11	2.97E-11	2.97E-11	2.97E-11	1.16E-05	2.97E-11	2.97E-11	2.97E-11
Qing	2	2.92E-09	**1.62E-01**	3.69E-11	3.02E-11	1.78E-10	3.02E-11	4.50E-11	9.91E-11	3.02E-11	3.02E-11	1.21E-10	3.02E-11	3.02E-11	3.02E-11
	5	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.01E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	6.82E-03	3.02E-11	3.02E-11	3.02E-11
	10	3.01E-11	3.01E-11	3.01E-11	3.01E-11	3.01E-11	3.01E-11	3.01E-11	3.01E-11	3.01E-11	3.01E-11	**5.01E-01**	3.01E-11	3.01E-11	3.01E-11
	20	3.00E-11	3.01E-11	3.01E-11	3.01E-11	4.09E-12	3.01E-11	3.01E-11	3.00E-11	3.01E-11	3.00E-11	1.94E-09	3.01E-11	3.01E-11	3.01E-11
Rastrigin	2	1.69E-10	**8.41E-01**	1.30E-10	3.97E-11	4.92E-09	8.83E-11	5.48E-10	2.51E-09	2.96E-11	2.96E-11	1.99E-08	3.01E-09	5.39E-11	6.57E-11
	5	1.20E-12	1.04E-12	1.21E-12	1.19E-12	1.21E-12	9.76E-13	1.21E-12	1.05E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12
	10	1.21E-12	1.05E-12	1.21E-12	1.21E-12	1.21E-12	1.15E-12	1.21E-12	1.19E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12
	20	1.21E-12	1.20E-12	1.21E-12	1.21E-12	1.21E-12	1.20E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12
Xin-She Yang 1	2	1.78E-10	**9.35E-01**	3.47E-10	3.02E-11	4.50E-11	5.96E-09	3.34E-03	1.61E-06	3.02E-11	3.02E-11	1.09E-05	3.16E-05	4.31E-08	1.46E-10
	5	8.99E-11	**2.52E-01**	2.15E-10	3.02E-11	3.02E-11	7.38E-11	2.44E-03	4.94E-05	3.02E-11	3.02E-11	3.09E-06	2.51E-05	3.02E-11	7.38E-11
	10	2.23E-09	3.78E-02	6.05E-07	3.02E-11	1.43E-08	6.07E-11	2.98E-02	**3.26E-01**	6.72E-10	3.02E-11	**7.01E-02**	2.37E-02	3.69E-11	4.57E-09
	20	5.23E-11	2.67E-09	3.02E-11	3.02E-11	3.21E-08	3.02E-11	**8.88E-01**	8.99E-11	2.03E-09	3.02E-11	3.02E-11	**6.31E-01**	3.34E-11	3.02E-11

inclusion to be 6.7. The number of electrodes is $L = 32$ and the contact impedance is set to be constant across all electrodes with $z_\ell = 0.03$. The first current applied to the electrodes has the form $I^1 = \{I^1_\ell\}_{\ell=0}^{L-1}$, with $I_\ell = \sin(\frac{2\pi\ell}{L})$ and we obtained the fifteen more current patterns by 'rotating' the values of the first current pattern for a total of sixteen current patterns. A 1% random (additive) noise is added to the voltage data as $U_{\text{data}} = (1 + 0.01 \cdot \text{rand}(L)) \cdot U$ to model the error obtained from the EIT experiments. In our simulations, one noise seed comprises sixteen different noise vectors added to the corresponding sixteen current-voltage measurements. The algorithm is applied for 20 independent runs with the same noise seed for all the runs, and a a maximum number of function evaluations (6 000) as the stopping criterion.

The results obtained by PEOA in solving the inverse conductivity problem of EIT are shown in Table 14 and Figure 11. We observed that the algorithm approximated the conductivity value, the center, and the shape of elliptical inclusion quite well, but was less accurate in approximating one of the axis lengths and angle of rotation.

B. ESTIMATING PARAMETERS OF A PENDULUM-MASS-SPRING-DAMPER SYSTEM

Given a mathematical model of a system, some of its parameter values might be unknown. While one can search for some of the parameters from the literature, the others need to be estimated. Parameter identification is a minimization problem that solves for the parameters of the model that will best fit the available data. Depending on the problem, various techniques on estimating parameters of models can be found in the literature [106]–[112].

TABLE 12. ***p*-values of Wilcoxon rank-sum statistical test with 5% significance of the Philippine Eagle optimization algorithm compared to the 13 other examined algorithms for *5 different unimodal and nonseparable functions* of dimensions $D = 2, 5, 10, 20$. The *p*-values greater than or equal to 0.05 are shown in boldface. “NaN” in these results, also in boldface, indicates that no significant difference between the algorithms can be concluded using Wilcoxon’s rank-sum test.**

Function	D	GA	PSO	FPA	BA	CS	FA	WOA	MFO	BOA	ABC	IMODE	CHOA	WFS	TSA
Brown	2	3.02E-11	2.38E-07	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
	5	3.02E-11	1.62E-09	3.02E-11	3.01E-11	3.01E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	1.46E-10	3.02E-11	3.02E-11	3.02E-11
	10	3.02E-11	3.34E-11	3.01E-11	3.02E-11	3.01E-11	3.02E-11	3.02E-11	3.00E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
	20	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.01E-11	3.02E-11	3.02E-11	2.88E-11	3.00E-11	3.02E-11	1.11E-07	3.02E-11	3.02E-11	3.02E-11
Rosenbrock	2	3.08E-11	5.62E-11	1.12E-10	3.77E-11	4.60E-11	5.62E-11	3.08E-11	2.52E-11	2.52E-11	2.52E-11	2.68E-10	2.52E-11	2.52E-11	2.52E-11
	5	2.95E-11	2.95E-11	2.95E-11	2.95E-11	2.95E-11	2.95E-11	2.95E-11	2.95E-11	2.95E-11	2.95E-11	2.95E-11	2.95E-11	2.95E-11	2.92E-11
	10	2.13E-11	2.13E-11	2.13E-11	2.13E-11	2.13E-11	2.13E-11	2.13E-11	2.13E-11	2.13E-11	2.13E-11	5.51E-03	2.13E-11	2.13E-11	2.13E-11
	20	2.60E-11	2.60E-11	2.60E-11	2.60E-11	2.60E-11	2.60E-11	2.60E-11	2.60E-11	2.60E-11	2.60E-11	**4.58E-01**	2.59E-11	2.60E-11	2.59E-11
Schwefel 2.22	2	3.01E-11	3.02E-11	**1.10E-01**	3.02E-11	1.95E-02	3.02E-11	4.13E-03	1.35E-02	3.02E-11	3.02E-11	**9.63E-02**	**1.22E-01**	3.02E-11	1.53E-05
	5	3.02E-11	**7.85E-02**	3.02E-11	3.02E-11	1.25E-07	3.02E-11	2.78E-07	1.25E-07	3.02E-11	3.02E-11	2.86E-05	9.53E-07	3.02E-11	1.25E-07
	10	3.02E-11	2.39E-04	3.02E-11	3.02E-11	3.02E-11	3.02E-11	9.30E-10	2.93E-11	6.04E-11	3.02E-11	4.74E-06	1.33E-10	3.02E-11	4.50E-11
	20	3.02E-11	9.53E-07	3.02E-11	3.02E-11	1.21E-12	3.02E-11	2.26E-10	2.43E-11	3.02E-11	3.02E-11	5.48E-11	6.05E-11	3.02E-11	5.49E-11
Xin-She Yang 3	2	**NaN**	**NaN**	**NaN**	5.59E-05	**NaN**	**NaN**	**NaN**	**NaN**	4.57E-12	5.69E-11	**NaN**	**NaN**	**NaN**	**NaN**
	5	5.19E-07	1.69E-14	**NaN**	1.69E-14	1.31E-07	2.10E-08	**NaN**	1.69E-14	2.36E-05	1.21E-12	**NaN**	1.69E-14	1.21E-12	4.16E-14
	10	1.17E-13	1.69E-14	**NaN**	1.69E-14	6.49E-13	3.37E-13	**3.34E-01**	1.69E-14	1.69E-14	1.69E-14	1.69E-14	1.69E-14	1.19E-12	5.30E-13
	20	2.71E-14	1.69E-14	**NaN**	1.69E-14	2.71E-14	1.18E-12	**NaN**	1.69E-14	1.69E-14	1.19E-12	1.69E-14	1.69E-14	1.17E-12	1.08E-12
Zakharov	2	9.75E-10	4.08E-05	3.34E-11	5.49E-11	3.69E-11	3.69E-11	3.34E-11	4.07E-11	3.02E-11	3.02E-11	5.49E-11	3.33E-11	3.02E-11	3.02E-11
	5	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.01E-11	3.01E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
	10	3.02E-11	3.02E-11	3.01E-11	3.02E-11	3.01E-11	3.02E-11	3.00E-11	3.01E-11	3.00E-11	3.02E-11	2.33E-09	3.02E-11	3.02E-11	3.02E-11
	20	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.01E-11	3.02E-11	3.02E-11	3.02E-11	3.01E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11

TABLE 13. ***p*-values of Wilcoxon rank-sum statistical test with 5% significance of the Philippine Eagle optimization algorithm compared to the 13 other examined algorithms for *5 different multimodal and nonseparable functions* of dimensions $D = 2, 5, 10, 20$. The *p*-values greater than or equal to 0.05 are shown in boldface. “NaN” in these results, also in boldface, indicates that no significant difference between the algorithms can be concluded using Wilcoxon’s rank-sum test.**

Function	D	GA	PSO	FPA	BA	CS	FA	WOA	MFO	BOA	ABC	IMODE	CHOA	WFS	TSA
Ackley	2	3.02E-11	4.07E-11	6.10E-03	2.93E-11	1.08E-02	3.02E-11	5.71E-04	3.15E-02	3.02E-11	3.02E-11	1.08E-02	4.68E-02	3.02E-11	3.37E-05
	5	3.02E-11	**1.41E-01**	3.01E-11	3.02E-11	3.01E-11	3.02E-11	3.01E-11	3.33E-11	3.02E-11	3.02E-11	2.37E-10	3.01E-11	3.02E-11	3.02E-11
	10	3.02E-11	3.01E-11	3.01E-11	3.02E-11	3.01E-11	3.02E-11	3.02E-11	3.01E-11	3.01E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
	20	3.02E-11	3.02E-11	3.00E-11	3.02E-11	3.01E-11	3.02E-11	3.02E-11	3.01E-11	3.01E-11	3.02E-11	3.02E-11	3.01E-11	3.02E-11	3.02E-11
Periodic	2	2.35E-07	2.92E-08	3.43E-07	1.17E-13	1.45E-04	1.10E-02	5.59E-05	2.10E-08	1.45E-04	1.20E-12	**NaN**	4.77E-08	**NaN**	1.09E-08
	5	1.69E-14	1.60E-13	9.23E-13	1.69E-14	1.69E-14	1.17E-13	2.16E-06	1.18E-13	1.21E-12	1.21E-12	**NaN**	4.57E-12	4.41E-12	1.21E-12
	10	1.69E-14	4.16E-14	1.21E-12	1.69E-14	1.13E-12	1.69E-14	4.74E-08	1.00E-12	1.21E-12	1.21E-12	**NaN**	4.57E-12	1.16E-12	1.21E-12
	20	1.69E-14	1.20E-13	1.21E-12	1.69E-14	1.20E-12	1.69E-14	1.27E-05	1.14E-12	1.21E-12	1.21E-12	2.42E-10	1.21E-12	1.17E-12	1.21E-12
Griewank	2	2.98E-11	3.66E-11	3.00E-11	2.97E-11	3.00E-11	2.85E-11	2.78E-11	2.24E-11	3.00E-11	3.00E-11	3.00E-11	3.00E-11	3.00E-11	3.00E-11
	5	3.02E-11	3.01E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11
	10	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	9.51E-06	3.02E-11	3.02E-11	3.02E-11
	20	3.02E-11	2.99E-11	3.02E-11	3.02E-11	3.01E-11	3.02E-11	3.01E-11	3.01E-11	3.02E-11	3.02E-11	**2.52E-01**	3.02E-11	3.02E-11	3.02E-11
Salomon	2	1.56E-11	3.31E-03	3.02E-11	2.97E-11	3.02E-11	3.02E-11	3.76E-08	1.77E-09	3.02E-11	3.02E-11	1.78E-04	1.14E-04	3.02E-11	4.11E-12
	5	2.58E-11	1.87E-11	4.57E-12	3.01E-11	4.57E-12	4.57E-12	6.20E-12	3.58E-11	3.32E-11	3.02E-11	1.23E-09	9.10E-12	3.07E-11	8.15E-12
	10	1.95E-11	1.75E-11	1.75E-11	2.99E-11	2.36E-12	1.21E-12	1.27E-11	2.81E-11	2.94E-11	3.02E-11	2.36E-12	1.21E-12	3.01E-11	1.34E-11
	20	1.96E-11	2.63E-11	2.97E-11	3.01E-11	1.75E-11	2.63E-12	1.08E-10	2.95E-11	2.43E-11	3.02E-11	4.12E-11	1.66E-11	3.02E-11	9.53E-12
Xin-She Yang 4	2	4.63E-08	**NaN**	5.58E-03	1.19E-12	4.60E-08	1.08E-12	2.16E-02	4.19E-02	1.21E-12	1.21E-12	**NaN**	**8.15E-02**	1.20E-12	3.45E-07
	5	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	5.85E-09	3.87E-13	1.21E-12	1.21E-12	**NaN**	1.21E-12	1.21E-12	1.21E-12
	10	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	4.57E-12	1.12E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12
	20	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.93E-10	1.15E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12	1.21E-12

As an application of POEA, we present an approach in identifying the parameters of a pendulum system model. This model involves a neutral delay differential equation (NDDE), which is a differential equation with delay both in state and the derivative. NDDEs have been used in modeling various applications in science and engineering [113]–[119].

In this work, we consider a Pendulum-Mass-Spring-Damper (PMSD) system consisting of a mass M mounted on a linear spring. Attached to the spring via a hinged rod of length l is a pendulum of mass m [120]. The angular deflection of the pendulum from the downward position is assumed to be negligible. The parameter C is the damping coefficient. Furthermore, it is assumed that external force does not act on the system. This mechanical system can be modeled using the following delay differential equation of neutral type

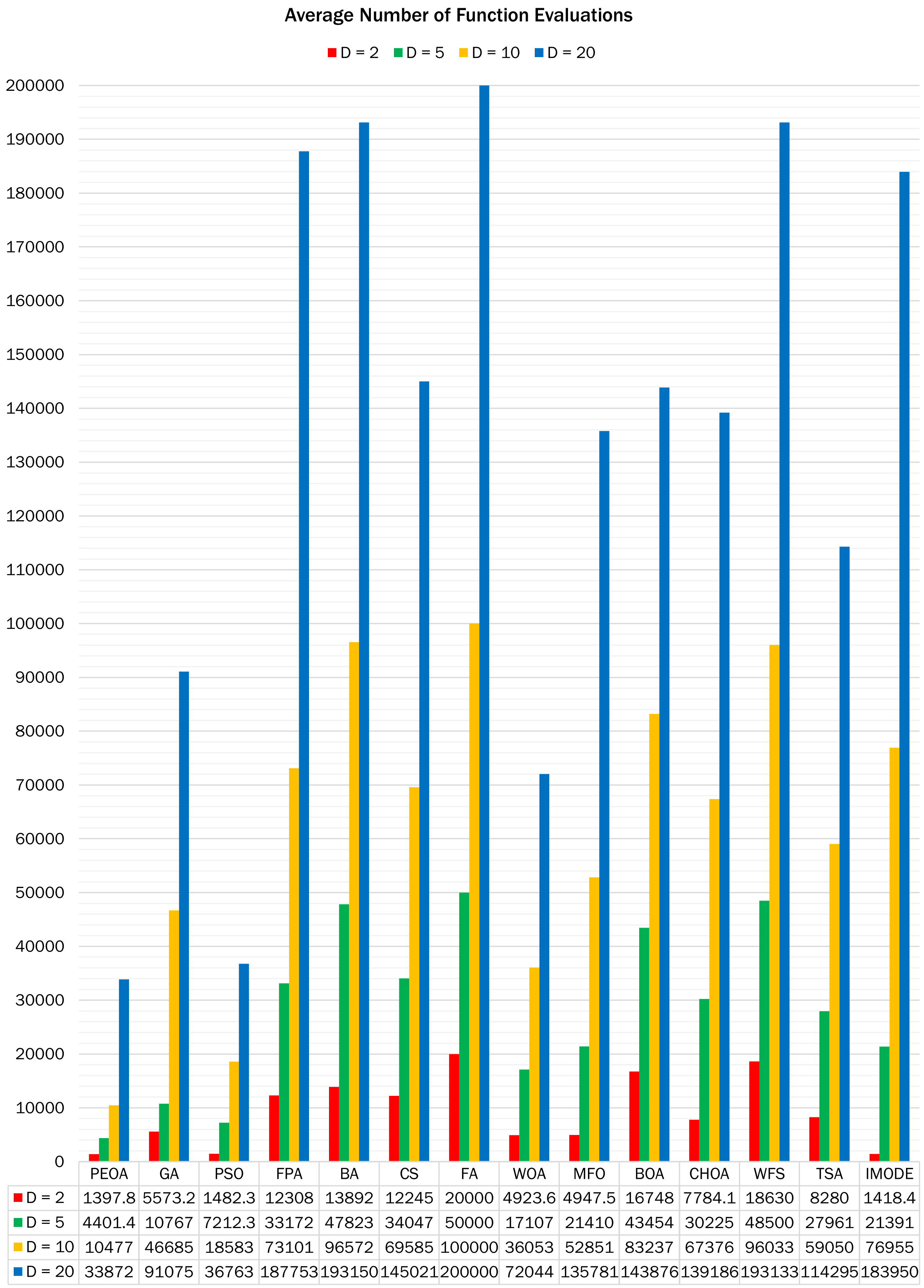
$$M\ddot{x}(t) + C\dot{x}(t) + Kx(t) + m\ddot{x}(t-\tau) = 0. \tag{16}$$

Here, K and C denote the stiffness and damping coefficients, respectively. The position, velocity, and acceleration of the system at a given time t are represented by the quantities $x(t)$, $\dot{x}(t)$, and $\ddot{x}(t)$, respectively. By dividing both sides of (16) by M, we obtain the following modified equivalent equation

$$\ddot{y} + 2\zeta\dot{y} + y + p\ddot{y}(t-\tau) = 0. \tag{17}$$

For this model, the history function is given by $\phi(t) = \cos(t/2)$ [120].

The parameters of (17) are estimated from a set of simulated noisy data, which are generated in two steps. First, the following parameter values from [120] are used to solve (17): $\tau = 1$, $\zeta = 0.05$, and $p = 0$. Secondly, the noisy data $y^*(t_i)$, $i = 1, 2, \ldots, n$ are generated by assuming a normal distribution, with the standard deviation equal to 10% of the standard deviation of the computed solution of the model [112]. For this study, we set $n = 50$. We find the



	PEOA	GA	PSO	FPA	BA	CS	FA	WOA	MFO	BOA	CHOA	WFS	TSA	IMODE
🟥 D = 2	1397.8	5573.2	1482.3	12308	13892	12245	20000	4923.6	4947.5	16748	7784.1	18630	8280	1418.4
🟩 D = 5	4401.4	10767	7212.3	33172	47823	34047	50000	17107	21410	43454	30225	48500	27961	21391
🟨 D = 10	10477	46685	18583	73101	96572	69585	100000	36053	52851	83237	67376	96033	59050	76955
🟦 D = 20	33872	91075	36763	187753	193150	145021	200000	72044	135781	143876	139186	193133	114295	183950

FIGURE 10. **Average number of function evaluations of the Philippine Eagle optimization algorithm upon reaching a function value error of 10^{-8} compared to the 13 other examined algorithms. This is over 30 independent runs of each test function with 20 test functions per dimension.**

TABLE 14. Final solutions and their corresponding relative error (|*truevalue*−*avevalue*|/|*truevalue*|) generated by PEOA for the EIT inverse conductivity problem in a disk domain with one elliptical inclusion. Note that $7\pi/8 \approx 2.74889357$.

Parameter	σ_e	h	k	a	b	θ
bounds	$[5, 9]$	$[-1, 1]$	$[-1, 1]$	$[0, 2]$	$[0, 2]$	$[0, \pi]$
true value	6.7	-0.4	0.5	0.7	0.4	$7\pi/8$
ave. value	6.963	-0.399	0.484	0.689	0.449	2.583
rel. error	3.9E-02	1.4E-03	3.0E-02	1.4E-02	1.2E-01	6.0E-02

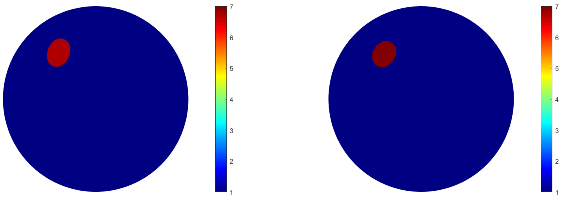


FIGURE 11. Left: true conductivity distribution. Right: reconstructed conductivity distribution (mean of the 20 approximate solutions).

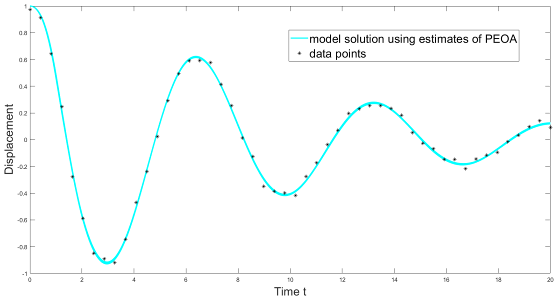


FIGURE 12. Plots of the solution curves (cyan) to the pendulum model in (17) using the estimated parameters obtained by PEOA.

TABLE 15. Estimated values of the parameters of the pendulum model.

Parameter	τ	ζ	p
bounds	[0.5,5]	[0,1]	[0,1]
true value	1	0.05	0.2
ave. value	1.007	0.051	0.205
rel. error	7E-03	2E-02	2.5E-03

minimum of least-squares error formulation given by

$$\min_{\theta \in \mathbb{R}^3} \frac{\sum_{i=1}^{50} \left(y_i^\star - y_\theta(t_i)\right)^2}{\sum_{i=1}^{50} \left(y_i^\star\right)^2},$$

where θ is the parameter vector containing the triple τ, ζ, and p. We denote $y_\theta(t_i)$ as the model solution at time t_i given θ.

Because PEOA is probabilistic, we run the algorithm 20 times independently. This way, we can gauge the accuracy and consistency of the solutions obtained. The results are presented in Figure 12 and Table 15. We can see that all the 20 obtained estimates are close to the true solution. The different plots of the $y(t)$ using the estimated parameters fit the simulated data well. Furthermore, the relative errors of the calculated parameters are all less than 2%.

V. CONCLUSION

This work proposes a novel, meta-heuristic, and nature-inspired optimization algorithm called the Philippine Eagle Optimization Algorithm. It is an algorithm that is inspired by the hunting behavior of the Philippine Eagle and uses three different global operators for its exploration strategy. It also has an intensive local search every iteration, contributing to its strong exploitation ability.

Twenty optimization test functions of varying properties on modality, separability, and dimension were solved using PEOA, and the results were compared to those obtained by 13 other optimization algorithms. PEOA was also applied to two real-world optimization problems: the inverse conductivity problem in Electrical Impedance Tomography (EIT) and parameter estimation in a Pendulum-Mass-Spring-Damper system (PMSD) involving neutral delay differential equations.

Results show that PEOA effectively solves the different benchmark tests implemented in this work. The algorithm outperforms the other examined algorithms in terms of accuracy and precision in finding the optimal solution of the tested functions. PEOA also uses the least number of function evaluations compared to the other algorithms, indicating that it employs a computationally inexpensive optimization process. Such a feature of PEOA is due to its heavy exploitation technique. Furthermore, PEOA can provide good results for the six unknowns in the EIT problem and gives proper estimates for the parameters involved in the PMSD model.

We emphasize that PEOA gave better results than IMODE in solving the test functions chosen in this paper. This is a significant highlight because IMODE ranked first in the CEC 2020 Competition on Single Objective Bound Constrained Numerical Optimization [67]. Since certain aspects of PEOA were derived from IMODE and its several source algorithms, PEOA can be considered a further improved version of these algorithms.

Therefore, PEOA is a competitive algorithm that can be applied to a variety of functions and problems while keeping the number of function evaluations at a minimum. It shows promising features in comparison to the other optimization algorithms selected. It also highlights the distinctive characteristics of the national bird of the Philippines, the Philippine Eagle, which could hopefully initiate conservation efforts for the critically endangered bird.

Future research will consider more modifications of PEOA that can further improve its performance. This includes thoroughly studying the sensitivity of the parameters of PEOA and tuning the parameters based on specific kinds of problems. Experimentation of PEOA to a broader scope

TABLE 16. Average, best, and worst function value errors and standard deviations over 30 independent runs obtained by the Philippine Eagle Optimization Algorithm compared to those obtained by the 13 other examined algorithms for the 20 different functions of varied types and having *dimension 2*.

Function		PEOA	GA	PSO	FPA	BA	CS	FA	WOA	MFO	BOA	CHOA	WFS	TSA	IMODE
Powell Sum, D = 2	Mean	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Best	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Worst	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Stdev	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Schwefel 2.20, D = 2	Mean	0	8.47E-06	0	0	8.68E+00	0	3.64E-04	0	0	1.74E-06	0	1.98E-03	0	0
	Best	0	4.17E-07	0	0	2.41E-05	0	7.53E-05	0	0	3.83E-07	0	2.09E-04	0	0
	Worst	0	3.93E-05	0	0	4.26E+01	0	9.03E-04	0	0	4.04E-06	0	5.20E-03	0	0
	Stdev	0	8.65E-06	0	0	1.08E+01	0	1.97E-04	0	0	9.12E-07	0	1.57E-03	0	0
Schwefel 2.21, D = 2	Mean	0	6.74E-06	0	0	2.85E+00	0	2.59E-04	0	0	1.47E-06	0	1.06E-03	0	0
	Best	0	6.32E-07	0	0	4.23E-06	0	5.24E-05	0	0	1.70E-07	0	6.91E-05	0	0
	Worst	0	2.94E-05	0	0	1.32E+01	3.39E-08	5.97E-04	0	0	4.01E-06	0	3.20E-03	0	0
	Stdev	0	6.60E-06	0	0	3.27E+00	0	1.70E-04	0	0	8.94E-07	0	7.99E-04	0	0
Sphere, D = 2	Mean	0	0	0	0	0	0	0	0	0	0	0	1.73E-08	0	0
	Best	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Worst	0	0	0	0	0	0	0	0	0	0	0	1.23E-07	0	0
	Stdev	0	0	0	0	0	0	0	0	0	0	0	2.22E-08	0	0
Sum Squares, D = 2	Mean	0	0	0	0	0	0	0	0	0	0	0	9.26E-08	0	0
	Best	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Worst	0	0	0	0	0	0	0	0	0	0	0	1.38E-06	0	0
	Stdev	0	0	0	0	0	0	0	0	0	0	0	2.55E-07	0	0
Alpine 1, D = 2	Mean	0	3.34E-05	0	0	3.17E-03	0	0	0	0	5.93E-05	0	3.06E-05	3.04E-03	0
	Best	0	0	0	0	3.09E-07	0	0	0	0	0	0	1.25E-06	0	0
	Worst	0	3.60E-04	0	0	5.01E-02	0	0	0	0	1.78E-03	0	1.35E-04	2.19E-02	0
	Stdev	0	7.67E-05	0	0	1.18E-02	0	0	0	0	3.25E-04	0	3.62E-05	6.13E-03	0
Wavy, D = 2	Mean	0	4.14E-02	2.96E-03	0	3.63E-01	0	0	0	1.78E-02	2.33E-02	0	1.26E-07	0	0
	Best	0	0	0	0	0	0	0	0	0	1.17E-07	0	0	0	0
	Worst	0	1.78E-01	8.88E-02	0	5.89E-01	0	1.84E-08	0	8.88E-02	1.09E-01	0	6.83E-07	0	0
	Stdev	0	5.58E-02	1.62E-02	0	1.92E-01	0	0	0	3.61E-02	4.13E-02	0	1.72E-07	0	0
Qing, D = 2	Mean	0	0	0	1.12E-07	2.49E+06	0	2.14E-05	1.39E-07	0	1.59E-04	2.58E-04	3.64E-04	7.92E-06	0
	Best	0	0	0	0	4.32E+02	0	1.26E-06	0	0	5.91E-06	1.30E-05	1.21E-06	2.45E-07	0
	Worst	0	0	0	9.87E-07	2.98E+07	0	1.28E-04	2.41E-06	0	5.86E-04	9.01E-04	1.78E-03	3.24E-05	0
	Stdev	0	0	0	1.94E-07	5.79E+06	0	2.62E-05	4.80E-07	0	1.41E-04	2.11E-04	4.37E-04	7.98E-06	0
Rastrigin, D = 2	Mean	0	2.98E-01	6.63E-02	1.01E-07	4.31E+00	0	6.41E-08	0	3.32E-02	1.53E-01	0	1.80E-06	2.66E-01	0
	Best	0	0	0	0	0	0	0	0	0	8.18E-08	0	0	0	0
	Worst	0	1.99E+00	9.95E-01	1.57E-06	1.69E+01	0	2.90E-07	0	9.95E-01	1.26E+00	0	1.82E-05	1.99E+00	0
	Stdev	0	5.32E-01	2.52E-01	3.05E-07	4.14E+00	0	7.57E-08	0	1.82E-01	3.71E-01	0	3.48E-06	6.36E-01	0
Xin-She Yang 1, D = 2	Mean	7.95E-08	4.72E-05	1.41E-03	1.06E-05	1.10E-02	9.29E-06	2.06E-06	4.34E-05	0	4.34E-04	3.05E-06	3.50E-06	1.02E-03	0
	Best	0	7.33E-08	0	5.32E-08	2.71E-05	2.42E-07	2.37E-08	0	0	9.81E-06	0	5.42E-08	2.16E-08	0
	Worst	5.26E-07	7.69E-04	2.91E-02	4.62E-05	7.01E-02	4.29E-05	1.07E-05	8.41E-04	0	1.65E-03	6.38E-05	4.67E-05	7.32E-03	0
	Stdev	1.29E-07	1.40E-04	5.44E-03	1.18E-05	1.73E-02	9.47E-06	2.72E-06	1.72E-04	0	4.22E-04	1.25E-05	9.51E-06	1.67E-03	0
Brown, D = 2	Mean	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Best	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Worst	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Stdev	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Rosenbrock, D = 2	Mean	0	2.29E-01	3.22E-03	0	4.04E-01	2.43E-07	1.01E-08	5.83E-06	7.14E-02	1.01E-03	2.90E-04	2.85E-04	5.90E-02	0
	Best	0	0	0	0	0	0	0	0	3.40E-07	1.58E-05	6.50E-06	0	4.43E-07	0
	Worst	0	3.45E+00	5.20E-02	1.44E-08	3.61E+00	6.22E-06	5.44E-08	9.79E-05	7.93E-01	1.30E-02	1.52E-03	8.52E-03	1.00E+00	0
	Stdev	0	6.78E-01	1.05E-02	0	1.02E+00	1.13E-06	1.13E-08	1.92E-05	1.83E-01	2.33E-03	3.37E-04	1.56E-03	2.27E-01	0
Schwefel 2.22, D = 2	Mean	0	8.40E-06	0	0	1.55E+01	0	3.72E-04	0	0	1.26E-06	0	1.64E-03	0	0
	Best	0	4.25E-07	0	0	1.15E-05	0	8.98E-05	0	0	2.70E-07	0	1.19E-04	0	0
	Worst	0	5.32E-05	0	0	7.54E+01	0	9.05E-04	0	0	4.06E-06	0	4.93E-03	0	0
	Stdev	0	1.12E-05	0	0	2.03E+01	0	1.92E-04	0	0	1.01E-06	0	1.46E-03	0	0
Xin-She Yang 3, D = 2	Mean	0	0	0	0	4.33E-01	0	0	0	0	3.83E+00	0	0	0	0
	Best	0	0	0	0	0	0	0	0	0	4.45E+01	0	0	0	0
	Worst	0	1.01E-08	0	0	1.00E+00	0	0	0	0	1.83E-05	0	0	0	0
	Stdev	0	0	0	0	5.04E-01	0	0	0	0	1.13E+01	0	0	0	0
Zakharov, D = 2	Mean	0	0	0	0	0	0	0	0	0	0	0	3.10E-08	0	0
	Best	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Worst	0	0	0	0	0	0	0	0	0	0	0	1.71E-07	0	0
	Stdev	0	0	0	0	0	0	0	0	0	0	0	4.23E-08	0	0
Ackley, D = 2	Mean	0	8.60E-02	0	0	5.53E+00	0	2.60E-04	0	0	1.29E-04	0	1.10E-03	0	0
	Best	0	1.12E-06	0	0	1.73E-05	0	4.61E-05	0	0	8.42E-06	0	1.74E-04	0	0
	Worst	0	2.58E+00	0	4.21E-08	1.50E+01	6.45E-08	5.84E-04	0	0	6.20E-04	0	5.15E-03	0	0
	Stdev	0	4.71E-01	0	0	3.92E+00	1.09E-08	1.37E-04	0	0	1.40E-04	0	1.03E-03	0	0
Periodic, D = 2	Mean	0	6.33E-02	6.05E-02	5.53E-04	9.67E-02	1.89E-03	1.69E-02	4.33E-02	7.00E-02	2.20E-02	5.50E-02	0	7.34E-02	0
	Best	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Worst	0	1.00E-01	1.00E-01	5.44E-03	1.00E-01	5.46E-02	1.00E-01	1.00E-01	1.00E-01	1.01E-01	1.01E-01	0	1.00E-01	0
	Stdev	0	4.90E-02	4.93E-02	1.33E-03	1.83E-02	9.95E-03	3.78E-02	5.04E-02	4.66E-02	4.11E-02	4.46E-02	0	4.50E-02	0
Griewank, D = 2	Mean	0	3.06E-02	7.23E-03	8.97E-04	1.81E-01	6.70E-05	3.08E-03	2.22E-03	1.52E-02	1.99E-02	2.03E-03	1.23E-06	7.90E-03	0
	Best	0	0	0	1.20E-07	2.71E-02	2.09E-08	0	0	0	4.83E-06	0	1.24E-08	0	0
	Worst	0	2.66E-01	2.71E-02	7.40E-03	6.98E-01	5.33E-04	9.86E-03	7.40E-03	7.89E-02	7.42E-02	7.76E-03	1.21E-05	2.96E-02	0
	Stdev	0	5.69E-02	6.64E-03	1.88E-03	1.74E-01	1.24E-04	3.83E-03	3.45E-03	1.97E-02	1.65E-02	3.43E-03	2.34E-06	7.11E-03	0
Salomon, D = 2	Mean	0	1.73E-01	6.30E-02	3.50E-04	1.35E+00	4.82E-04	3.33E-05	3.66E-02	6.33E-02	3.34E-02	1.33E-02	1.85E-04	8.66E-02	0
	Best	0	3.39E-07	0	3.80E-06	5.00E-01	1.82E-06	2.70E-06	0	0	7.51E-05	0	1.07E-05	0	0
	Worst	0	7.00E-01	9.99E-02	3.88E-03	3.00E+00	1.95E-03	8.49E-05	9.99E-02	9.99E-02	1.00E-01	9.99E-02	6.11E-04	9.99E-02	0
	Stdev	0	1.57E-01	4.71E-02	7.18E-04	5.74E-01	4.92E-04	1.90E-05	4.90E-02	4.90E-02	4.22E-02	2.94E-02	1.57E-04	3.45E-02	0
Xin-She Yang 4, D = 2	Mean	0	4.00E-01	0	1.34E-05	8.00E-01	3.95E-05	6.60E-04	1.67E-01	1.33E-01	1.40E-01	6.60E-02	1.35E-04	6.33E-01	0
	Best	0	4.50E-07	0	1.29E-08	5.96E-06	3.92E-07	6.30E-06	0	0	2.67E-04	0	1.00E-05	0	0
	Worst	0	1.00E+00	0	1.23E-04	1.00E+00	2.06E-04	1.86E-02	1.00E+00	1.00E+00	1.00E+00	7.92E-01	4.40E-04	1.00E+00	0
	Stdev	0	4.98E-01	0	3.35E-05	4.07E-01	5.42E-05	3.39E-03	3.79E-01	3.46E-01	3.14E-01	2.05E-01	1.25E-04	4.90E-01	0

TABLE 17. Average, best, and worst function value errors and standard deviations over 30 independent runs obtained by the Philippine Eagle optimization algorithm compared to those obtained by the 13 other examined algorithms for the 20 different functions of varied types and having *dimension 5*.

Function		PEOA	GA	PSO	FPA	BA	CS	FA	WOA	MFO	BOA	CHOA	WFS	TSA	IMODE
Powell Sum, D = 5	Mean	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Best	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Worst	0	1.09E-07	0	0	0	0	0	0	0	0	0	0	0	0
	Stdev	0	1.97E-08	0	0	0	0	0	0	0	0	0	0	0	0
Schwefel 2.20, D = 5	Mean	0	3.07E-05	0	0	5.51E+01	0	5.52E-03	0	0	2.48E-07	0	6.64E-02	0	0
	Best	0	9.01E-06	0	0	6.00E+00	0	2.68E-03	0	0	6.41E-08	0	1.85E-02	0	0
	Worst	0	9.11E-05	0	0	1.09E+02	0	7.78E-03	0	0	6.17E-07	0	1.05E-01	0	0
	Stdev	0	1.72E-05	0	0	3.09E+01	0	1.24E-03	0	0	1.22E-07	0	2.26E-02	0	0
Schwefel 2.21, D = 5	Mean	0	3.41E-01	0	0	2.34E+01	0	1.81E-03	1.36E-08	0	1.02E-06	0	1.19E-01	0	0
	Best	0	5.52E-06	0	0	8.18E+00	0	8.24E-04	0	0	3.71E-07	0	2.85E-02	0	0
	Worst	0	1.54E+00	0	0	5.26E+01	0	2.53E-03	1.79E-07	0	1.96E-06	0	3.17E-01	0	0
	Stdev	0	4.57E-01	0	0	1.11E+01	0	4.66E-04	3.13E-08	0	3.91E-07	0	5.84E-02	0	0
Sphere, D = 5	Mean	0	0	0	0	6.08E-08	0	2.63E-08	0	0	0	0	3.92E-06	0	0
	Best	0	0	0	0	0	0	0	0	0	0	0	1.69E-07	0	0
	Worst	0	0	0	0	1.15E-07	0	5.92E-08	0	0	0	0	1.03E-05	0	0
	Stdev	0	0	0	0	2.48E-08	0	1.15E-08	0	0	0	0	2.41E-06	0	0
Sum Squares, D = 5	Mean	0	0	0	0	1.93E-07	0	2.31E-07	0	0	0	0	3.79E-05	0	0
	Best	0	0	0	0	9.55E-08	0	4.41E-08	0	0	0	0	5.08E-06	0	0
	Worst	0	1.90E-08	0	0	3.74E-07	0	4.43E-07	0	0	0	0	1.21E-04	0	0
	Stdev	0	0	0	0	8.08E-08	0	1.17E-07	0	0	0	0	3.00E-05	0	0
Alpine 1, D = 5	Mean	0	2.25E-05	0	6.68E-05	4.59E-01	0	0	0	0	1.08E-03	0	4.25E-02	3.68E-01	0
	Best	0	1.34E-06	0	0	6.14E-05	0	0	0	0	0	0	7.15E-04	0	0
	Worst	0	1.46E-04	0	1.97E-03	1.93E+00	0	0	0	0	8.83E-03	0	2.58E-01	1.46E+00	0
	Stdev	0	3.22E-05	0	3.59E-04	6.37E-01	0	0	0	0	1.78E-03	0	5.49E-02	3.88E-01	0
Wavy, D = 5	Mean	0	1.13E-01	5.20E-02	5.97E-02	4.30E-01	2.53E-08	1.10E-01	0	1.46E-01	2.52E-01	0	8.23E-05	8.83E-02	0
	Best	0	0	0	4.94E-05	1.42E-01	0	3.55E-02	0	0	7.42E-02	0	1.14E-05	0	0
	Worst	0	2.52E-01	2.62E-01	1.12E-01	7.26E-01	2.53E-07	2.34E-01	0	3.61E-01	3.31E-01	0	2.13E-04	1.79E-01	0
	Stdev	0	6.06E-02	7.51E-02	3.25E-02	1.29E-01	5.64E-08	6.88E-02	0	9.80E-02	6.79E-02	0	5.64E-05	7.38E-02	0
Qing, D = 5	Mean	0	2.37E-08	0	5.65E-04	1.51E+09	1.82E-06	2.32E-03	1.88E-04	0	1.27E-01	5.03E-01	4.95E+00	7.74E-03	0
	Best	0	0	0	7.25E-05	2.78E+06	6.71E-08	3.18E-04	3.76E-06	0	2.33E-02	6.98E-02	2.18E-01	8.13E-04	0
	Worst	0	1.02E-07	0	3.29E-03	1.00E+10	7.91E-06	4.16E-03	9.01E-04	0	3.09E-01	1.20E+00	3.07E+01	1.86E-02	0
	Stdev	0	2.33E-08	0	6.44E-04	1.95E+09	1.88E-06	1.17E-03	2.56E-04	0	6.56E-02	3.37E-01	7.73E+00	4.02E-03	0
Rastrigin, D = 5	Mean	0	6.63E-01	1.30E+00	1.36E+00	1.46E+01	4.11E-05	2.42E+00	0	3.48E+00	9.04E+00	1.65E-01	1.74E-03	2.96E+00	0
	Best	0	0	0	3.74E-02	1.99E+00	1.27E-08	1.98E-06	0	9.95E-01	1.49E-07	0	3.04E-04	0	0
	Worst	0	9.95E+00	5.97E+00	2.36E+00	3.48E+01	5.03E-04	5.97E+00	0	1.19E+01	1.52E+01	2.96E+00	4.73E-03	5.04E+00	0
	Stdev	0	1.87E+00	1.47E+00	6.71E-01	9.90E+00	1.02E-04	1.49E+00	0	2.32E+00	2.97E+00	6.42E-01	1.04E-03	2.03E+00	0
Xin-She Yang 1, D = 5	Mean	1.70E-07	9.00E-05	2.21E-03	8.55E-06	3.75E-01	1.64E-05	5.04E-06	5.12E-06	0	6.85E-05	0	7.69E-05	2.10E-02	0
	Best	0	2.77E-07	0	1.20E-07	6.22E-03	1.30E-06	2.48E-07	0	0	1.23E-06	0	2.72E-06	2.55E-07	0
	Worst	1.11E-06	6.75E-04	5.80E-02	8.27E-05	1.18E+00	5.47E-05	2.06E-05	1.42E-04	0	4.97E-04	0	4.93E-04	2.64E-01	0
	Stdev	2.85E-07	1.67E-04	1.06E-02	1.53E-05	3.58E-01	1.61E-05	4.53E-06	2.58E-05	0	9.51E-05	0	1.14E-04	5.31E-02	0
Brown, D = 5	Mean	0	0	0	0	8.57E-08	0	0	0	0	0	0	5.12E-05	0	0
	Best	0	0	0	0	1.59E-08	0	0	0	0	0	0	8.48E-07	0	0
	Worst	0	0	0	0	1.56E-07	0	1.41E-08	0	0	0	0	4.08E-04	0	0
	Stdev	0	0	0	0	3.82E-08	0	0	0	0	0	0	8.78E-05	0	0
Rosenbrock, D = 5	Mean	0	8.96E-01	9.20E-01	1.18E-08	3.58E+01	4.64E-04	5.49E-02	3.58E-01	4.15E-01	5.22E-01	2.63E+00	1.98E+00	3.62E+00	0
	Best	0	8.57E-05	6.57E-04	0	6.32E-03	0	2.06E-02	7.13E-08	1.33E-04	7.92E-03	1.36E+00	1.98E-02	1.08E+00	0
	Worst	0	8.68E+00	5.57E+00	8.72E-08	4.69E+02	6.71E-03	1.25E-01	7.35E-01	5.41E+00	2.61E+00	3.10E+00	6.17E+00	6.32E+00	0
	Stdev	0	2.01E+00	1.74E+00	1.74E-08	1.03E+02	1.53E-03	2.33E-02	1.80E-01	1.04E+00	6.02E-01	5.04E-01	1.97E+00	9.82E-01	0
Schwefel 2.22, D = 5	Mean	0	3.15E-05	0	7.87E-03	1.46E+02	0	5.06E-03	0	0	3.45E-07	0	9.11E-02	0	0
	Best	0	5.23E-06	0	2.69E-08	2.42E+01	0	2.93E-03	0	0	1.30E-07	0	4.32E-02	0	0
	Worst	0	7.60E-05	0	1.90E-01	3.02E+02	0	7.51E-03	0	0	6.71E-07	0	1.69E-01	0	0
	Stdev	0	2.12E-05	0	3.50E-02	6.31E+01	0	1.36E-03	0	0	1.37E-07	0	3.28E-02	0	0
Xin-She Yang 3, D = 5	Mean	0	1.20E+00	2.00E+00	0	1.00E+00	8.30E-02	1.40E+00	0	2.00E+00	9.33E-01	2.00E+00	6.13E-04	2.00E+00	0
	Best	0	0	2.00E+00	0	1.00E+00	0	3.27E-08	0	2.00E+00	0	2.00E+00	1.00E-05	2.00E+00	0
	Worst	0	2.00E+00	2.00E+00	0	1.00E+00	1.90E+00	2.00E+00	0	2.00E+00	2.00E+00	2.00E+00	7.61E-03	2.00E+00	0
	Stdev	0	9.96E-01	0	0	0	3.48E-01	9.32E-01	0	0	1.01E+00	0	1.39E-03	4.31E-05	0
Zakharov, D = 5	Mean	0	0	0	0	5.36E-01	0	8.51E-08	0	0	0	0	9.61E-04	0	0
	Best	0	0	0	0	2.59E-08	0	1.45E-08	0	0	0	0	8.73E-06	0	0
	Worst	0	1.46E-08	0	0	1.61E+01	0	2.37E-07	0	0	0	0	5.42E-03	0	0
	Stdev	0	0	0	0	2.94E+00	0	4.92E-08	0	0	0	0	1.48E-03	0	0
Ackley, D = 5	Mean	0	7.73E-02	0	0	1.48E+01	0	1.75E-03	0	0	5.84E-07	0	1.03E-01	0	0
	Best	0	1.10E-05	0	0	8.77E+00	0	7.76E-04	0	0	3.45E-08	0	3.38E-02	0	0
	Worst	0	2.32E+00	0	0	1.89E+01	0	2.50E-03	0	0	1.74E-06	0	2.64E-01	0	0
	Stdev	0	4.23E-01	0	0	2.80E+00	0	4.33E-04	0	0	4.81E-07	0	5.41E-02	0	0
Periodic, D = 5	Mean	0	1.00E-01	1.12E-01	1.00E-01	1.00E-01	1.00E-01	9.67E-02	5.68E-02	1.49E-01	1.40E-01	1.52E-01	6.13E-05	1.08E-01	0
	Best	0	1.00E-01	1.00E-01	1.00E-01	1.00E-01	1.00E-01	9.99E-08	0	1.00E-01	1.10E-01	0	0	1.00E-01	0
	Worst	0	1.00E-01	2.32E-01	1.01E-01	1.00E-01	1.00E-01	1.00E-01	1.01E-01	3.96E-01	1.73E-01	2.67E-01	2.10E-04	1.24E-01	0
	Stdev	0	0	3.04E-02	1.55E-04	2.41E-08	9.69E-06	1.83E-02	5.05E-02	1.12E-01	1.67E-02	5.79E-02	4.45E-05	5.79E-03	0
Griewank, D = 5	Mean	0	6.57E-03	6.82E-02	5.15E-02	6.32E-01	1.69E-02	4.29E-02	1.97E-02	1.43E-01	2.81E-01	4.85E-02	7.83E-04	1.37E-01	0
	Best	0	0	9.86E-03	4.45E-03	1.13E-01	7.34E-04	7.40E-03	0	1.72E-02	8.09E-02	0	1.77E-04	2.33E-02	0
	Worst	0	4.19E-02	2.25E-01	1.04E-01	1.69E+00	3.35E-02	1.06E-01	2.39E-01	3.50E-01	4.33E-01	1.57E-01	3.16E-03	3.90E-01	0
	Stdev	0	1.35E-02	5.90E-02	2.44E-02	3.95E-01	6.92E-03	2.81E-02	4.80E-02	8.67E-02	7.85E-02	4.55E-02	6.28E-04	8.09E-02	0
Salomon, D = 5	Mean	3.33E-03	1.77E-01	1.07E-01	9.99E-02	5.09E+00	9.99E-02	9.99E-02	1.03E-01	1.77E-01	1.04E-01	9.96E-02	9.98E-02	1.07E-01	0
	Best	0	9.99E-02	9.99E-02	9.99E-02	1.30E+00	9.99E-02	9.99E-02	9.99E-02	9.99E-02	9.99E-02	9.14E-02	9.13E-02	9.99E-02	0
	Worst	9.99E-02	3.00E-01	2.00E-01	9.99E-02	8.50E+00	9.99E-02	9.99E-02	2.00E-01	5.00E-01	2.00E-01	9.99E-02	1.04E-01	2.00E-01	0
	Stdev	1.82E-02	6.26E-02	2.54E-02	0	1.55E+00	0	0	1.83E-02	9.71E-02	1.82E-02	1.54E-03	1.76E-03	2.54E-02	0
Xin-She Yang 4, D = 5	Mean	0	1.00E+00	1.00E+00	8.99E-01	1.00E+00	9.40E-01	9.62E-01	7.67E-01	1.00E+00	1.00E+00	1.00E+00	8.35E-03	1.00E+00	0
	Best	0	1.00E+00	1.00E+00	8.88E-02	1.00E+00	9.29E-02	1.38E-01	0	1.00E+00	1.00E+00	1.00E+00	3.10E-03	1.00E+00	0
	Worst	0	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.01E+00	1.00E+00	1.00E+00	1.34E-02	1.00E+00	0
	Stdev	0	0	0	2.64E-01	0	2.12E-01	1.60E-01	4.30E-01	1.16E-03	3.23E-04	3.11E-04	3.02E-03	1.20E-04	0

of optimization functions and algorithms, finding more real-world applications where PEOA can be used, comparing the results of PEOA in real-world applications with other algorithms, and creating versions of PEOA that can handle constrained or multi-objective optimization problems may also be considered.

TABLE 18. **Average, best, and worst function value errors and standard deviations over 30 independent runs obtained by the Philippine Eagle optimization algorithm compared to those obtained by the 13 other examined algorithms for the 20 different functions of varied types and having *dimension 10*.**

Function		PEOA	GA	PSO	FPA	BA	CS	FA	WOA	MFO	BOA	CHOA	WFS	TSA	IMODE
Powell Sum, D = 10	Mean	0	1.03E-07	0	0	0	0	0	0	0	0	0	0	0	0
	Best	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Worst	0	5.14E-07	0	0	0	0	0	0	0	0	0	0	0	0
	Stdev	0	1.63E-07	0	0	0	0	0	0	0	0	0	0	0	0
Schwefel 2.20, D = 10	Mean	0	5.42E-05	0	3.47E-08	1.62E+02	0	1.82E-02	0	0	1.12E-08	0	9.23E-02	0	0
	Best	0	1.45E-05	0	0	7.06E+01	0	1.15E-02	0	0	0	0	4.49E-02	0	0
	Worst	0	9.64E-05	0	1.93E-07	2.81E+02	0	2.70E-02	0	0	1.87E-08	0	1.37E-01	0	0
	Stdev	0	2.31E-05	0	4.86E-08	5.31E+01	0	3.46E-03	0	0	0	0	2.27E-02	0	0
Schwefel 2.21, D = 10	Mean	0	7.26E-02	0	2.95E-03	3.35E+01	0	4.03E-03	4.00E-08	3.70E-01	6.46E-08	0	2.89E-01	0	0
	Best	0	1.20E-05	0	1.64E-06	1.78E+01	0	2.77E-03	0	0	1.97E-08	0	1.17E-01	0	0
	Worst	0	3.79E-01	1.39E-08	3.15E-02	5.27E+01	0	5.45E-03	5.02E-07	4.42E+00	1.39E-07	0	4.58E-01	0	0
	Stdev	0	1.17E-01	0	6.97E-03	9.74E+00	0	5.66E-04	1.03E-07	9.85E-01	2.57E-08	0	8.17E-02	0	0
Sphere, D = 10	Mean	0	1.07E-08	0	0	6.60E-07	0	1.16E-07	0	0	0	0	3.82E-06	0	0
	Best	0	0	0	0	3.21E-07	0	5.84E-08	0	0	0	0	1.69E-06	0	0
	Worst	0	6.41E-08	0	0	9.98E-07	0	1.89E-07	0	0	0	0	7.61E-06	0	0
	Stdev	0	1.15E-08	0	0	1.98E-07	0	3.93E-08	0	0	0	0	1.69E-06	0	0
Sum Squares, D = 10	Mean	0	2.35E-08	0	0	3.41E-06	0	2.56E-06	0	0	0	0	1.21E-04	0	0
	Best	0	0	0	0	1.61E-06	0	1.05E-06	0	0	0	0	3.58E-05	0	0
	Worst	0	5.92E-08	0	0	5.72E-06	0	4.72E-06	0	0	0	0	2.81E-04	0	0
	Stdev	0	1.54E-08	0	0	1.03E-06	0	8.85E-07	0	0	0	0	5.73E-05	0	0
Alpine 1, D = 10	Mean	0	5.12E-05	0	4.04E-02	1.72E+00	0	2.49E-03	0	0	4.26E-03	0	5.03E-01	2.36E+00	0
	Best	0	4.44E-06	0	0	1.25E-02	0	0	0	0	0	0	8.07E-02	2.88E-01	0
	Worst	0	2.04E-04	0	2.77E-01	5.68E+00	0	2.43E-02	0	0	3.98E-02	0	1.43E+00	4.46E+00	0
	Stdev	0	4.81E-05	0	7.63E-02	1.29E+00	0	5.90E-03	0	0	9.25E-03	0	3.63E-01	1.12E+00	0
Wavy, D = 10	Mean	0	4.80E-02	1.41E-01	1.95E-01	5.03E-01	2.70E-02	2.96E-01	0	2.88E-01	4.33E-01	0	5.36E-05	1.79E-01	0
	Best	0	1.74E-08	1.78E-02	7.66E-02	2.86E-01	4.26E-04	1.55E-01	0	8.88E-02	3.26E-01	0	1.45E-05	0	0
	Worst	0	1.08E-01	4.20E-01	2.73E-01	7.63E-01	5.82E-02	4.81E-01	0	5.32E-01	5.01E-01	0	1.19E-04	3.73E-01	0
	Stdev	0	2.25E-02	1.04E-01	4.91E-02	1.18E-01	1.67E-02	7.52E-02	0	1.07E-01	5.25E-02	0	2.50E-05	7.70E-02	0
Qing, D = 10	Mean	0	8.11E-08	0	1.78E-02	8.20E+09	3.74E-03	2.57E-02	5.67E-02	0	6.48E+00	2.45E+01	1.47E+02	4.17E-01	0
	Best	0	1.18E-08	0	0	2.43E+08	1.21E-04	1.28E-02	6.58E-04	0	2.78E+00	9.31E+00	4.84E+01	1.51E-01	0
	Worst	0	1.97E-07	0	7.00E-02	2.41E+10	2.92E-02	3.95E-02	8.74E-01	0	1.10E+01	8.89E+01	3.34E+02	1.38E+00	0
	Stdev	0	5.25E-08	0	2.10E-02	5.65E+09	5.26E-03	5.97E-03	1.77E-01	0	1.91E+00	2.14E+01	9.72E+01	2.47E-01	0
Rastrigin, D = 10	Mean	0	4.54E-07	6.20E+00	7.64E+00	3.79E+01	2.06E+00	8.56E+00	0	2.21E+01	2.96E+01	4.30E-01	2.43E-03	1.80E+01	0
	Best	0	4.02E-08	9.95E-01	2.20E+00	1.09E+01	8.09E-02	1.99E+00	0	6.96E+00	0	0	7.48E-04	5.41E+00	0
	Worst	0	9.99E-07	1.59E+01	1.42E+01	7.16E+01	4.16E+00	2.29E+01	0	5.18E+01	4.60E+01	9.31E+00	7.51E-03	4.23E+01	0
	Stdev	0	2.70E-07	3.61E+00	2.84E+00	1.75E+01	9.96E-01	4.60E+00	0	1.24E+01	1.41E+01	1.80E+00	1.53E-03	8.71E+00	0
Xin-She Yang 1, D = 10	Mean	2.99E-07	1.34E-05	1.34E-02	1.87E-05	3.11E+01	4.19E-06	1.22E-05	0	8.23E-04	5.27E-06	6.60E-08	8.00E-05	4.88E-03	0
	Best	0	1.75E-07	0	0	7.45E-01	3.51E-08	1.29E-06	0	0	5.74E-07	0	3.51E-06	0	0
	Worst	4.62E-06	7.02E-05	3.62E-01	2.96E-04	2.60E+02	3.71E-05	3.53E-05	3.21E-08	1.33E-02	1.59E-05	1.69E-06	3.28E-04	6.09E-02	0
	Stdev	8.83E-07	1.79E-05	6.59E-02	6.06E-05	5.98E+01	7.50E-06	8.89E-06	0	2.98E-03	4.77E-06	3.07E-07	8.91E-05	1.35E-02	0
Brown, D = 10	Mean	0	0	0	0	1.21E-06	0	5.47E-08	0	2.00E-01	0	0	2.31E-03	0	0
	Best	0	0	0	0	5.89E-07	0	2.14E-08	0	0	0	0	2.53E-05	0	0
	Worst	0	2.76E-08	0	0	1.94E-06	0	9.67E-08	0	2.00E+00	0	0	1.36E-02	0	0
	Stdev	0	0	0	0	3.28E-07	0	2.04E-08	0	4.84E-01	0	0	3.66E-03	0	0
Rosenbrock, D = 10	Mean	0	2.29E+00	7.80E-01	2.66E-01	1.85E+01	9.02E-04	3.47E+00	4.51E+00	8.79E+01	8.20E+00	8.44E+00	1.63E+01	8.23E+00	0
	Best	0	6.02E-03	1.21E-02	0	6.70E-02	0	1.13E+00	4.04E+00	2.45E-04	6.77E+00	7.19E+00	9.89E-01	5.28E+00	0
	Worst	0	9.06E+00	4.40E+00	3.99E+00	2.50E+02	1.70E-02	7.87E+00	4.97E+00	2.51E+03	8.54E+00	8.96E+00	1.17E+02	9.48E+00	0
	Stdev	0	2.42E+00	1.39E+00	1.01E+00	5.04E+01	3.25E-03	1.35E+00	2.45E-01	4.57E+02	3.47E-01	5.84E-01	2.45E+01	1.06E+00	0
Schwefel 2.22, D = 10	Mean	0	5.39E-05	0	5.06E+01	1.68E+05	7.30E-02	1.85E-02	0	6.33E+01	0	0	1.45E-01	0	0
	Best	0	1.13E-05	0	3.59E-05	1.45E+02	0	1.12E-02	0	0	0	0	8.21E-02	0	0
	Worst	0	1.55E-04	0	2.31E+02	3.28E+06	1.03E+00	2.53E-02	0	2.00E+02	1.36E-08	0	2.44E-01	0	0
	Stdev	0	3.42E-05	0	5.47E+01	6.07E+05	2.52E-01	3.66E-03	0	6.69E+01	0	0	3.92E-02	0	0
Xin-She Yang 3, D = 10	Mean	0	1.93E+00	2.00E+00	0	1.00E+00	1.64E+00	2.00E+00	6.66E-02	2.00E+00	2.00E+00	2.00E+00	3.33E-04	2.00E+00	2.00E+00
	Best	0	3.67E-08	2.00E+00	0	1.00E+00	6.68E-07	2.00E+00	0	2.00E+00	2.00E+00	2.00E+00	1.10E-04	2.00E+00	2.00E+00
	Worst	0	2.00E+00	2.00E+00	0	1.00E+00	2.00E+00	2.00E+00	2.00E+00	2.00E+00	2.00E+00	2.00E+00	8.40E-04	2.00E+00	2.00E+00
	Stdev	0	3.65E-01	0	0	0	7.51E-01	7.95E-05	3.65E-01	0	0	0	1.59E-04	1.18E-04	0
Zakharov, D = 10	Mean	0	3.44E-08	0	0	2.09E+01	0	5.74E-07	8.75E-08	1.04E+01	0	0	3.00E-02	0	0
	Best	0	0	0	0	8.23E-07	0	1.21E-07	0	0	0	0	1.64E-03	0	0
	Worst	0	9.90E-08	0	0	5.42E+02	0	1.26E-06	1.32E-06	3.37E+01	0	0	1.84E-01	0	0
	Stdev	0	2.44E-08	0	0	9.87E+01	0	2.49E-07	2.54E-07	1.39E+01	0	0	3.76E-02	0	0
Ackley, D = 10	Mean	0	4.73E-05	0	1.93E-01	1.70E+01	0	2.76E-03	0	3.85E-02	0	0	1.14E-01	8.27E-01	0
	Best	0	1.30E-05	0	0	1.38E+01	0	2.00E-03	0	0	0	0	3.06E-02	0	0
	Worst	0	7.92E-05	0	2.32E+00	1.93E+01	0	3.41E-03	0	1.16E+00	0	0	1.81E-01	3.60E+00	0
	Stdev	0	1.70E-05	0	5.34E-01	1.35E+00	0	3.43E-04	0	2.11E-01	0	0	3.94E-02	1.52E+00	0
Periodic, D = 10	Mean	0	1.00E-01	1.62E-01	9.94E-02	1.00E-01	1.01E-01	1.00E-01	7.09E-02	6.13E-01	4.60E-01	7.55E-01	7.30E-05	2.26E-01	0
	Best	0	1.00E-01	1.00E-01	8.66E-03	1.00E-01	1.00E-01	1.00E-01	0	1.00E-01	2.53E-01	0	2.00E-05	1.37E-01	0
	Worst	0	1.00E-01	1.33E+00	1.11E-01	1.00E-01	1.02E-01	1.00E-01	1.09E-01	1.28E+00	6.21E-01	1.32E+00	1.20E-04	3.85E-01	0
	Stdev	0	1.36E-08	2.49E-01	2.22E-02	1.78E-07	4.07E-04	1.38E-07	4.72E-02	3.64E-01	9.49E-02	3.27E-01	2.78E-05	6.18E-02	0
Griewank, D = 10	Mean	0	1.08E-02	6.17E-02	4.58E-02	1.40E+00	2.73E-02	5.09E-02	2.60E-02	1.19E-01	1.66E-01	5.39E-02	1.84E-04	2.87E-01	0
	Best	0	0	1.48E-02	1.25E-02	3.57E-01	1.35E-04	5.72E-06	0	3.20E-02	0	0	7.47E-05	7.69E-03	0
	Worst	0	1.87E-01	1.35E-01	7.97E-02	3.27E+00	7.16E-02	1.75E-01	3.14E-01	3.30E-01	3.97E-01	3.15E-01	4.03E-04	7.47E-01	0
	Stdev	0	3.50E-02	2.67E-02	1.45E-02	7.90E-01	1.60E-02	4.98E-02	7.15E-02	6.58E-02	1.20E-01	8.39E-02	7.94E-05	1.83E-01	0
Salomon, D = 10	Mean	0	2.73E-01	1.67E-01	1.67E-01	8.45E+00	1.03E-01	9.99E-02	8.66E-02	5.00E-01	2.02E-01	9.99E-02	1.66E-01	1.60E-01	9.33E-02
	Best	0	9.99E-02	9.99E-02	9.99E-02	4.70E+00	9.99E-02	9.99E-02	0	2.00E-01	2.00E-01	9.99E-02	9.99E-02	9.99E-02	0
	Worst	0	5.00E-01	3.00E-01	5.00E-01	1.17E+01	2.00E-01	9.99E-02	2.00E-01	9.00E-01	2.57E-01	9.99E-02	4.59E-01	2.00E-01	9.99E-02
	Stdev	0	8.68E-02	5.47E-02	8.84E-02	1.83E+00	1.83E-02	0	5.71E-02	1.88E-01	1.04E-02	0	8.02E-02	4.98E-02	2.50E-02
Xin-She Yang 4, D = 10	Mean	0	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	9.67E-01	1.00E+00	1.00E+00	1.00E+00	1.36E-02	1.00E+00	9.67E-01
	Best	0	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	0	1.00E+00	1.00E+00	1.00E+00	7.01E-03	1.00E+00	4.56E-04
	Worst	0	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00	2.34E-02	1.00E+00	1.00E+00
	Stdev	0	0	0	1.71E-07	0	0	0	1.83E-01	1.59E-05	1.50E-06	2.26E-05	4.26E-03	1.32E-05	1.82E-01

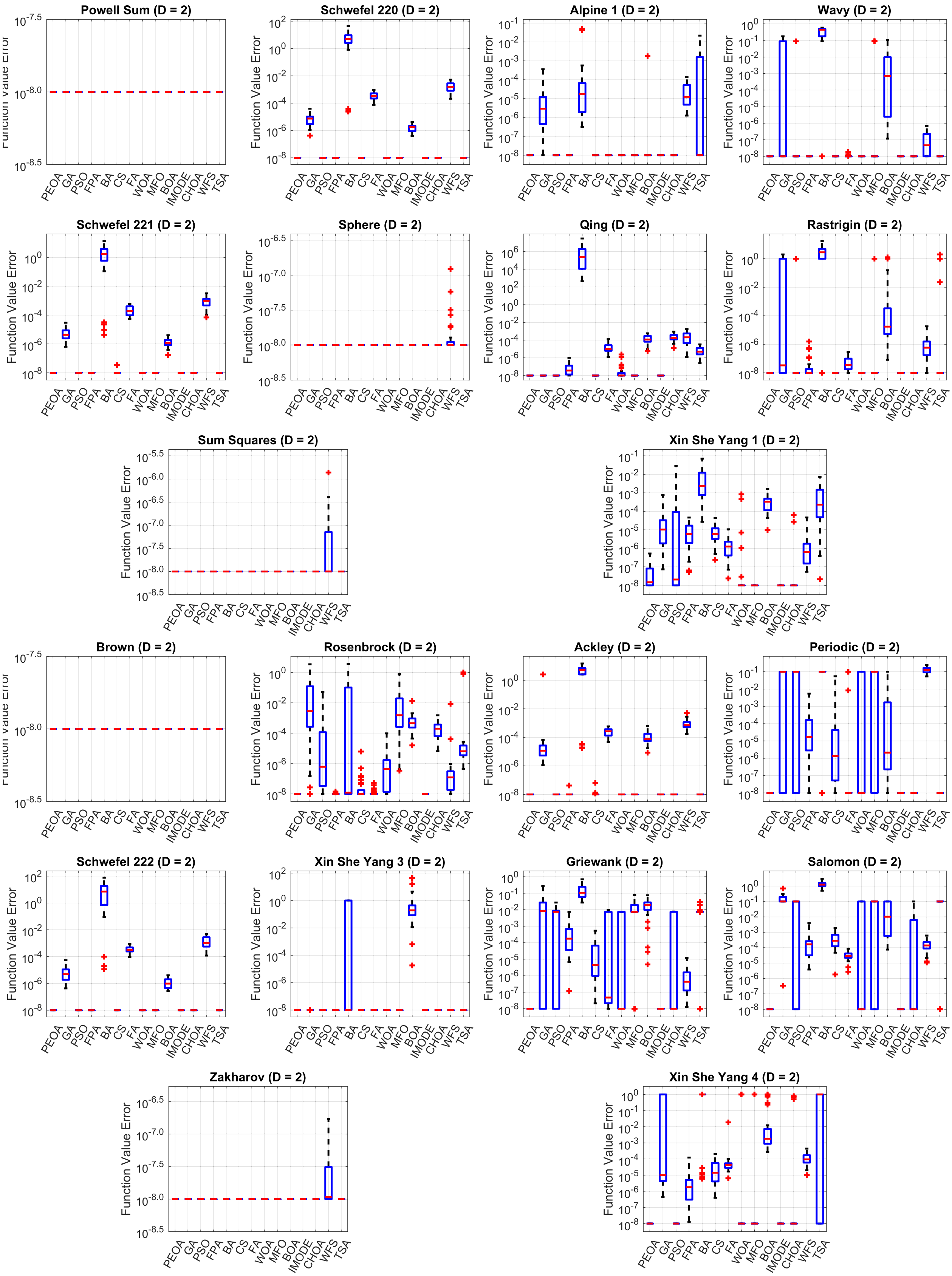


FIGURE 13. Boxplots over 30 independent runs (in logarithmic scale) of the function value errors obtained by the Philippine Eagle optimization algorithm and the 13 other examined algorithms for the 20 different functions of varied types and having *2 dimensions*.

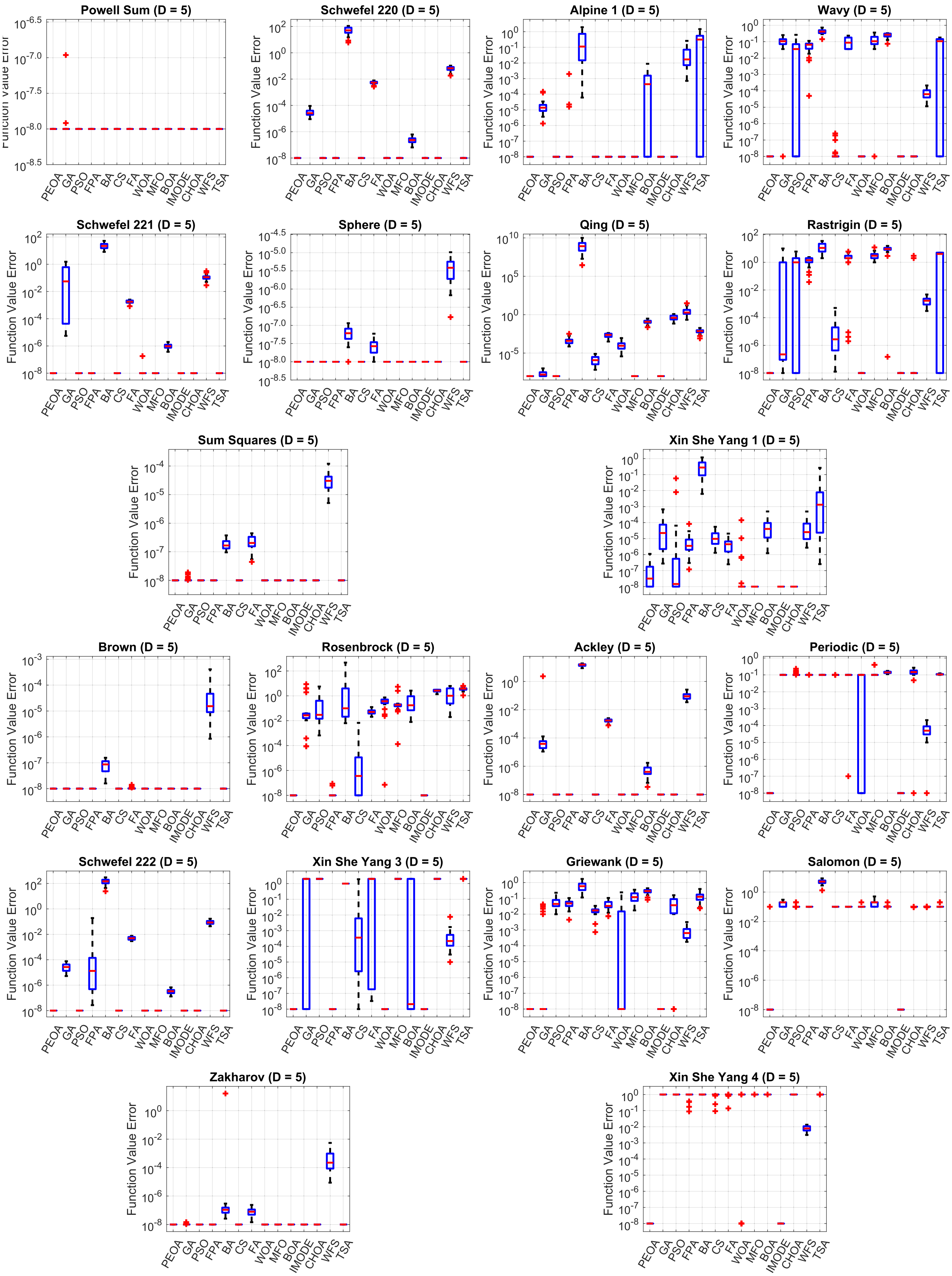


FIGURE 14. Boxplots over 30 independent runs (in logarithmic scale) of the function value errors obtained by the Philippine Eagle optimization algorithm and the 13 other examined algorithms for the 20 different functions of varied types and having *5 dimensions.*

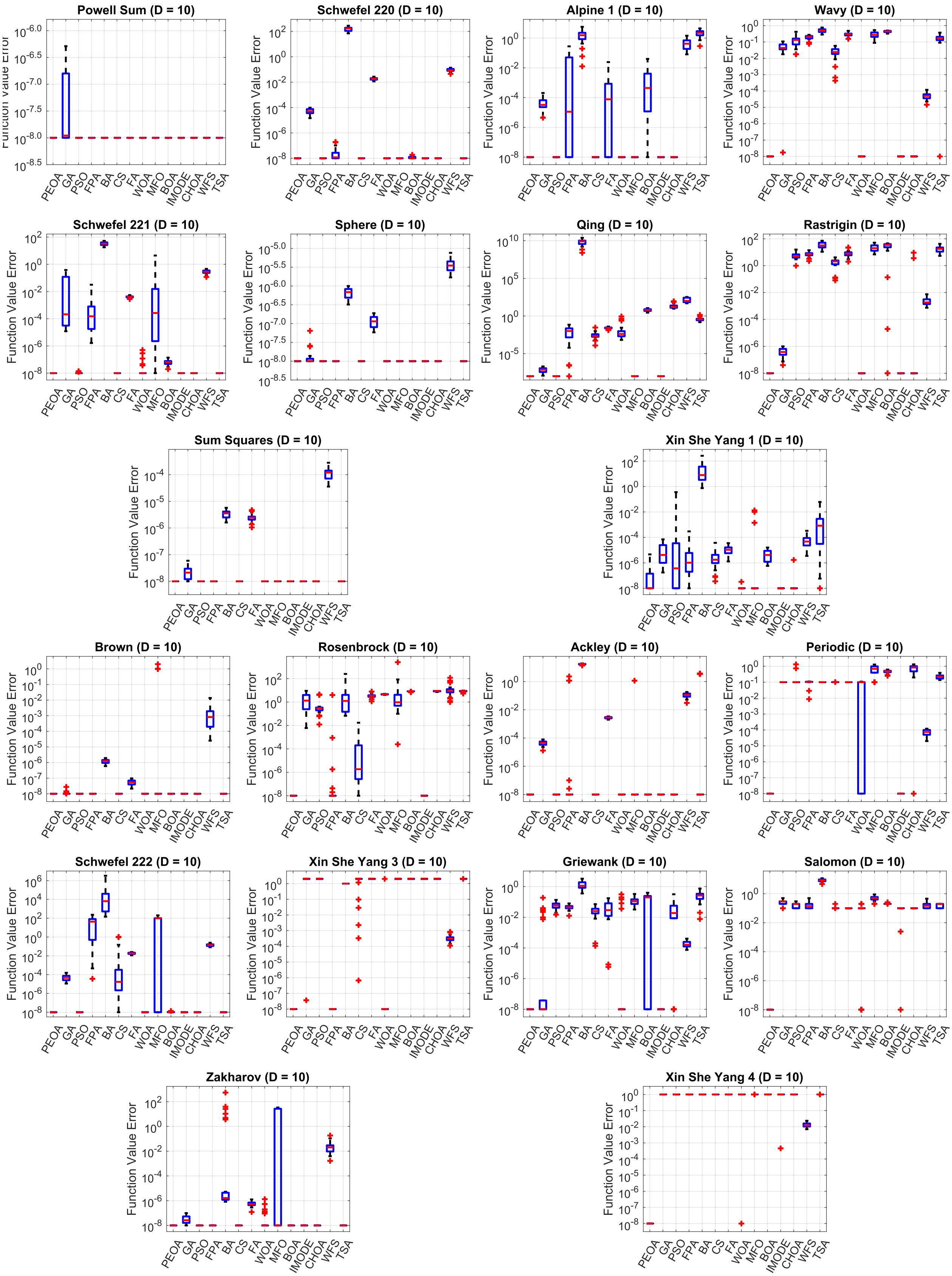


FIGURE 15. Boxplots over 30 independent runs (in logarithmic scale) of the function value errors obtained by the Philippine Eagle optimization algorithm and the 13 other examined algorithms for the 20 different functions of varied types and having *10 dimensions*.

APPENDIX A
RESULTS FOR FUNCTIONS WITH 2, 5, AND 10 DIMENSIONS

Tables 16, 17, and 18 present the average, best, and worst function value errors as well as the standard deviations obtained for functions with dimensions $D = 2, 5, 10$, respectively, using the different examined algorithms. The cells having a value of 0 are highlighted in green for emphasis.

On the other hand, Figures 13, 14, and 15 present the boxplots for functions with dimensions $D = 2, 5,$ and 10 respectively. They show the function value errors of the corresponding algorithms on the bottom axis. All values less than or equal to 10^{-8} are treated as 10^{-8} in the boxplots. Also, the logarithmic scale is used here.

REFERENCES

[1] X.-S. Yang, *Nature-Inspired Metaheuristic Algorithms*. Bristol, U.K.: Luniver Press, 2008.

[2] X.-S. Yang, *Engineering Optimization: An Introduction with Metaheuristic Applications*, Sep. 2010.

[3] C. Floudas and P. Pardalos, *Encyclopedia of Optimization*. Cham, Switzerland: Springer, 2008. [Online]. Available: https://books.google.com.ph/books?id=1a6lSRbQ4YsC

[4] X.-S. Yang, *Optimization Techniques and Applications With Examples*. Hoboken, NJ, USA: Wiley, Sep. 2018.

[5] J. H. Holland, "Genetic algorithms," *Sci. Amer.*, vol. 267, no. 1, pp. 66–72, Jul. 1992.

[6] R. Storn and K. Price, "Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces," *J. Global Optim.*, vol. 11, no. 4, pp. 341–359, 1997.

[7] X.-S. Yang, *Firefly Algorithms for Multimodal Optimization*. Berlin, Germany: Springer, 2009, pp. 169–178.

[8] X.-S. Yang and S. Deb, "Cuckoo search via Lévy flights," in *Proc. World Congr. Nature Biol. Inspired Comput. (NaBIC)*, Dec. 2009, pp. 210–214.

[9] X.-S. Yang, *A New Metaheuristic Bat-Inspired Algorithm*. Berlin, Germany: Springer, 2010, pp. 65–74.

[10] X.-S. Yang, *Flower Pollination Algorithm for Global Optimization*. Berlin, Germany: Springer, 2012, pp. 240–249.

[11] S. Mirjalili, "Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm," *Knowl.-Based Syst.*, vol. 89, pp. 228–249, Nov. 2015.

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

[13] S. Arora and S. Singh, "Butterfly optimization algorithm: A novel approach for global optimization," *Soft Comput.*, vol. 23, no. 3, pp. 715–734, 2018.

[14] N. Covic and B. Lacevic, "Wingsuit flying search—A novel global optimization algorithm," *IEEE Access*, vol. 8, pp. 53883–53900, 2020.

[15] S. Kaur, L. K. Awasthi, A. L. Sangal, and G. Dhiman, "Tunicate swarm algorithm: A new bio-inspired based Metaheuristic paradigm for global optimization," *Eng. Appl. Artif. Intell.*, vol. 90, Apr. 2020, Art. no. 103541.

[16] M. Khishe and M. R. Mosavi, "Chimp optimization algorithm," *Expert Syst. Appl.*, vol. 149, Jul. 2020, Art. no. 113338. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417420301639

[17] K. M. Sallam, S. M. Elsayed, R. K. Chakrabortty, and M. J. Ryan, "Improved multi-operator differential evolution algorithm for solving unconstrained problems," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jul. 2020, pp. 1–8.

[18] 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.

[19] P. Agrawal, H. F. Abutarboush, T. Ganesh, and A. W. Mohamed, "Metaheuristic algorithms on feature selection: A survey of one decade of research (2009–2019)," *IEEE Access*, vol. 9, pp. 26766–26791, 2021.

[20] M. Ahsan, K. Gupta, A. Nag, S. Poudyal, A. Kouzani, and M. Mahmud, "Applications and evaluations of bio-inspired approaches in cloud security: A review," *IEEE Access*, vol. 8, pp. 180799–180814, 2020.

[21] G. G. Tejani, V. J. Savsani, V. K. Patel, and S. Mirjalili, "An improved heat transfer search algorithm for unconstrained optimization problems," *J. Comput. Design Eng.*, vol. 6, no. 1, pp. 13–32, Jan. 2019.

[22] A. R. Ferrolino, J. E. C. Lope, and R. G. Mendoza, "Optimal location of sensors for early detection of tsunami waves," in *Proc. Int. Conf. Comput. Sci.*, Cham, Switzerland: Springer, 2020, pp. 562–575.

[23] A. Ferrolino, R. Mendoza, I. Magdalena, and J. E. Lope, "Application of particle swarm optimization in optimal placement of tsunami sensors," *PeerJ Comput. Sci.*, vol. 6, p. e333, Dec. 2020.

[24] T. Srihari, M. Boppa, S. Anil Kumar, and H. Pulluri, "The application of genetic algorithm with multi-parent crossover to optimal power flow problem," in *Innovations in Electrical and Electronics Engineering*. Cham, Switzerland: Springer, 2020, pp. 417–427.

[25] M. Mahdavi, H. H. Alhelou, A. Bagheri, S. Z. Djokic, and R. A. V. Ramos, "A comprehensive review of Metaheuristic methods for the reconfiguration of electric power distribution systems and comparison with a novel approach based on efficient genetic algorithm," *IEEE Access*, vol. 9, pp. 122872–122906, 2021.

[26] E. Jordan, D. E. Shin, S. Leekha, and S. Azarm, "Optimization in the context of COVID-19 prediction and control: A literature review," *IEEE Access*, vol. 9, pp. 130072–130093, 2021.

[27] G. G. Tejani, V. J. Savsani, and V. K. Patel, "Modified sub-population teaching-learning-based optimization for design of truss structures with natural frequency constraints," *J. Struct. Mech.*, vol. 44, no. 4, pp. 495–513, 2016.

[28] G. G. Tejani, N. Pholdee, S. Bureerat, D. Prayogo, and A. H. Gandomi, "Structural optimization using multi-objective modified adaptive symbiotic organisms search," *Expert Syst. Appl.*, vol. 125, pp. 425–441, Jul. 2019.

[29] G. G. Tejani, V. J. Savsani, S. Bureerat, V. K. Patel, and P. Savsani, "Topology optimization of truss subjected to static and dynamic constraints by integrating simulated annealing into passing vehicle search algorithms," *Eng. With Comput.*, vol. 35, no. 2, pp. 499–517, Apr. 2019.

[30] T. Kunakote, N. Sabangban, S. Kumar, G. G. Tejani, N. Panagant, N. Pholdee, S. Bureerat, and A. R. Yildiz, "Comparative performance of twelve metaheuristics for wind farm layout optimisation," *Arch. Comput. Methods Eng.*, vol. 29, pp. 717–730, Apr. 2021.

[31] S. Kumar, G. G. Tejani, N. Pholdee, and S. Bureerat, "Multiobjecitve structural optimization using improved heat transfer search," *Knowl.-Based Syst.*, vol. 219, May 2021, Art. no. 106811.

[32] S. Kumar, G. G. Tejani, N. Pholdee, and S. Bureerat, "Improved metaheuristics through migration-based search and an acceptance probability for truss optimization," *Asian J. Civil Eng.*, vol. 21, no. 7, pp. 1217–1237, Nov. 2020.

[33] V. J. Savsani, G. G. Tejani, and V. K. Patel, "Truss topology optimization with static and dynamic constraints using modified subpopulation teaching–learning-based optimization," *Eng. Optim.*, vol. 48, no. 11, pp. 1990–2006, Nov. 2016.

[34] S. Kumar, P. Jangir, G. G. Tejani, M. Premkumar, and H. H. Alhelou, "MOPGO: A new physics-based multi-objective plasma generation optimizer for solving structural optimization problems," *IEEE Access*, vol. 9, pp. 84982–85016, 2021.

[35] W. Kaidi, M. Khishe, and M. Mohammadi, "Dynamic levy flight chimp optimization," *Knowl.-Based Syst.*, vol. 235, Jan. 2022, Art. no. 107625. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S095070512100887X

[36] M. Khishe, M. Nezhadshahbodaghi, M. R. Mosavi, and D. Martin, "A weighted chimp optimization algorithm," *IEEE Access*, vol. 9, pp. 158508–158539, 2021.

[37] J. Wang, M. Khishe, M. Kaveh, and H. Mohammadi, "Binary chimp optimization algorithm (BChOA): A new binary meta-heuristic for solving optimization problems," *Cognit. Comput.*, vol. 13, no. 5, pp. 1297–1316, Sep. 2021, doi: 10.1007/s12559-021-09933-7.

[38] E. Pashaei and E. Pashaei, "An efficient binary chimp optimization algorithm for feature selection in biomedical data classification," *Neural Comput. Appl.*, Jan. 2022, doi: 10.1007/s00521-021-06775-0.

[39] T. Hu, M. Khishe, M. Mohammadi, G.-R. Parvizi, S. H. Taher Karim, and T. A. Rashid, "Real-time COVID-19 diagnosis from X-ray images using deep CNN and extreme learning machines stabilized by chimp optimization algorithm," *Biomed. Signal Process. Control*, vol. 68, Jul. 2021, Art. no. 102764. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S174680942100361X

[40] C. Wu, M. Khishe, M. Mohammadi, S. H. T. Karim, and T. A. Rashid, "Evolving deep convolutional neutral network by hybrid sine–cosine and extreme learning machine for real-time COVID19 diagnosis from X-ray images," *Soft Comput.*, May 2021, doi: 10.1007/s00500-021-05839-6.

[41] H. Li, X. Liu, Z. Huang, C. Zeng, P. Zou, Z. Chu, and J. Yi, "Newly emerging nature-inspired Optimization–Algorithm review, unified framework, evaluation, and behavioural parameter optimization," *IEEE Access*, vol. 8, pp. 72620–72649, 2020, doi: 10.1109/ACCESS.2020.2987689.

[42] D. Molina, J. Poyatos, J. D. Ser, S. García, A. Hussain, and F. Herrera, "Comprehensive taxonomies of Nature- and bio-inspired optimization: Inspiration versus algorithmic behavior, critical analysis recommendations," *Cognit. Comput.*, vol. 12, no. 5, pp. 897–939, Sep. 2020, doi: 10.1007/s12559-020-09730-8.

[43] I. Fister, X.-S. Yang, I. Fister, J. Brest, and D. Fister, "A brief review of nature-inspired algorithms for optimization," 2013, *arXiv:1307.4186*, doi: 10.48550/ARXIV.1307.4186.

[44] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. IEEE ICNN*, vol. 4, Nov./Dec. 1995, pp. 1942–1948.

[45] C.-X. Zhang, K.-Q. Zhou, S.-Q. Ye, and A. M. Zain, "An improved cuckoo search algorithm utilizing nonlinear inertia weight and differential evolution for function optimization problem," *IEEE Access*, vol. 9, pp. 161352–161373, 2021.

[46] I. Fister, J. Brest, A. Iglesias, A. Galvez, S. Deb, and I. Fister, "On selection of a benchmark by determining the Algorithms' qualities," *IEEE Access*, vol. 9, pp. 51166–51178, 2021. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85100836310&doi=10.%11092fACCESS.2021.3058285&partnerID=40&md5=f8c01ac858e40a8165894b55b9e4bc61

[47] Z.-M. Gao, J. Zhao, Y.-R. Hu, and H.-F. Chen, "The challenge for the nature-inspired global optimization algorithms: Non-symmetric benchmark functions," *IEEE Access*, vol. 9, pp. 106317–106339, 2021. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85112619386&doi=10.%11092fACCESS.2021.3100365&partnerID=40&md5=a4f7aefc8b0c0bc3cc17ab8ecb0cd0e0

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

[49] A. Kazikova, M. Pluhacek, and R. Senkerik, "How does the number of objective function evaluations impact our understanding of metaheuristics behavior?" *IEEE Access*, vol. 9, pp. 44032–44048, 2021. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85103755169&doi=10.%11092fACCESS.2021.3066135&partnerID=40&md5=74264b3dc6e4ffa7586c869a56a58df0

[50] S. D. Majetic. (2010). *Philippine Eagle an Endangered Species, Creative Commons Licensed (By-Sa 2.0). File: Philippine Eagle an Endangered Species.Jpg, Creative Commons Licensed (BY-SA 2.0), Via Wikimedia Commons*. [Online]. Available: https://upload.wikimedia.org/wikipedia/commons/5/54/Philippine_Eagle_an_Endangered_Species.jpg

[51] J. Ibanez, A. M. Sumaya, G. Tampos, and D. Salvador, "Preventing philippine eagle hunting: What are we missing?" *J. Threatened Taxa*, vol. 8, no. 13, p. 9505, Nov. 2016.

[52] A. U. Luczon, I. K. C. Fontanilla, P. S. Ong, Z. U. Basiao, A. M. T. Sumaya, and J. P. Quilang, "Genetic diversity of the critically endangered philippine eagle pithecophaga jefferyi (aves: Accipitridae) and notes on its conservation," *J. Threatened Taxa*, vol. 6, no. 10, pp. 6335–6344, Sep. 2014.

[53] B. International, "Pithecophaga Jefferyi," in *IUCN Red List Threatened Species*, Oct. 2016. [Online]. Available: https://www.iucnredlist.org/

[54] R. S. Kennedy, "Notes on the biology and population status of the monkey-eating eagle of the Philippines," *Wilson Bull*, vol. 89, no. 1, pp. 1–20, 1977.

[55] C. B. Concepcion, M. Sulapas, and J. C. Ibañez, "Notes on food habits and breeding and nestling behavior of Philippine Eagles in Mount Apo Natural Park, Mindanao, Philippines," *Banwa*, vol. 3, nos. 1–2, pp. 81–95, 2006. [Online]. Available: http://ojs.upmin.edu.ph/index.php/banwa-archives/article/download/19/24

[56] M. Cavazzuti, *Optimization Methods*. Berlin, Germany: Springer, 2013, doi: 10.1007/978-3-642-31187-1.

[57] S. Elsayed, N. Hamza, and R. Sarker, "Testing united multi-operator evolutionary algorithms-II on single objective optimization problems," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jul. 2016, pp. 2966–2973.

[58] J. Zhang and A. C. Sanderson, "JADE: Adaptive differential evolution with optional external archive," *IEEE Trans. Evol. Comput.*, vol. 13, no. 5, pp. 945–958, Oct. 2009.

[59] A. M. Reynolds and M. A. Frye, "Free-flight odor tracking in Drosophila is consistent with an optimal intermittent scale-free search," *PLoS ONE*, vol. 2, no. 4, pp. 1–9, 2007.

[60] M. Gutowski, "Levy flights as an underlying mechanism for global optimization algorithms," Jul. 2001, *arXiv:math-ph/0106003*.

[61] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, "Harris hawks optimization: Algorithm and applications," *Future Gener. Comput. Syst.*, vol. 97, pp. 849–872, Aug. 2019.

[62] R. Tanabe and A. S. Fukunaga, "Improving the search performance of SHADE using linear population size reduction," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jul. 2014, pp. 1658–1665.

[63] N. Covic and B. Lacevic, "Wingsuit flying search—A novel global optimization algorithm," *IEEE Access*, vol. 8, pp. 53883–53900, 2020.

[64] D. Ortiz-Boyer, C. Hervá-Martínez, and N. García-Pedrajas, "CIXL2: A crossover operator for evolutionary algorithms based on population features," *J. Artif. Intell. Res.*, vol. 24, no. 1, pp. 1–48, 2005.

[65] X. Yao, Y. Liu, K.-H. Liang, and G. Lin, *Fast Evolutionary Algorithms*. Berlin, Germany: Springer, 2003, pp. 45–94.

[66] S. Surjanovic and D. Bingham. (Sep. 25, 2021). *Virtual Library of Simulation Experiments: Test Functions and Datasets*. [Online]. Available: http://www.sfu.ca/~ssurjano

[67] C. T. Yue, K. V. Price, P. N. Suganthan, J. J. Liang, M. Z. Ali, B. Y. Qu, N. H. Awad, and P. P. Biswas, "Problem definitions and evaluation criteria for the cec 2020 special session and competition on single objective bound constrained numerical optimization," Comput. Intell. Lab., Zhengzhou Univ., Zhengzhou China, Nanyang Technol. Univ., Singapore, Tech. Rep. 201911, Nov. 2019.

[68] X.-S. Yang. (Sep. 25, 2021). *Flower Pollination Algorithm*. MATLAB Central File Exchange. [Online]. Available: https://www.mathworks.com/matlabcentral/fileexchange/45112-flower-polli%nation-algorithm

[69] X.-S. Yang. (Sep. 25, 2021). *Bat Algorithm (Demo)*. MATLAB Central File Exchange. [Online]. Available: https://www.mathworks.com/matlabcentral/fileexchange/37582-bat-algorith%m-demo

[70] X.-S. Yang. (Sep. 25, 2021). *Cuckoo Search (CS) Algorithm*. MATLAB Central File Exchange. [Online]. Available: https://www.mathworks.com/matlabcentral/fileexchange/29809-cuckoo-searc%h-cs-algorithm

[71] X.-S. Yang. (Sep. 25, 2021). *Firefly Algorithm*. MATLAB Central File Exchange. [Online]. Available: https://www.mathworks.com/matlabcentral/fileexchange/29693-firefly-algo%rithm

[72] S. Mirjalili. (2021). *The Whale Optimization Algorithm*. MATLAB Central File Exchange. [Online]. Available: https://www.mathworks.com/matlabcentral/fileexchange/55667-the-whale-op%timization-algorithm

[73] S. Mirjalili. (Sep. 25, 2021). *Moth-Flame Optimization (MFO) Algorithm*. MATLAB Central File Exchange. [Online]. Available: https://www.mathworks.com/matlabcentral/fileexchange/52269-moth-flame-o%ptimization-mfo-algorithm

[74] S. Arora. (Sep. 25, 2021). *Butterfly Optimization Algorithm (BOA) Source Codes Demo v1.0*. [Online]. Available: https://www.mathworks.com/matlabcentral/mlc-downloads/downloads/b4a529a%c-c709-4752-8ae1-1d172b8968fc/67a434dc-8224-4f4e-a835-bc92c4630a73/previews/BO%A.m/index.html

[75] M. Khishe and M. Mosavi, "Chimp optimization algorithm," *Expert Syst. With Appl.*, vol. 149, Jul. 2020, Art. no. 113338, doi: 10.1016/j.eswa.2020.113338.

[76] N. Covic and B. Lacevic. (2020). *Wingsuit-Flying-Search*. [Online]. Available: https://github.com/ncovic1/Global-Optimization-Heuristic-Algorithms.git%

[77] G. Dhiman. (Feb. 10, 2022). *Tunicate Swarm Algorithm (TSA)*. MATLAB Central File Exchange. [Online]. Available: https://www.mathworks.com/matlabcentral/fileexchange/75182-tunicate-swa%rm-algorithm-tsa

[78] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics Bull.*, vol. 1, no. 6, pp. 80–83, 1945. [Online]. Available: http://www.jstor.org/stable/3001968

[79] W. Qiao, M. Khishe, and S. Ravakhah, ''Underwater targets classification using local wavelet acoustic pattern and multi-layer perceptron neural network optimized by modified whale optimization algorithm,'' *Ocean Eng.*, vol. 219, Jan. 2021, Art. no. 108415.

[80] M. R. Mosavi and M. Khishe, ''Training a feed-forward neural network using particle swarm optimizer with autonomous groups for sonar target classification,'' *J. Circuits, Syst. Comput.*, vol. 26, no. 11, Nov. 2017, Art. no. 1750185, doi: 10.1142/S0218126617501857.

[81] M. Khishe and H. Mohammadi, ''Passive sonar target classification using multi-layer perceptron trained by salp swarm algorithm,'' *Ocean Eng.*, vol. 181, pp. 98–108, Jun. 2019.

[82] E. A. Enriquez. (2021). *Philippine-Eagle-Optimization-Algorithm.* [Online]. Available: https://github.com/ErikaAntonette/Philippine-Eagle-Optimization-Algorit%hm.git

[83] J. Newell, D. Isaacson, and J. Mueller, ''Electrical impedance tomography,'' *IEEE Trans. Med. Imag.*, vol. 21, pp. 553–554, 2002.

[84] E. Teschner, M. Imhoff, and S. Leonhardt, *Electrical Impedance Tomography: The Realisation of Regional Ventilation Monitoring*, 2nd ed. Lübeck, Germany: Dräger Medical GmbH, 2015.

[85] D. Holder, *Clinical and Physiological Applications of Electrical Impedance Tomography*. London, U.K.: UCL Press, 1993.

[86] N. D. Harris, A. J. Suggett, D. C. Barber, and B. H. Brown, ''Applications of applied potential tomography (APT) in respiratory medicine,'' *Clin. Phys. Physiolog. Meas.*, vol. 8, no. 4A, pp. 155–165, Nov. 1987.

[87] I. Frerichs, J. Hinz, P. Herrmann, G. Weisser, G. Hahn, T. Dudykevych, M. Quintel, and G. Hellige, ''Detection of local lung air content by electrical impedance tomography compared with electron beam CT,'' *J. Appl. Physiol.*, vol. 93, no. 2, pp. 660–666, 2002.

[88] D. G. Tingay, A. D. Waldmann, I. Frerichs, S. Ranganathan, and A. Adler, ''Electrical impedance tomography can identify ventilation and perfusion defects: A neonatal case,'' *Amer. J. Respiratory Crit. Care Med.*, vol. 199, no. 3, pp. 384–386, Feb. 2019.

[89] A. Adler, Y. Berthiaume, R. Guardo, and R. Amyot, ''Imaging of pulmonary edema with electrical impedance tomography,'' in *Proc. 17th Int. Conf. Eng. Med. Biol. Soc.*, Sep. 1995, pp. 557–558.

[90] V. Cherepenin, A. Karpov, A. Korjenevsky, V. Kornienko, A. Mazaletskaya, D. Mazourov, and D. Meister, ''A 3D electrical impedance tomography (EIT) system for breast cancer detection,'' *Physiolog. Meas.*, vol. 22, no. 1, pp. 9–18, Feb. 2001.

[91] R. J. Halter, A. Hartov, and K. D. Paulsen, ''A broadband high-frequency electrical impedance tomography system for breast imaging,'' *IEEE Trans. Biomed. Eng.*, vol. 55, no. 2, pp. 650–659, Feb. 2008.

[92] A. P. Bagshaw, A. D. Liston, R. H. Bayford, A. Tizzard, A. P. Gibson, A. T. Tidswell, M. K. Sparkes, H. Dehghani, C. D. Binnie, and D. Holder, ''Electrical impedance tomography of human brain function using reconstruction algorithms based on the finite element method,'' *NeuroImage*, vol. 20, no. 2, pp. 752–764, 2003.

[93] D. S. Holder, ''Electrical impedance tomography (EIT) of brain function,'' *Brain Topography*, vol. 5, no. 2, pp. 87–93, 1992.

[94] A. C. Velasco, M. Darbas, R. Mendoza, M. Bacon, and J. C. de Leon, ''Comparative study of heuristic algorithms for electrical impedance tomography,'' *Philippine J. Sci.*, vol. 149, pp. 747–761, Jan. 2020.

[95] E. Somersalo, M. Cheney, and D. Isaacson, ''Existence and uniqueness for electrode models for electric current computed tomography,'' *SIAM J. Appl. Math.*, vol. 52, no. 4, pp. 1023–1040, Jul. 1992.

[96] M. Darbas, J. Heleine, R. Mendoza, and A. C. Velasco, ''Sensitivity analysis of the complete electrode model for electrical impedance tomography,'' *AIMS Math.*, vol. 6, no. 7, pp. 7333–7366, 2021.

[97] M. Zamani, R. Bayford, and A. Demosthenous, ''Adaptive electrical impedance tomography resolution enhancement using statistically quantized projected image sub-bands,'' *IEEE Access*, vol. 8, pp. 99797–99805, 2020.

[98] X. He, Y. Zhang, J. Li, Z. Song, J. Wen, J. Ma, and Y. Hu, ''An improved algorithm GVSPM-F for electrical impedance tomography,'' *IEEE Access*, vol. 9, pp. 12592–12600, 2021.

[99] W. Shang, W. Xue, Y. Li, and Y. Xu, ''Improved primal–dual interior-point method using the Lawson-norm for inverse problems,'' *IEEE Access*, vol. 8, pp. 41053–41061, 2020.

[100] A. C. Velasco, M. Darbas, and R. Mendoza, ''Numerical resolution of the electrical impedance tomography inverse problem with fixed inclusions,'' *Comput. Sci.*, vol. 16, no. 3, pp. 1063–1076, 2021.

[101] W. Zhang, T. Zhang, X. Liu, B. Yang, M. Dai, X. Shi, X. Dong, F. Fu, and C. Xu, ''Target adaptive differential iterative reconstruction (TADI): A robust algorithm for real-time electrical impedance tomography,'' *IEEE Access*, vol. 9, pp. 141999–142011, 2021.

[102] R. Mendoza and S. Keeling, ''A two-phase segmentation approach to the impedance tomography problem,'' *Inverse Problems*, vol. 33, no. 1, Jan. 2017, Art. no. 015001.

[103] H. Wang, K. Liu, Y. Wu, S. Wang, Z. Zhang, F. Li, and J. Yao, ''Image reconstruction for electrical impedance tomography using radial basis function neural network based on hybrid particle swarm optimization algorithm,'' *IEEE Sensors J.*, vol. 21, no. 2, pp. 1926–1934, Jan. 2021.

[104] Y. Zhang, H. Chen, L. Yang, K. Liu, F. Li, C. Bai, H. Wu, and J. Yao, ''A proportional genetic algorithm for image reconstruction of static electrical impedance tomography,'' *IEEE Sensors J.*, vol. 20, no. 24, pp. 15026–15033, Dec. 2020.

[105] R. G. Mendoza and J. E. C. Lope, ''Reconstructing images in electrical impedance tomography using hybrid genetic algorithms,'' *Sci. Diliman*, vol. 24, no. 2, pp. 1–17, 2012.

[106] J. Krueger, ''Parameter estimation methods for ordinary differential equation models with applications to microbiology,'' Ph.D. dissertation, Dept. Math., Virginia Polytech. Inst. State Univ., Tempe, Arizona, 2017.

[107] A. Alfi, ''PSO with adaptive mutation and inertia weight and its application in parameter estimation of dynamic systems,'' *Acta Autom. Sin.*, vol. 37, no. 5, pp. 541–549, May 2011.

[108] J. Ri non, R. Mendoza, and V. Mendoza, ''Parameter estimation of an S-system model using hybrid genetic algorithm with the aid of sensitivity analysis,'' in *Proceedings of the 2019 Philippine Computing Science Congress*. Makati, Philippines: Philippine Computing Society, 2019, pp. 94–102.

[109] L. Lu, Z. Guo, Z. Wang, Z. Huang, and M. Liu, ''Parameter estimation for a capacitive coupling communication channel within a metal cabinet based on a modified artificial immune algorithm,'' *IEEE Access*, vol. 9, pp. 75683–75698, 2021.

[110] A. G. A. Muthalif, M. K. M. Razali, N. H. D. Nordin, and S. B. A. Hamid, ''Parametric estimation from empirical data using particle swarm optimization method for different magnetorheological damper models,'' *IEEE Access*, vol. 9, pp. 72602–72613, 2021.

[111] O. Rodriguez-Abreo, J. M. Hernandez-Paredes, A. F. Rangel, C. Fuentes-Silva, and F. A. C. Velasquez, ''Parameter identification of motors by cuckoo search using steady-state relations,'' *IEEE Access*, vol. 9, pp. 72017–72024, 2021, doi: 10.1109/access.2021.3078578.

[112] C. U. Jamilla, R. G. Mendoza, and V. M. P. Mendoza, ''Parameter estimation in neutral delay differential equations using genetic algorithm with multi-parent crossover,'' *IEEE Access*, vol. 9, pp. 131348–131364, 2021.

[113] C. Jamilla, R. Mendoza, and I. Mező, ''Solutions of neutral delay differential equations using a generalized Lambert W function,'' *Appl. Math. Comput.*, vol. 382, Oct. 2020, Art. no. 125334.

[114] Y. N. Kyrychko and S. J. Hogan, ''On the use of delay equations in engineering applications,'' *J. Vibrat. Control*, vol. 16, nos. 7–8, pp. 943–960, Jun. 2010.

[115] M. Fliess, H. Mounier, P. Rouchon, and J. Rudolph, ''Controllability and motion planning for linear delay systems with an application to a flexible rod,'' in *Proc. 34th IEEE Conf. Decis. Control*, Dec. 1995, pp. 2046–2051.

[116] D. J. Nagy, L. Bencsik, and T. Insperger, ''Experimental estimation of tactile reaction delay during stick balancing using cepstral analysis,'' *Mech. Syst. Signal Process.*, vol. 138, Apr. 2020, Art. no. 106554.

[117] A. Domoshnitsky, S. Levi, R. H. Kappel, E. Litsyn, and R. Yavich, ''Stability of neutral delay differential equations with applications in a model of human balancing,'' *Math. Model. Natural Phenomena*, vol. 16, p. 21, Jan. 2021.

[118] C. Jamilla, R. Mendoza, and V. V. M. Mendoza, ''Explicit solution of a Lotka–Sharpe–McKendrick system involving neutral delay differential equations using the r-Lambert w function,'' *Math. Biosci. Eng.*, vol. 17, no. 5, pp. 5686–5708, 2020.

[119] C. T. H. Baker, G. A. Bocharov, C. A. H. Paul, and F. A. Rihan, ''Modelling and analysis of time-lags in some basic patterns of cell proliferation,'' *J. Math. Biol.*, vol. 37, no. 4, pp. 341–371, Oct. 1998.

[120] Y. Kyrychko, K. Blyuss, A. Gonzalez-Buelga, S. Hogan, and D. Wagg, ''Real-time dynamic substructuring in a coupled oscillator-pendulum system,'' *Proc. Roy. Soc. A, Math., Phys. Eng. Sci.*, vol. 462, no. 2068, pp. 1271–1294, 2006.

ERIKA ANTONETTE T. ENRIQUEZ received the B.S. degree in mathematics from the University of the Philippines Diliman, in 2019, where she is currently pursuing the M.S. degree in applied mathematics with the Institute of Mathematics.

She is also an Instructor with the University of the Philippines Diliman. Her research interests include numerical analysis, numerical optimization, nature-inspired algorithms, swarm intelligence, and metaheuristics.

RENIER G. MENDOZA received the B.S. degree in mathematics and M.S. degree in applied mathematics degree from the University of the Philippines Diliman, in 2007 and 2010, respectively, and the Dr.rer.nat. degree in mathematics from the Karl-Franzens University of Graz, Austria, in 2015.

From 2007 to 2015, he was an Instructor with the Institute of Mathematics, University of the Philippines Diliman, where he has been an Assistant Professor, since 2015. He is the first prize winner of the 2021 National Academy of Science and Technology Talent Search for Young Scientists. He works on numerical optimization, mathematical modeling, and partial differential equations. His currently research focuses on image reconstruction in electrical impedance tomography, applications of shallow water equations in tsunami mitigation strategies, solutions of neutral delay differential equations, data graduation in Hilbert spaces, and modeling the transmission of neglected tropical diseases.

ARRIANNE CRYSTAL T. VELASCO received the B.S. degree in mathematics and the M.S. degree in applied mathematics from the University of the Philippines Diliman, in 2010 and 2015, respectively, and the Ph.D. degree in mathematics from the Université de Picardie Jules Verne, Amiens, France, in 2020.

From 2011 to 2020, she was an Instructor with the Institute of Mathematics, University of the Philippines Diliman, where she has been an Assistant Professor, since November 2020. She is currently working on the study of the complete electrode model for electroencephalography, the development of metaheuristic algorithms on real-world applications, and the mathematical modeling of a volcanic ash dispersion. Her research areas of interests include numerical analysis, numerical optimization, partial differential equations, and inverse problems.