Hindawi
BioMed Research International
Volume 2020, Article ID 8506365, 17 pages
https://doi.org/10.1155/2020/8506365

Research Article

An Innovative Excited-ACS-IDGWO Algorithm for Optimal Biomedical Data Feature Selection

Davies Segera, Mwangi Mbuthia, and Abraham Nyete

Department of Electrical and Information Engineering, University of Nairobi, Nairobi 30197, Kenya

Correspondence should be addressed to Davies Segera; davies.segera@uonbi.ac.ke

Received 2 February 2020; Revised 24 May 2020; Accepted 24 July 2020; Published 18 August 2020

Academic Editor: Bilal Alatas

Finding an optimal set of discriminative features is still a crucial but challenging task in biomedical science. The complexity of the task is intensified when any of the two scenarios arise: a highly dimensioned dataset and a small sample-sized dataset. The first scenario poses a big challenge to existing machine learning approaches since the search space for identifying the most relevant feature subset is so diverse to be explored quickly while utilizing minimal computational resources. On the other hand, the second aspect poses a challenge of too few samples to learn from. Though many hybrid metaheuristic approaches (i.e., combining multiple search algorithms) have been proposed in the literature to address these challenges with very attractive performance compared to their counterpart standard standalone metaheuristics, more superior hybrid approaches can be achieved if the individual metaheuristics within the proposed hybrid algorithms are improved prior to the hybridization. Motivated by this, we propose a new hybrid Excited- (E-) Adaptive Cuckoo Search- (ACS-) Intensification Dedicated Grey Wolf Optimization (IDGWO), i.e., EACSIDGWO. EACSIDGWO is an algorithm where the step size of ACS and the nonlinear control strategy of parameter $\vec{a}$ of the IDGWO are innovatively made adaptive via the concept of the complete voltage and current responses of a direct current (DC) excited resistor-capacitor (RC) circuit. Since the population has a higher diversity at early stages of the proposed EACSIDGWO algorithm, both the ACS and IDGWO are jointly involved in local exploitation. On the other hand, to enhance mature convergence at latter stages of the proposed algorithm, the role of ACS is switched to global exploration while the IDGWO is still left conducting the local exploitation. To prove that the proposed algorithm is superior in providing a good learning from fewer instances and an optimal feature selection from information-rich biomedical data, all these while maintaining a high classification accuracy of the data, the EACSIDGWO is employed to solve the feature selection problem. The EACSIDGWO as a feature selector is tested on six standard biomedical datasets from the University of California at Irvine (UCI) repository. The experimental results are compared with the state-of-the-art feature selection techniques, including binary ant-colony optimization (BACO), binary genetic algorithm (BGA), binary particle swarm optimization (BPSO), and extended binary cuckoo search algorithm (EBCSA). These results reveal that the EACSIDGWO has comprehensive superiority in tackling the feature selection problem, which proves the capability of the proposed algorithm in solving real-world complex problems. Furthermore, the superiority of the proposed algorithm is proved via various numerical techniques like ranking methods and statistical analysis.

1. Introduction

Currently, there is a growing research interest in developing and deploying population-based metaheuristics to tackle combinatorial optimization challenges. This is because they are simple, flexible with an inexpensive computational cost, and gradient-free [1].

Many researchers have applied these optimization algorithms in various research domains because of their ability to achieve best solutions.

The optimization challenge grows bigger when tackling highly dimensioned datasets. This is because these datasets have a vast feature space with many classes. Due to the presence of redundant and noninformative attributes within these datasets, the process of effective machine learning greatly hindered. Thus, the construction of efficient classifiers with high predictive power largely depends on selection of informative features [2].

Feature selection (FS) is one of the main steps in data preprocessing that aims at selecting a subset of attributes out of the whole dataset resulting into removal of noisy non-informative and redundant features. This in turn increases the accuracy of a considered classifier or clustering model [3].

FS algorithms can be broadly categorized into two classes: filter and wrapper techniques [4, 5]. Filters include techniques independent of classifiers and work directly on presented data. Moreover, these methods in many situations determine the correlations between features. On the contrary, wrapper approaches engage classifiers and mainly determine interactions between dataset features. From literature, wrapper approaches have proved to be superior compared to filters for classification algorithms [6, 7].

To utilize wrapper-based techniques, three key factors need to be outlined: considered classifiers (i.e., k-nearest neighbor (KNN), support vector machine (SVM)), evaluation criteria for the identified feature subset, and a search technique utilized in determining a subset of optimal features [8].

Many researchers have pointed out that determining an optimal subset of attributes is not only challenging but computationally expensive as well. Though, in the recent past, metaheuristics have proved to be reliable and efficient tools in tackling many optimization tasks (e.g., engineering designs problems, machine learning, feature selection, and data mining), they are not efficient in solving problems with high computational complexity [5, 9–11].

In the recent past, a number of metaheuristic search algorithms have been utilized for FS using highly dimensioned datasets. Some of these metaheuristics are the grey wolf optimization (GWO) [12, 13], genetic algorithm (GA) [14], particle swarm optimization (PSO) [11], ant-colony optimization (ACO) [15], differential evolution algorithm (DEA) [16], cuckoo search algorithm (CSA) [17], and dragonfly algorithm (DA) [18]. Though, many of these algorithms have already made an important contribution in the field of feature selection, in many cases, they offer acceptable solutions without a guarantee of determining optimal solutions since they do not explore the entire search space [11].

Some of the new modifications that have been proposed to improve the performance of these metaheuristics include chaotic maps [19], evolutionary methods [20], sine cosine algorithms [21], biogeography-based optimization, and local searches [22].

While designing or utilizing a metaheuristic, it should be noted that diversification (exploring the search space) and intensification (exploiting optimal solutions obtained so far) are two contradicting principles that must be balanced efficiently in order to achieve an improved performance of the metaheuristic [9].

In this regard, one promising alternative is developing a memetic algorithm whereby an integration of (at least) two algorithms is done with the aim of enhancing the overall performance.

Motivated by this, a good number of hybrid algorithms have been proposed in the recent past to solve a variety of optimizations and feature selection problems [23]. However, to enhance diversification and intensification of these hybrid algorithms, exploration and fine-tuning within their basic constituent algorithms is needed prior to hybridization [24].

This emphasizes, too, that there are a number of techniques lying within these memetic algorithms that are yet to be investigated.

Firstly, the technique of combining one or more nature-inspired algorithms (NIAs) needs to be determined. Secondly, the criterion of determining how many NIAs need to be combined within the search space has to be accomplished. Thirdly, the method of determining the application area upon which the proposed memetic algorithm has to be done. Finally, the criterion of applying the memetic algorithm in a specific domain has to be accomplished [24].

Inspired by the aforementioned, this paper proposes a new hybrid algorithm called Excited- (E-) Adaptive Cuckoo Search- (ACS-) Intensification Dedicated Grey Wolf Optimization (IDGWO), i.e., EACSIDGWO algorithm to solve the feature selection problem in biomedical science. In the proposed algorithm, the concept of the complete voltage and current responses of a direct current (DC) excited resistor capacitor (RC) circuit is innovatively utilized to make the step size of ACS and the nonlinear control strategy of parameter $\vec{a}$ of the IDGWO adaptive. Since the population has a higher diversity during early stages of the proposed algorithm, both the ACS and IDGWO are jointly utilized to attain accelerated convergence. However, to enhance mature convergence while striking an effective balance between exploitation and exploration in latter stages, the role of ACS is switched to global exploration while the IDGWO is still left conducting the local exploitation.

The remainder of this paper is organized as follows: Section 2 discusses the existing literature within the same research domain. Section 3 presents the background information of the CS and the GWO, respectively, where their inspirations and mathematical models are given emphasis. The continuous version of the proposed EACSIDGWO algorithm is presented in Section 4 while the details of its binary version are given in Section 5. The experimental methodology considered in this paper is presented in Section 6 while the results on feature selection are discussed in Section 7. Finally, conclusions and the suggested future works are given in Section 8.

2. Literature Reviews

2.1. Review of Hybridization of GWO with Other Search Algorithms. Combining two or more metaheuristics to attain better solutions is currently a new insight in the area of optimization. In the literature, many researchers have utilized GWO in the field of hybrid metaheuristics. For instance, in [25], a hybrid of GWO and artificial bee colony (ABC) is proposed to improve performance of a complex system. In [26], GWO is hybridized with ant lion optimizer (ALO) for wrapper feature selection. Alomoush et al. [27] proposed a hybrid of GWO and harmony search (HS). In this memetic, GWO updates the bandwidth and pitch adjustment rate in HS, which in return improves the global optimization abilities of the hybrid algorithm. In [28], Arora et al. combined GWO with the crow search algorithm (CSA). The performance of the derived memetic as a feature selector is evaluated using 21 datasets. The obtained results reveal that the combined algorithm is superior in solving complex optimization algorithms. In [29], a novel combination between GWO and PSO is utilized as a load-balancing technique in the cloud-computing arena. The conclusions point out that the hybrid algorithm improved both the convergence speed and the simplicity in comparison with other algorithms. Zhu et al. [30] hybridized GWO with differential evolution (DE). The hybrid algorithm was tested on 23 different functions and a nondeterministic polynomial hard problem. The obtained results indicate that this combination achieved superior exploration. In [31], a new memetic combining the exploration ability of the fireworks algorithm (FWA) with the exploitation ability of GWO is proposed. Utilizing 16 benchmark functions with varied dimensions and complexities, the experimental results indicate that the hybrid algorithm attained attractive global search abilities and convergence speeds.

2.2. Review of Hybridization of CS with Other Search Algorithms. Utilizing the concept of rand and best agents within a population, Cheng et al. [32] developed an ensemble cuckoo search variant combining three different CS approaches that coexist within the entire search domain. These CS variants actively compete to derive superior generations for numerical optimization. To maintain population diversity, he introduced an external archive. The statistical results obtained reveal that the ensemble CS attained attractive converge speeds as well as robustness. In [33], GWO is hybridized with CS, i.e., GWOCS for the extraction of parameters for different PV cell models situated in different conditions. Zhang et al. [34] developed an ensemble CS algorithm that foremost divides a population into two smaller groups and then utilizes CS and differential evolution (DE) on the derived subgroups independently. The subgroups are free to share useful information by division. Further, the CS and DE algorithms can freely utilize each other's merits to complement their weaknesses. This approach proved to balance the quality of solutions and the computation consumption. In [34], CS is hybridized with a covariance matrix adaptation evolution approach, i.e., CMA-CS to improve the performance of CS in different optimization problems.

Despite the advantages portrayed by the aforementioned hybrid GWO and CS metaheuristics for optimization and feature selection, superior hybrid approaches can be achieved if the single GWO and CS algorithms are improved prior to hybridization. Furthermore, the no-free-lunch (NFL) theorem has logically proved that there has been, is, and will be no single metaheuristic capable of solving all optimization and feature selection problems [33]. While a given metaheuristic can show an attractive performance on specific datasets, its performance might degrade when applied to similar or different types of datasets [34]. Thus, there is still a dire need to improve existing algorithms or develop new ones to solve function optimization problems as well as feature selection problems efficiently.

3. Standard Cuckoo Search (CS)

3.1. Inspiration of CS

3.1.1. The Behavior of Cuckoo Birds. To date, more than a thousand different species of birds are in existence in nature [35]. For most of these species, the female birds lay eggs in nests they have built themselves [36]. However, there exist some types of birds that do not build nests of their own, but instead lay their eggs in other different species' nests, leaving the responsibility of taking care of their eggs to the host birds. The cuckoos are the most famous of these brood parasites [37].

There are three types of brood parasites: intraspecific brood parasites, cooperative breeding, and nest takeover [38].The cuckoo strategy is full of amazing traits; foremost, it replaces one host egg with its own to increase the chances of its egg being hatched by the host bird. Next, it tries to mimic the pattern and color(s) of this host eggs with the aim of reducing the chances of its egg being noticed and discarded by the host bird. It is also important to point out that the timing of laying its egg is amazing since it cleverly selects a nest where a host bird has just laid eggs, implying that the cuckoo's egg will hatch prior to the host eggs. The first action taken by the hatched cuckoo is evicting the host eggs that are yet to hatch out of the nest by blind propelling in order to increase its chances of being fed well by the host bird [37]. In addition, this young cuckoo mimics the call of host chicks thus enhancing more access to the food provided by the host bird [39].

However, if this host bird is able to identify the cuckoo's egg, it can either discard it from the nest or quit this nest to build a completely new nest in a different location.

3.1.2. Le'vy Flights. From literature, many researchers have shown that the behavior of many flying animals, birds, and insects can be demonstrated by a Le'vy flight [40–43]. Le'vy flights are evident when some birds, insects, and animals follow a long path with sudden turns in combination with random-short moves [43].These Le'vy flights have been successfully applied in optimization [41, 43–45]. A Le'vy flight is a random walk characterized with step lengths whose distribution is according to a heavy-tailed probability distribution.

3.2. Cuckoo Search (CS) Algorithm. CS is a metaheuristic swarm-based global optimization based on cuckoos that was proposed by Yang and Deb in 2009.The CS combines the obligate brood parasitic nature of cuckoos with the Le'vy flight existing in fruit flies and some birds [38]. There are three basic idealized rules for the CS, namely:

(i) A female cuckoo lays one egg at a time and puts it in a randomly chosen nest

(ii) The best nests with high-quality eggs (highest fitness/solutions) will carry over to the next generations

(iii) The number of available host nests is kept fixed, and the host bird can discover the egg laid by the female cuckoo (alien egg) with a probability $P_a \in [0, 1]$. Depending on the value of P_a, the host bird can either throw away the alien egg or abandon the nest. An assumption that only a fraction of P_a nests are replaced by new ones

Based on the above rules, an illustration of the CS is shown in Algorithm 1.

3.3. Mathematical Modelling of the Standard CS. Considering Algorithm 1, the standard CS has three major steps [46–48]:

(1) Exploitation (intensification) by the use of Le'vy flight random walk (LFRW)

(2) Exploration (diversification) using biased selective random walk (BSRW)

(3) Elitist scheme via greedy selection

3.3.1. Intensification Using Le'vy Flight Random Walk (LFRW). In this phase, new solutions are generated around the current best solution, which in return enhances the speed of the local search. This phase is achieved via the LFRW that is generally presented in (1) where the step size is derived from the Le'vy distribution.

$$X_{i,\text{gen}+1} = X_{i,\text{gen}} + \alpha \oplus \text{Le}\grave{}\text{vy}(\lambda), \tag{1}$$

where $X_{i,\text{gen}}$ is the i^{th} nest in the gen^{th} generation and $X_{i,\text{gen}+1}$ is a new nest generated by the Le'vy flight. $\oplus$ implies entry-wise multiplications, and α is the step size where $\alpha > 0$ and is formulated in (2). The formula in equation (1) ensures that a new solution will be close to the current best solution.

$$\alpha = \alpha_0 \times \left(X_{i,\text{gen}} - X_{\text{best}}\right), \tag{2}$$

where X_{best} is the current solution and α_0 is a scaler that is set to 0.01 in the standard CSA [38, 49]. $\text{Le}\grave{}\text{vy}\,(\lambda)$ is a random number derived from the Le'vy distribution and is formulated in

$$\text{Le}\grave{}\text{vy}\,(\lambda) \sim \frac{\partial \times \varepsilon}{|\varphi|^{1/\lambda}}, \tag{3}$$

where λ is a constant whose value is 1.5 as suggested by Yang in the standard CS [38]. ε and φ are random numbers derived from a normal distribution whose mean and standard deviation is 1. ∂ is a parameter computed in

$$\partial = \left(\frac{\lceil(1+\lambda) \times \sin\left((\pi \times \lambda)/2\right)}{\lceil\left((1+\lambda/2) \times \lambda \times 2^{((\pi\times\lambda)/2)}\right)}\right)^{1/\lambda}, \tag{4}$$

where $\lceil$ is a gamma function. The final form of Le'vy flight random walk (LFRW) is a combination of equations (1) to (4) as presented in

$$X_{i,\text{gen}+1} = X_{i,\text{gen}} + \alpha_0 \frac{\partial \times \varepsilon}{|\varphi|^{1/\lambda}} \left(X_{i,\text{gen}} - X_{\text{best}}\right) \tag{5}$$

3.3.2. Diversification by the Use of Biased Selective Random Walk (BSRW). In this phase, new solutions are randomly generated in locations far from the current best solution, an approach that ensures that the CSA is not trapped in the local optimum thus enhancing suitable diversity and exploration of the entire search space [48]. This phase of the CSA is achieved by utilizing the BSRW which is efficient in exploring the entire search space especially when it is large since the step size in the Le'vy flight is much longer in the long run [46, 48].

To find new solutions that are far from the current best solution, foremost, a trial solution is obtained by using a mutation of the current best solution and a differential step size from two solutions selected randomly. Then, a new solution is derived from a crossover operator between the current best solution and the two trial solutions [48]. The formulation of the BSRW is given in [47].

$$X_{i,\text{gen}+1} = \begin{cases} X_{i,\text{gen}} + s \times \left(x_{a,j,\text{gen}} - x_{b,j,\text{gen}}\right) \text{ with } P_a \\ X_{i,\text{gen}} \text{ with the remaining } P_a \end{cases}, \tag{6}$$

where a and b are two random indexes, s is a random number in the range [0, 1], and P_a is the probability discovery whose best value is 0.25 [38, 48].

3.3.3. Elitist Scheme via Greedy Selection. After each random walk process, the cuckoo search algorithm utilizes the greedy strategy to select solutions with better fitness values that will be passed to the next generation. This facilitates maintenance of good solutions [48].

4. Grey Wolf Optimization (GWO) Algorithm

GWO is a recent nature-inspired metaheuristic algorithm that was proposed by Mirjalili et al. in 2014 [28, 50, 51]. The GWO imitates both the hunting and leadership traits of the grey wolves. The grey wolves belong to the Canidae family and follow a social hierarchy that is very strict. In most cases, a pack of between 5 and 12 wolves is involved in hunting. To efficiently simulate the leadership hierarchy of the conventional GWO algorithm, four levels are considered: alpha (α), beta (β), delta (δ), and omega (ω). Alpha, which is either a male or female is at the topmost of the hierarchy and is regarded as the leader of the pack. This leader makes all suitable decisions for the pack which are not limited to discipline and order, hunting, sleeping location, and waking-up time for the entire pack. Beta is known to assist the alpha in decision-making, and their main task is the feedback suggestions. Delta behaves like scouts, caretakers, sentinels, hunters, and elders. They control and guide the omega wolves by obeying both the beta and alpha wolves. The omega wolves are the least in the hierarchy and must obey all the other wolves [28, 50, 51].

The GWO algorithm is modelled mathematically in four stages that are described as follows.

4.1. Leadership Hierarchy. The mathematical model of the GWO is anchored on the social hierarchy of the grey wolves. The alpha (α) is considered the best solution in the

```
Begin:
    Initialize $P_a = 0.25$.
    Define objective function $f(x), x = (x_1, x_2, \cdots, x_d)$, where $d$ is the number of dimensions
    Generate initial population of $n$ host bird nests, $X_i (i = 1, 2, \cdots, n)$
    while $g \le g_{\max}$ or any other stopping criteria
        Generate a new cuckoo (solution) randomly via Le'vy flight according to Equation (1)
        Evaluate the fitness of the new cuckoo, $F_i$
        Randomly choose a nest from among the host nests $n$ (For example $j$)
        if $F_i > F_j$then
            Replace nest j by the new cuckoo $i$
        end
        Abandon a fraction of $P_a$ worst nests and generate new ones according to Equation (6)
        Keep best solutions (or those nests with quality solutions)
        Rank these solutions, then keep the current best
    end while
    Report the final best
end
```

Algorithm 1: Pseudocode for the standard CS.

population while beta (β) and delta (δ) are termed as the second and third best solutions, respectively. Lastly, the omega (ω) is assumed as the rest of the solutions in the population [28, 50, 51].

4.2. Encircling the Prey. Equation (7) and equation (8) represent the mathematical model for the wolves' encircling trait [50].

$$\vec{D} = \left|\vec{C}.\vec{X}_p(t) - \vec{X}(t)\right|, \tag{7}$$

$$\vec{X}(t+1) = \vec{X}_p(t) - \vec{A}.\vec{D}, \tag{8}$$

where $\vec{D}$ is the distance between the prey and a given wolf. $\vec{X}$ is the wolf's position vector, and $\vec{X}_p$ depicts the prey's position vector at iteration t. $\vec{A}$ and $\vec{C}$ are random vectors computed as shown in [50].

$$\vec{A} = 2\vec{a}.\vec{r}_1 - \vec{a}, \tag{9}$$

$$\vec{C} = 2.\vec{r}_2, \tag{10}$$

where $\vec{r}_1$ and $\vec{r}_2$ are randomly generated vectors in the range [0, 1] and $\vec{a}$ is a set vector that linearly decreases from 2 to 0 over the iterations.

4.3. Hunting the Prey. In the hunting stage, the alpha is considered the best applicant for the solution while its two assistants (beta and delta) are expected to know the possible location of the prey. Thus, the best three solutions that have been achieved until a given iteration are preserved and are used to compel the remaining wolves in the pack (i.e., omega) to update their positions in the search space consistent with the optimal location.

The mechanism utilized in updating the wolves' positions is given in

$$\vec{X}(t+1) = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3}, \tag{11}$$

where $\vec{X}_1$, $\vec{X}_2$, and $\vec{X}_3$ are defined and computed using

$$\begin{aligned} \vec{X}_1 &= \vec{X}_\alpha - \vec{A}_1.\vec{D}_\alpha, \\ \vec{X}_2 &= \vec{X}_\beta - \vec{A}_2.\vec{D}_\beta, \\ \vec{X}_3 &= \vec{X}_\delta - \vec{A}_3.\vec{D}_\delta, \end{aligned} \tag{12}$$

where $\vec{X}_\alpha$, $\vec{X}_\beta$, and $\vec{X}_\delta$ are the three best wolves (solutions) in the pack at a given iteration t. $\vec{A}_1$, $\vec{A}_2$, and $\vec{A}_3$ are calculated using Equation (9), while $\vec{D}_\alpha$, $\vec{D}_\beta$, and $\vec{D}_\delta$ are calculated using

$$\begin{aligned} \vec{D}_\alpha &= \left|\vec{C}_1.\vec{X}_\alpha - \vec{X}\right|, \\ \vec{D}_\beta &= \left|\vec{C}_2.\vec{X}_\beta - \vec{X}\right|, \\ \vec{D}_\delta &= \left|\vec{C}_3.\vec{X}_\delta - \vec{X}\right|, \end{aligned} \tag{13}$$

where $\vec{C}_1$, $\vec{C}_2$, and $\vec{C}_3$ are calculated based on Equation (10).

4.4. Searching and Attacking the Prey. The grey wolves can only attack the prey when it stops moving. This is modelled mathematically based on vector $\vec{A}$ that is utilized in Equation (9). Vector $\vec{A}$ is comprised of values that span within the range $[-2a, 2a]$, and the value of $\vec{a}$ is decreased from 2 to 0 over the course of iterations using

$$\vec{a} = 2 - \left(\frac{2 \times \text{iter}}{\text{Max}_{\text{iter}}}\right), \tag{14}$$

where iter is the iteration number and Max_{iter} is the optimal total number of iterations.

When $|\vec{A}| < 1$, the wolves are forced to attack the prey, and when $|\vec{A}| > 1$, the wolf diverges out from the current prey.Searching for the prey is the exploration phase while attacking it is the exploitation phase.

5. Excited-Adaptive Cuckoo Search-Intensification Dedicated Grey Wolf Optimization (EACSIDGWO)

In general, effective balancing between diversification (global search) and intensification (local search) in a metaheuristic plays a beneficial and crucial role in achieving excellent performance of an algorithm [52–54]. However, it is difficult to achieve this balance with a single metaheuristic (for example, either using CSA or GWO) [52, 53]. For instance, CSA is efficient at exploring the promising area of the whole search space (diversification) but ineffective at fine-tuning the end of the search space (exploitation/intensification) [55, 56]. On the other hand, GWO is good at intensification (exploitation) but inefficient at diversification (exploration) [32, 57].

For this reason, in trying to enhance mature convergence while ensuring that the required effective balance between diversification and intensification is met, a hybrid algorithm called Excited-Adaptive Cuckoo Search-Intensification Dedicated Grey Wolf Optimization (EACSIDGWO) utilizing the strengths of each algorithm (i.e., CSA's diversification and GWO's intensification abilities) is proposed in this paper. Moreover, the adaptability of the proposed EACSIDGWO is guided innovatively by the complete voltage and current responses of a DC excited RC circuit (whose analysis results in first order differential equations) that finds continual applications in electronics, communications, and control systems [58].

5.1. Adaptive Cuckoo Search (ACS)

5.1.1. Adaptive Step Size via the Complete Voltage Response of the DC Excited RC Circuit. From the details of the standard CS algorithm presented in Section 2, it is evident that the algorithm lacks a criterion to control its step size through the iteration process. Control of the step size is key in guiding the CS algorithm to reach either its global maxima or minima [48, 59].

Inspired by the complete voltage response of a direct current (DC) excited RC circuit which increases with time, a novel mechanism to control the step size is proposed. Contrary to prior research [48, 59] where the step size decays with generations, in this research, the step size grows with generations with the aim of strengthening the diversification (exploration) ability of the CS, which is a component of the proposed EACSIDGWO algorithm.

The solution to the first order differential equation of the direct current-excited RC circuit motivated the formulation of a new variant of ACS in this paper.

The complete voltage response of the RC circuit to a sudden application of a DC voltage source, with the assumption that the capacitor is initially not charged, is given in

$$v(t) = \begin{cases} 0, & t < 0, \\ V_s\left(1 - e^{-\tau/\tau}\right), & t > 0, \end{cases} \tag{15}$$

where $\tau = R * C$ is the time constant, which expresses the rapidity with which this voltage $v(t)$ rises to the value of V_s which is a constant DC voltage source. R and C are the equivalent resistance and capacitance in the circuit, respectively.

Considering the situation when $t > 0$, equation (15) can be rewritten as presented in

$$\begin{aligned} v(t) &= V_s\left(1 - \left(e^{-t}\right)^{\tau}\right), \\ v(t) &= V_s\left(1 - \left(\frac{1}{e^t}\right)^{\tau}\right). \end{aligned} \tag{16}$$

As $t \to \infty$, the component $(1/e^t) \to 0$ forcing $v(t \to \infty) \to V_s$. We adopt this concept, i.e., the exponential growth of $v(t)$ to control the step size of the cuckoo search algorithm by introducing the proposed

$$\text{step}_{\text{gen}+1} = \text{step}_{\text{Max}} \times \left(1 - \left(\frac{\text{gen}_{\text{Max}} - \text{gen}}{\text{gen}_{\text{Max}}}\right)^{\tau}\right), \tag{17}$$

where gen is the current generation (iteration), step_{Max} is the upper bound of the step size step, and gen_{Max} is the maximum number of generations (iterations).

To ensure that the $\text{step}_{\text{gen}+1}$ is proportional to the fitness of a given individual nest within the search space in the current generation, the nonlinear modulation index τ is formulated in

$$\tau_{i,\text{gen}} = \left|\frac{\Big(\Big(\alpha_{\text{nest}f_{\text{gen}}} + \beta_{\text{nest}f_{\text{gen}}} + \delta_{\text{nest}f_{\text{gen}}}\Big)/3\Big) - \text{i}_{\text{nest}f_{\text{gen}}}}{\Big(\Big(\Big(\alpha_{\text{nest}f_{\text{gen}}} + \beta_{\text{nest}f_{\text{gen}}} + \delta_{\text{nest}f_{\text{gen}}}\Big)/3\Big)\Big) - \text{worst}_{\text{nest}f_{\text{gen}}}}\right|, \tag{18}$$

where $\tau_{i,\text{gen}}$ is the nonlinear modulation index for the i^{th} nest in generation gen, $\alpha_{\text{nest}f_{\text{gen}}}$ is the fitness value of the alpha (α) nest (overall best nest) in generation gen, $\beta_{\text{nest}f_{\text{gen}}}$ is the fitness value of the beta (β) nest (2nd best nest) in generation gen, $\delta_{\text{nest}f_{\text{gen}}}$ is the fitness value of the delta (δ) nest (3rd best nest) in generation gen, $\text{i}_{\text{nest}f_{\text{gen}}}$ is the fitness value of the i^{th} nest in generation gen, and $\text{worst}_{\text{nest}f_{\text{gen}}}$ is the fitness value of the worst nest among the remaining omega (ω) nests (i.e., nests whose fitness values are not featured among the top three fitness values).

Thus, equation (17) is further modified as

$$\text{step}_{i,\text{gen}+1} = \text{step}_{\text{Max}} \times \left(1 - \left(\frac{\text{gen}_{\text{Max}} - \text{gen}}{\text{gen}_{\text{Max}}}\right)^{\tau_{i,\text{gen}}}\right), \tag{19}$$

where $\text{step}_{i,\text{gen}+1}$ is the step size for the for the i^{th} nest in generation gen + 1. From equation (19), the step size $\text{step}_{i,\text{gen}+1}$ is nonlinearly increasing from relatively small values to values close to step_{Max}. The reason for proposing a nonlinearly increasing strategy are as follows. Foremost, at the early stages of the proposed EACSIDGWO algorithm, whereby ACS is a component, the population has a higher diversity. A higher diversity implies a stronger ability to explore the global space. Our aim at this point is to accelerate convergence. Therefore, the value of the step size $\text{step}_{i,,\,\text{gen}+1}$ is set to a smaller value.

It is important to point out that the anticipated accelerated convergence is a joint effort attained by foremost setting the $\text{step}_{i,\text{gen}+1}$ of the ACS to a small value at early stages and utilizing the IDGWO (whose details are presented in Section 4.2) whose core task is exploitation.

On the other hand, since the proposed EACSIDGWO algorithm is a hybrid algorithm where the ACS cooperatively works with the IDGWO, all the nests will be attracted to the global optima, i.e., the alpha (α) nest at the later stage. This will compel them to converge prematurely without being given enough room to explore the search space. Such a situation will lead the nests away from a local optimum and encourage diversification. For this reason, the value of the step size $\text{step}_{i,\text{gen}+1}$ is set to a larger value, i.e., step_{Max}. In this paper, the step_{Max} is set to 1.

In other words, our main reason for proposing a nonlinearly increasing step size $\text{step}_{i,\,\text{gen}+1}$ is that its small values at the initial stages of the proposed EACSIDGWO algorithm facilitates "local exploitation" while its larger values in the later stages will facilitate "global exploration".

The ACS can then be modeled as presented in

$$X_{i,\text{gen}+1} = X_{i,\text{gen}} + \text{rand}\ n \times \text{step}_{i,\,\text{gen}+1}. \tag{20}$$

Equation (20) is a formulation of the new search space for the ACS from the current solution.

Moreover, if this step size is considered proportional to the global best solution, then equation (20) can be formulated as given in

$$X_{i,\text{gen}+1} = X_{i,\text{gen}} + \text{rand}\ n \times \text{step}_{i,\,\text{gen}+1} * \left(X_{i,\text{gen}} - X_{g\text{best},\,\text{gen}}\right), \tag{21}$$

where $X_{g\text{best},\boxtimes\text{gen}}$ is the global best solution among all X_i for $i = 1, 2, \cdots, n$ at generation gen, and n is the number of host bird nests.

Thus, from equations (17), (18), (19), (20), (21), it is evident that the diversification ability of the ACS is heightened as the number of generations (gen) approach the maximum number of generations (gen_{Max}). This is because the value of the step size rapidly increases towards the set maximum value of step (step_{Max}).

5.2. Intensification Dedicated Grey Wolf Optimizer (IDGWO)

5.2.1. Nonlinearly Controlling Parameter $\vec{a}$ via the Complete Current Response of the DC Excited RC Circuit. It is evident from Section 4.4 that parameter $\vec{a}$ plays a critical role in balancing the diversification (exploration) and the intensification (exploitation) of a search agent.

A large value of control parameter $\vec{a}$ facilitates diversification while a smaller value of this parameter facilitates intensification. Thus, a suitable selection of the control parameter $\vec{a}$ can enhance a good balance between global diversification (exploration) and local intensification (exploitation).

In the original GWO (described in Section 3), the value of $\vec{a}$ linearly decreases from 2 to 0 (refer to equation (14)). However, the search process of the GWO algorithm is both nonlinear and complicated, which cannot be truly reflected by the linear control strategy of $\vec{a}$ presented in equation (14).

In addition, Mittal et al. [60] proposed that an attractive performance can be attained if parameter $\vec{a}$ is nonlinearly decreased rather than decreased linearly.

Inspired by the complete current response of a direct current (DC) excited RC circuit which increases with time, a novel nonlinear adjustment mechanism of control parameter $\vec{a}$ is formulated in this paper.

The complete current response of the RC circuit to a sudden application of a DC voltage source, with the assumption that the capacitor is initially not charged, is given in

$$i(t) = \frac{V_s}{R}\left(\left(\frac{1}{e^t}\right)^{\tau}\right). \tag{22}$$

As $t \to \infty$, the component $(1/e^t) \to 0$ forcing $i(t \to \infty) \to 0$. We adopt this concept, i.e., the exponential decay of $i(t)$ to formulate a novel improved strategy, i.e., equation (23) to generate the values for control parameter $\vec{a}$.

$$\vec{a}_{i,\text{gen}} = a_o \times \left(\frac{\text{gen}_{\text{Max}} - \text{gen}}{\text{gen}_{\text{Max}}}\right)^{\tau_{i,\text{gen}}}, \tag{23}$$

where gen is the current generation (iteration), a_o is the initial higher value of parameter a and gen_{Max} is the maximum number of generations (iterations). $\tau_{i,\text{gen}}$ is the nonlinear modulation index described earlier by equation (18).

Consequently, vector $\vec{A}$ is computed as given in

$$\vec{A} = 2\vec{a}_{i,\text{gen}} \cdot \vec{r}_1 - \vec{a}_{i,\text{gen}}. \tag{24}$$

Equation (23) is a nonlinear decreasing control parameter for $\vec{a}_{i,\text{gen}}$ whose initial upper limit is equal to the value a_o while its final lower limit is zero.

From the original literature of GWO, the value $|\vec{A}| < 1$ compels the grey wolves to move towards the prey (exploitation) while $|\vec{A}| > 1$ compels them to move away from the prey in search of a fitter prey (exploration). Thus, setting a_o to 1 will always force the wolves to move to the prey which will enable us the dedicated modified GWO algorithm, a component of proposed EACSIDGWO, for intensification.

5.2.2. Enhanced Mature Convergence via a Fitness Value-Based Position-Updating Criterion. Both diversification and intensification are crucial for population-based optimization algorithms [60]. However, from the detailed account of the conventional GWO (refer to Section 3), it is evident that all the other wolves are attracted towards the three leaders α, β, and δ; a scenario that will force the algorithm to converge prematurely without attaining sufficient diversification of the search space. In other words, the conventional GWO is prone to premature convergence.

In reference to the position-updated criterion of GWO described by equation (11), a new candidate individual is obtained by moving the old individual towards the best leader (α wolf), the second best leader (β wolf), and the third best leader (δ wolf). This approach will force all the other grey wolves to crowd in a reduced section of the search space that might be different from the optimal region and without giving them a leeway to escape from such a region. In an effort to overcome this major drawback, in this paper, a scheme that promotes mature converge is devised.

Instead of averaging the values of vectors $\vec{X}_1$, $\vec{X}_2$, and $\vec{X}_3$ (a form of recombining them) as a mechanism of updating the wolves' positions (refer to equation (11)), in this paper, we make full use of the information of their fitness values as a criteria of arriving at new positions for the wolves.

Foremost, the search agents of the populations $\vec{X}_1$, $\vec{X}_2$, and $\vec{X}_3$ are computed as given in

$$\vec{X}_1(i,j) = \vec{X}_\alpha(j) - \vec{A}_1.\vec{D}_\alpha, \tag{25}$$

$$\vec{X}_2(i,j) = \vec{X}_\beta(j) - \vec{A}_2.\vec{D}_\beta, \tag{26}$$

$$\vec{X}_3(i,j) = \vec{X}_\delta(j) - \vec{A}_3.\vec{D}_\delta, \tag{27}$$

where $i = 1, 2, \cdots, n$ and $j = 1, 2, \cdots, d$. n is the population size while d is the dimension of the search space.

Next, the fitness value for each search agent in each of the derived populations, i.e., $\vec{X}_1$, $\vec{X}_2$, and $\vec{X}_3$ is evaluated. Further, a new population with the fittest values is derived from these three populations, i.e., $\vec{X}_1$, $\vec{X}_2$, and $\vec{X}_3$.

Equations (28) and (29) represent the process undertaken to derive this new population.

$$= \max\left(\bigcup_{j=1}^{3} X_j f_{i,\text{gen}}\right), \tag{28}$$

$$X_{i,\text{gen}+1} = \bigcup_{j=1}^{3} \vec{X}_{j_{i,\text{gen}}\text{Index}}, \tag{29}$$

where $\vec{X}_{j_{i,\text{gen}}}$ is vector j computed using search agent i during iteration gen, $X_j f_{i,\text{gen}}$ is the fitness value of vector $\vec{X}_{j_{i,\text{gen}}}$.

5.3. Proposed EACSIDGWO (Continuous Version). We cooperatively combined the proposed adaptive cuckoo search (ACS) and the intensification-dedicated grey wolf optimization (IDGWO) and developed the EACSIDGWO. In the EACSIDGWO algorithm, the ACS is actively involved in intensification (exploitation) during the early stage when the population has higher diversity and diversification at later stages. On the other hand, the IDGWO is only actively involved in intensification in all the stages of the proposed algorithm. By doing so, an effective balance between diversification and intensification is achieved. In addition, mature convergence is enhanced which in the end leads to high-quality solutions.

6. Proposed EACSIDGWO (Binary Version)

Selection of features is binary by nature [61]. Therefore, the proposed EACSIDGWO algorithm cannot be utilized in selection of features without further modifications.

In the proposed EACSIDGWO algorithm, the new positions of the search agents will have continuous solutions, which must be converted into corresponding binary values.

In this paper, this conversion is achieved by foremost applying squashing of the continuous solutions in each dimension using a sigmoid (S-shaped) transfer function [61]. This will compel the search agents to move into a binary search space as depicted by equation (31).

$$S = \frac{1}{1 + e^{-10\left(X^d{}_{i,\text{gen}} - 0.5\right)}}, \tag{30}$$

where $X^d{}_{i,\text{gen}}$ is a continuous-valued position of the i^{th} search agent in the d^{th} dimension during generation gen.

The output S of the sigmoid transfer function is still a continuous value, and thus, it has to be the threshold to reach the binary-value one. Normally, the sigmoid function maps smoothly the infinite input to a finite output [61]. To arrive at the binary solution when a sigmoid function is used, the commonly stochastic threshold is applied as presented in

$$y^d{}_{i,\text{gen}} = \begin{cases} 0 \text{ if } \text{rand} < S, \\ 1 \text{ if } \text{rand} \geq S, \end{cases} \tag{31}$$

$$Y_{i,\text{gen}} = \bigcup_{i=1}^{n} y^d{}_{i,\text{gen}}, \tag{32}$$

where $y^d{}_{i,\text{gen}}$ is the binary updated position at generation gen in the d^{th} dimension and rand is a random number drawn from a uniform distribution $\in [0, 1]$. $Y_{i,\text{gen}}$ is the equivalent binary vector of the i^{th} search agent at generation gen.

Using this approach, the original solutions remain in the continuous domain of the proposed EACSIDGWO algorithm and can be converted to binary when the need arises.

The pseudocode of the binary version of the proposed EACSIDGWO algorithm is presented in Algorithm 3.

```
Begin:
Initialize population size $n$, parameter $a$, coefficient vectors $A$, $C$ and maximum number of iterations $\text{Max}_{iter}$
Set $t := 0$ {Counter initialization}
for ($i = 1$: $i \leq n$) do
    Randomly generate an initial population $X_i(t)$
  Evaluate the fitness function of each agent (solution) i.e. $f(X_i)$
end for
Assign the values of the $1^{st}$,$2^{nd}$ and $3^{rd}$ best solutions i.e. $X_\alpha$, $X_\beta$ and $X_\delta$, respectively
repeat
for ($i = 1$: $i \leq n$) do
  Update each search agent in the population using Equation (11)
Decrease the value of $a$ using Equation (14)
Update the coefficients $A$ and $C$ as shown in Equation (9) and Equation (10), respectively
Evaluate the fitness function of each search agent (vector) $f(X_i)$
end for
Update the vectors $X_\alpha$, $X_\beta$ and $X_\delta$
Set $t = t + 1$ {Iteration counter increasing}
Until ($t < \text{Max}_{iter}$) {termination criteria satisfied}
Report the best solution $X_\alpha$
```

ALGORITHM 2: Pseudocode for the GWO.

7. Experimental Methodology

In this section, detailed accounts of the biomedical datasets, evaluation metrics, proposed fitness function, and the parameter setting for the considered metaheuristic algorithms are outlined.

7.1. Considered Biomedical Datasets. To validate the performance of the considered metaheuristic algorithms, six benchmark biomedical datasets extracted from the UCI Irvine Machine [62] were utilized. Each dataset has two classes, and the performance of each of these algorithms is evaluated based on its ability to classify these classes correctly. Details of these datasets are given in Table 1.

7.2. Evaluation Metrics. For the considered feature selection problem, the following evaluation metrics were utilized to compare the performance of each considered feature selection technique.

Average Accuracy (Avg_Acc). It is one of the commonly used classification metric that represents the number of correctly classified instances by using a particular feature set. The mathematical formulation of this metric is given in Equation (33).

$$\text{Avg\_Acc} = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{k}\sum_{j=1}^{k}\text{Acc}_j, \tag{33}$$

where N is the number of times (runs) a given metaheuristic algorithm is run, k represents the number of folds utilized, and Acc_j is the accuracy reported during fold j. Acc_j is defined in equation (34).

$$\text{Acc}_j = \frac{\text{TP}_j + \text{TN}_j}{\text{TP}_j + \text{TN}_j + \text{FP}_j + \text{FN}_j}, \tag{34}$$

where TP and FN denote the number of positive samples in fold j that are accurately and falsely predicted, respectively, and TN and FP represent the number of negative samples in the same fold that are predicted accurately and wrongly, respectively [63].

Average Feature Length (Avg_NFeat). This metric characterizes the average length of selected features to the total number of features in the dataset. Equation (35) gives its mathematical formulation.

$$\text{Avg\_NFeat} = \frac{1}{N}\sum_{i=1}^{N}\text{Sel\_Feat}_i, \tag{35}$$

where Sel_Feat_i is the number of selected features in the testing dataset during run i.

Minimum Accuracy (Min_Acc). It is the least value of accuracy reported during N runs. Equation (36) depicts its formulation.

$$\text{Min \_Acc} = \min\left(\bigcup_{j=1}^{N}\text{Avg\_crossAcc}_j\right), \tag{36}$$

where Avg_crossAcc_i is given by

$$\text{Avg\_crossAcc}_i = \frac{1}{k}\sum_{j=1}^{k}\text{Acc}_j. \tag{37}$$

Maximum Accuracy (Max_Acc). It is the largest value of accuracy reported during N runs. Its mathematical formulation is given by

$$\text{Max \_Acc} = \max\left(\bigcup_{j=1}^{N}\text{Avg}_{\text{crossAcc}\,j}\right). \tag{38}$$

```
Input: labelled biomedical dataset $D$, $MaxIter$, ACS and IDGWO parameters value, number of host bird nests ($n$), number of dimensions (features) $d$, Lower bound ($L_b$) and Upper bound ($U_b$)
Output: Best Fitness, Best Search Agent
1  for each nest i (i =1, 2...n) do
2  for each dimension j(j =1,2,…,d) do
3     $X^j_{i,0}$=random number drawn from $[L_b, U_b]$
4     end
5  Convert continuous values of $X_{i,0}$ to binary using Eq. (31), (32) and (36)
6  Train a classifier to evaluate the accuracy of the equivalent binary vector of $X_{i,0}$ and store the value in $Xf_{i,0}$
7  end
8  $[\sim, Index] = Sort\ (Xf_0, 'descend')$
9  $\alpha_{nestf\,0} = Xf_0(Index(1))$
10  $\beta_{nestf\,0} = Xf_0(Index(2))$
11  $\delta_{nestf\,0} = Xf_0(Index(3))$
12  $\text{worst}_{nestf\,0} = Xf_0(Index(n))$
13  $\alpha_{nest0} = X_0(Index(1))$
14  $\beta_{nest0} = X_0(Index(2))$
15  $\delta_{nest0} = X_0(Index(3))$
16  While ($gen \le MaxIter$)
17     for each nest i (i =1, 2...n) do
18        Calculate $\tau_{i,gen}$ and $step_{i,⊠gen+1}$ using
          Eq. (18) and (19), respectively
19        Generate a new cuckoo nest $X_{i,gen+1}$
          using Eq. (21)
20  Convert continuous values of $X_{i,gen+1}$ to binary using Eq. (31), (32) and (36)
21  Train a classifier to evaluate the accuracy of the equivalent binary vector of $X_{i,gen+1}$ and store the value in $Xf_{i,gen+1}$
22  if( $Xf_{i,gen+1} > Xf_{i,0}$) then
23  $Xf_{i,0}$=$Xf_{i,gen+1}$
24  $X_{i,0}$=$X_{i,gen+1}$
25  end
26  end
27  Repeat step 8 to 15
28  for each nest i (i =1, 2...n) do
29  Calculate $\tau_{i,gen}$ and $a_{i,\,gen}$ using Eq. (18) and (23), respectively
30  for each dimension j(j =1,2,…,d) do
31  Calculate coefficients $A$ and $C$ as shown in Equation (24) and Equation (10), respectively
32  Compute vectors $\vec{X}_{1_{i,gen}}(j)$, $\vec{X}_{2_{i,gen}}(j)$ and $\vec{X}_{3_{i,gen}}(j)$ using Equations (25), (26) and (27), respectively.
33  end
34  Convert continuous values of $\vec{X}_{1_{i,gen}}$, $\vec{X}_{2_{i,gen}}$ and $\vec{X}_{3_{i,gen}}$ to binary using Eq. (31), (32) and (36)
35  Consecutively, train a classifier to evaluate the accuracies of the equivalent binary vectors of $\vec{X}_{1_{i,gen}}$, $\vec{X}_{2_{i,gen}}$ and $\vec{X}_{3_{i,gen}}$ and store the value in $X_1f_{i,gen}$, $X_2f_{i,gen}$ and $X_3f_{i,gen}$ , respectively.
36  Determine $\vec{X}_{i,gen+1}$ using equations (28) and (29), respectively
37  end
38  Repeat step 8 to 15
39  Abandon a fraction of $P_a$ worst nests and generate new ones according to Equation (6)
40  Keep best solutions(or those nests with quality solutions)
41  Repeat step 8 to 15
  end
42  Best Search Agent= $\alpha_{nest0}$
43  Best Fitness=$\alpha_{nestf\,0}$
```

Algorithm 3. Pseudocode for the EACSIDGWO (Binary Version).

Table 1: Considered biomedical datasets.

Dataset	Number of features	Number of cases
Breast Cancer Wisconsin (prognosis)	33	198
Breast Cancer Wisconsin (diagnostic)	30	569
SPECTF Heart	44	267
Ovarian Cancer	4000	216
CNS	7129	60
Colon	2000	62

*Maximum Features Selected (Max_NFeat).*It is the largest number of selected features during N runs. Equation (39) gives its mathematical formulation.

$$\text{Max \_NFeat} = \max\left(\bigcup_{i=1}^{N}\text{Sel}_{\text{Feat}\,i}\right). \tag{39}$$

Minimum Features Selected (Min_NFeat). It is the least number of selected features during N runs. Equation (40) gives its mathematical formulation.

$$\text{Min \_NFeat} = \min\left(\bigcup_{i=1}^{N}\text{Sel}_{\text{Feat}\,i}\right). \tag{40}$$

7.3. Evaluation of the Classifier Performance. Since the support vector machine classifier has already made immense contributions in the field of microarray-based cancer classification [63], it was adopted in this paper to evaluate the classification accuracy using the selected subset of features returned by the various considered metaheuristic feature selection approaches. The Matlab fitcsvm function that trains and cross-validates an SVM model was adopted in this paper. We specified the kernel scale parameter to "auto" to allow the function to select the appropriate scale factor using a heuristic search.

With the SVM classifier, the data items are mapped points in an n – dimensional feature space (i.e., n = number of features) and each feature's value is a value of a given coordinate. The final output of this classifier is an optimal hyperplane which can be used to classify new cases [17, 63].

However, the performance of the SVM classifier is highly dependent on the selection of its kernel function [17, 63]. A reason why experiments were conducted using various kernels in this paper.

Selecting a suitable kernel is both dataset and problem specific and selected experimentally [17, 63]. Based on the conducted experiments, suitable kernel functions were selected for the considered datasets. The considered datasets and their suitable kernel functions are presented in Table 2.

More information of selecting suitable SVM kernel functions is presented in [63].

7.4. Fitness Function. The main aim of a feature selection exercise is to discover a subset of features from the whole set of existing features in a given dataset such that the considered optimization algorithm is able to achieve the highest possible accuracy using that subset. For instance, in datasets with many features (attributes), the objective is to minimize the number of selected features while improving the classification accuracy of the feature selection approach.

In classifications tasks, there exist higher chances that two feature subsets containing a different number of features will have the same accuracy [17]. However, if a subset with a large number of features is discovered earlier by a given optimization algorithm, it is likely that the one with least features will be ignored [17].

In trying to overcome this challenge, a fitness function proposed in [17] to evaluate the classification performance of optimization algorithms for feature selection tasks is adopted. This fitness function is given in

$$\text{Fit} = \alpha * \frac{|R|}{|N|} - \beta * \text{Avg\_crossAcc}_i, \tag{41}$$

where $|N|$ represents the total number of features within a given dataset, $|R|$ represents the number of selected features during run i, and Avg_crossAcc_i is the average cross-validation accuracy reported during run i (refer to Equation (37)). β and α are two weights corresponding to the significance of the classification quality and the subset length, respectively. In this paper, β is set to 0.8 and $\alpha = 0.2$ as adopted from [17].

It is important to point out that both terms are normalized by division by their largest possible values; i.e., the number of selected features $|R|$ is divided by the total number of features $|N|$, and average accuracy Avg_crossAcc_i is divided by the value 1.

7.5. Parameter Setting for the Considered Feature Selection Techniques. The performance of the proposed EACSIDGWO algorithm was compared to those of extended binary cuckoo search (EBCS), binary ant-colony optimization (BACO), binary genetic algorithm (BGA), and binary particle swarm optimization (BPSO) that were reported earlier in [17].

Table 3 indicates the selected parameter values for both the proposed BEACSIDGWO algorithm and each of the other algorithms as reported in [17].

To be consistent with the setup proposed in [17], the population size for the proposed EACSIDGWO was set to 30. Then, the algorithm was run 10 times to perform the feature selection task for each considered dataset. In addition, each run terminated when 10000 fitness function evaluations was attained. This approach allowed the proposed algorithm to utilize the fitness function at an equal number of times.

In this paper, all the experiments were conducted using Matlab 2017 running on Windows 10 operating system on a HP desktop with Intel® Core™ i7-3770CPU @ 3.4 GHZ with 12.0 GB of RAM.

Table 2: Selection of suitable kernel functions.

Dataset	Kernel function
Breast Cancer Wisconsin (prognosis)	Radial basis function (RBF)
Breast Cancer Wisconsin (diagnostic)	Radial basis function (RBF)
SPECTF Heart	Radial basis function (RBF)
Ovarian Cancer	Linear function
CNS	Linear function
Colon	Linear function

Table 3: Selection of parameter values for the considered approaches.

Algorithm	Parameter values
EACSIDGWO	$\text{step}_{\text{Max}} = 1$, $a_o = 1$, and $P_a = 0.25$
EBCS	$N_{\text{mut}} = 10$, $\lambda = 1$, $\alpha = 1$, and $P_a = 0.4$
BACO	$\Gamma_{\text{initial}} = 0.1$, $\alpha = 1$, and $p = 0.1$
BGA	$M_r = 0.1, C_r = 0.1$
BPSO	$C_1 = 1, C_2 = 2, \omega_{\text{initial}} = 0.9, \omega_{\text{vary-for}} = 0.9$

8. Results and Discussion

To examine the diversification and intensification of the proposed EACSIDGWOA, detailed comparative study is presented in this section.

The efficiency and the optimization performance of the proposed algorithm have been verified by comparing and analyzing its results with those of four other state-of-the-art optimization algorithms.

The experimental classification results have been probed through statistical tests, comparative analysis, and ranking methods.

Tables 4–9 provide the performance of all the considered optimization approaches for feature selection using the datasets described in Section 7.1. It is important to point out that the best result achieved in each column for all the considered biomedical datasets is highlighted in bold while the worst is italicized.

To prove that the proposed EACSIDGWO is superior over the other four-optimization algorithms, Wilcoxon rank-sum test, i.e., a nonparametric statistical test, is also performed. The statistical results for the p, h, and z values obtained from the pairwise comparisons of the four groups are tabulated in Table 10. Tables 11 and 12 present a comparison of the overall ranking of the results obtained by the considered algorithms.

8.1. Discussion

8.1.1. Investigation of the Obtained Classification Results. From Tables 4–9, the following observations can be made.

(i) The proposed EACSIDGWO algorithm outperformed all the other considered algorithms in terms of classification accuracy for all the utilized datasets. It recorded the highest classification accuracy on the three highly dimensioned datasets (i.e., Ovarian, CNS, and Colon) as well as the remaining three small sample-sized datasets. This promising performance is largely attributed to the cooperative exploitation conducted by ACS and IDGWO components of the proposed algorithm during the early generations, as well as the single-handed exploitation and exploration by IDGWO and ACS, respectively, at later generations

(ii) For four datasets, i.e., Ovarian, Heart, CNS and Colon, the proposed algorithm attained a value for Avg_Acc that is larger than the value for Max _Acc attained by the EBCS. EBCS is a variant of cuckoo search, which is a component of the proposed EACSIDGWO algorithm. This superior performance proves the competency of the proposed approach to efficiently determine the optima within the search space

(iii) With regard to the average feature length (Avg_NFeat), the proposed B-EACSIDGWO algorithm demonstrated a superior performance by selecting the least number of features compared to the other algorithms. According to the results reported in Tables 4–9, the proposed algorithm performed better on all the considered datasets

In comparison with the original number of features in the considered datasets, there is a notable reduction in the number selected features by the proposed approach. For instance, the actual number of features in ovarian cancer, CNS, and Colon cancer datasets is 4000, 7129, and 2000, respectively, whereas the number of selected features by the proposed EACSIDGWO is 274.8, 1208.1, and 538.5, respectively. This clearly indicates that the proposed algorithm is able to reduce the number of features as well as locate the most significant optimal feature subsets. The strength of the proposed EACSIDGWO lies in its well-formulated algorithm (refer to Section 5) that enhances both its diversification and intensification capabilities which enables it to eliminate redundant (noninformative) attributes and then actively searches within the high-performance regions of the feature space.

8.1.2. Statistical Analysis. The superiority of the proposed EACSIDGWO algorithm has been verified via Wilcoxon rank-sum test, i.e., a nonparametric test with a significance level of 5%. The results obtained for the pairwise comparison of the four groups are presented in Table 10. Observations from Table 10 reveal the statistical significance of the obtained experimental results for all the considered datasets. This clearly indicates that the proposed approach has an attractive performance in relation to the other four approaches. Thus, the overall statistical results by our algorithm are highly significant from the results of the four algorithms for all the considered datasets.

8.1.3. Ranking Methods. Tables 11 and 12 outline the detailed ranking of all the considered algorithms with their respective comparative analysis. The ranking is based on maximum

Table 4: Experimental results for the Ovarian Cancer dataset.

Algorithm		Accuracy			Number of features	
	Max _Acc	Min _Acc	Avg_Acc	Max _NFeat	Min _NFeat	Avg_NFeat
EACSIDGWO	**1.000**	**1.000**	**1.000**	**292**	**264**	**274.8**
EBCS	*0.991*	0.991	0.991	1855	1747	1811.6
BACO	*0.991*	*0.986*	*0.990*	*1971*	*1912*	*1945.7*
BGA	*0.991*	0.991	0.991	1830	1755	1887.3
BPSO	*0.991*	*0.986*	*0.990*	1913	1777	1857

Values in bold represent the best result, and values in italic denote the worst in each column, respectively.

Table 5: Experimental results for the Breast Cancer Wisconsin (Diagnostic) dataset.

Algorithm		Accuracy			Number of features	
	Max _Acc	Min _Acc	Avg_Acc	Max _NFeat	Max _NFeat	Avg_NFeat
EACSIDGWO	0.977	**0.974**	**0.975**	**3**	**3**	**3**
EBCS	**0.981**	**0.974**	0.973	4	**3**	3.1
BACO	*0.972*	*0.960*	*0.969*	*8*	*6*	*7*
BGA	0.975	0.965	0.972	6	**3**	3.6
BPSO	**0.981**	0.963	0.974	*8*	**3**	5.4

Values in bold represent the best result, and values in italic denote the worst in each column, respectively.

Table 6: Experimental results for the Breast Cancer Wisconsin (Prognosis) dataset.

Algorithm		Accuracy			Number of features	
	Max _Acc	Min _Acc	Avg_Acc	Max _NFeat	Min _NFeat	Avg_NFeat
EACSIDGWO	**0.879**	**0.864**	**0.873**	**7**	**3**	**5.6**
EBCS	0.874	0.828	0.856	8	4	6.2
BACO	*0.818*	*0.768*	*0.794*	*12*	*5*	*8.4*
BGA	0.874	0.793	0.843	10	4	6.5
BPSO	*0.848*	*0.798*	0.821	11	4	8.3

Values in bold represent the best result, and values in italic denote the worst in each column, respectively.

Table 7: Experimental results for the SPECTF Heart dataset.

Algorithm		Accuracy			Number of features	
	Max _Acc	Min _Acc	Avg_Acc	Max _NFeat	Min _NFeat	Avg_NFeat
EACSIDGWO	**0.884**	**0.861**	**0.875**	**6**	**3**	**4.5**
EBCS	0.873	0.846	0.861	8	5	6.2
BACO	*0.846*	*0.813*	*0.831*	*15*	*10*	*12.1*
BGA	**0.884**	0.846	0.866	11	4	8.4
BPSO	0.865	0.846	0.854	*15*	9	10.9

Values in bold represent the best result, and values in italic denote the worst in each column, respectively.

Table 8: Experimental results for the CNS dataset.

Algorithm		Accuracy			Number of features	
	Max _Acc	Min _Acc	Avg_Acc	Max _NFeat	Min _NFeat	Avg_NFeat
EACSIDGWO	**0.767**	**0.700**	**0.718**	**1623**	**807**	**1208.1**
EBCS	*0.667*	0.667	0.667	3490	3391	3446.7
BACO	*0.667*	*0.650*	*0.660*	*3589*	3432	*3522.9*
BGA	0.683	0.667	0.668	3566	*3438*	3489.7
BPSO	*0.667*	0.667	0.667	3547	3359	3474.3

Values in bold represent the best result, and values in italic denote the worst in each column, respectively.

Table 9: Experimental results for the Colon dataset.

Algorithm		Accuracy			Number of features	
	Max _Acc	Min _Acc	Avg_Acc	Max _NFeat	Min _NFeat	Avg_NFeat
EACSIDGWO	**0.919**	**0.887**	**0.905**	**637**	**397**	**538.5**
EBCS	0.903	0.871	0.887	*1016*	*961*	*988.7*
BACO	0.903	0.871	*0.881*	1002	932	976
BGA	*0.887*	0.871	0.882	1003	944	962.8
BPSO	*0.887*	*0.855*	0.879	1003	933	971.2

Values in bold represent the best result, and values in italic denote the worst in each column, respectively.

Table 10: Using Wilcoxon's rank-sum test at $p = 0.05$ to compare EACSIDGWO with other algorithms.

Dataset	Wilcoxon's rank-sum test	EBCS vs EACSIDGWO	BACO vs EACSIDGWO	BGA vs EACSIDGWO	BPSO vs EACSIDGWO
Ovarian Cancer	p value	0.000181651	0.000181651	0.000182672	0.000181651
	h value	1.000000000	1.000000000	1.000000000	1.000000000
	z value	3.743255786	3.743255786	3.741848283	3.743255786
Breast Cancer Wisconsin (diagnostic)	p value	0.022591996	0.000146767	0.017044126	0.000582314
	h value	1.000000000	1.000000000	1.000000000	1.000000000
	z value	2.28026466	3.796476695	2.38575448	3.439721266
Breast Cancer Wisconsin (prognosis)	p value	0.000730466	0.0001707	0.00073729	0.000174624
	h value	1.000000000	1.0000000	1.00000000	1.000000000
	z value	3.377881495	3.758843896	3.375323463	3.753152986
SPECTF Heart	p value	0.000321376	0.000176611	0.000176611	0.000177611
	h value	1.000000000	1.000000000	1.000000000	1.000000000
	z value	3.597430949	3.750317207	3.750317207	3.748901726
CNS	p value	0.000182672	0.000182672	0.000182672	0.000182672
	h value	1.000000000	1.000000000	1.000000000	1.000000000
	z value	3.741848283	3.741848283	3.741848283	3.741848283
COLON	p value	0.000182672	0.000182672	0.000182672	0.000181651
	h value	1.000000000	1.000000000	1.000000000	1.000000000
	z value	3.741848283	3.741848283	3.741848283	3.743255786

accuracy (Max _Acc), minimum accuracy (Min _Acc), average accuracy (Avg_Acc), maximum number of selected features (Max _NFeat), minimum number of selected features (Min _NFeat), and average number of selected features (Avg_NFeat). From the ranking, it is evident that the proposed EACSIDGWO algorithm obtained the best values in all these measures for all the datasets. Considering the final ranks, the proposed algorithm attained an attractive performance whose overall rank value is 37.This clearly reveals the superiority of EACSIDGWO algorithm in relation to the four state-of-the-art algorithms.

9. Conclusion

This paper proposed a new hybrid Excited- (E-) Adaptive Cuckoo Search- (ACS-) Intensification Dedicated Grey Wolf Optimizer (IDGWO), i.e., EACSIDGWO algorithm to solve the feature selection problem in biomedical science. In the proposed algorithm, the concept of the complete voltage and current responses of a direct current (DC) excited resistor capacitor (RC) circuit are innovatively utilized to make the step size of ACS and the nonlinear control strategy of parameter $\vec{a}$ of the IDGWO adaptive. Since the population has a higher diversity during early stages of the proposed algorithm, both the ACS and IDGWO are jointly utilized to attain accelerated convergence. However, to enhance mature convergence while striking an effective balance between exploitation and exploration in later stages, the role of ACS is switched to global exploration while the IDGWO is still left conducting the local exploitation. In order to test the efficiency of the proposed EACSIDGWO as a feature selector, six standard biomedical datasets from the University of California at Irvine (UCI) repository were utilized. The experimental results obtained prove that the proposed algorithm is superior to the state-of-the-art feature selection techniques, i.e., BACO, BGA, BPSO, and EBCSA in attaining a good learning from fewer instances and optimal feature selection from information-rich biomedical data, all these

Table 11: Overall ranking of considered algorithms.

		Datasets									
Algorithm	Measures	Ovarian Cancer	Breast Cancer Wisconsin (diagnostic)	Breast Cancer Wisconsin (prognosis)	SPECTF Heart	CNS	Colon	Sum of ranks	Overall rank	Total sum	Final ranks
EACSIDGWO	Max _Acc	1	2	1	1	1	1	7	1	37	1
	Min _Acc	1	1	1	1	1	1	6	1		
	Avg_Acc	1	1	1	1	1	1	6	1		
	Max _NFeat	1	1	1	1	1	1	6	1		
	Min _NFeat	1	1	1	1	1	1	6	1		
	Avg_NFeat	1	1	1	1	1	1	6	1		
EBCS	Max _Acc	2	1	2	3	3	2	13	2	84	2
	Min _Acc	2	1	2	2	2	2	11	2		
	Avg_Acc	2	2	2	3	3	2	14	2		
	Max _NFeat	3	2	2	2	2	4	15	2		
	Min _NFeat	2	1	2	3	3	5	16	2		
	Avg_NFeat	2	2	2	2	2	5	15	2		
BACO	Max _Acc	2	4	4	4	2	2	18	4	138	5
	Min _Acc	3	4	5	3	3	2	20	4		
	Avg_Acc	3	5	5	5	4	4	26	5		
	Max _NFeat	5	4	5	4	5	2	25	5		
	Min _NFeat	5	2	3	5	3	2	20	3		
	Avg_NFeat	5	5	5	5	5	4	29	5		

Table 12: Overall ranking of considered algorithms.

		Datasets									
Algorithm	Measures	Ovarian Cancer	Breast Cancer Wisconsin (diagnostic)	Breast Cancer Wisconsin (prognosis)	SPECTF Heart	CNS	Colon	Sum of ranks	Overall rank	Total sum	Final ranks
BGA	Max _Acc	2	3	2	1	2	3	13	2	95	3
	Min _Acc	2	2	5	2	2	2	15	3		
	Avg_Acc	2	4	3	2	2	3	16	3		
	Max _NFeat	2	3	3	3	4	3	18	3		
	Min _NFeat	3	1	2	2	4	4	16	2		
	Avg_NFeat	3	3	3	3	3	2	17	3		
BPSO	Max _Acc	2	1	3	3	3	3	15	3	110	4
	Min _Acc	3	3	2	2	2	3	15	3		
	Avg_Acc	3	2	4	4	3	5	21	4		
	Max _NFeat	4	4	4	4	3	3	22	4		
	Min _NFeat	4	1	2	4	2	3	16	2		
	Avg_NFeat	3	4	4	4	3	3	21	4		

while maintaining a high classification accuracy of the utilized data. In the future, utilizing this hybrid algorithm as a filter-feature selection approach seeking to evaluate the generality of the selected features will be a valuable contribution.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no competing interests regarding the publication of this paper.

Acknowledgments

This work is fully supported by the African Development Bank (AfDB), through the Ministry of Education, Kenya Support for Capacity Building.

References

[1] J. Han, J. Pei, and M. Kamber, *Data Mining: Concepts and Techniques*, Elsevier, Amsterdam, The Netherlands, 2011.

[2] H. Liu and H. Motoda, *Feature Selection for Knowledge Discovery and Data Mining*, vol. 454, Springer, 2012.

[3] M. Bennasar, Y. Hicks, and R. Setchi, "Feature selection using joint mutual information maximisation," *Expert Systems with Applications*, vol. 42, no. 22, pp. 8520–8532, 2015.

[4] M. Dash and H. Liu, "Feature selection for classification," *Intelligent Data Analysis*, vol. 1, no. 3, pp. 131–156, 1997.

[5] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003.

[6] H. Liu and Z. Zhao, "Manipulating data and dimension reduction methods: feature selection," in *Encyclopedia Complexity Systems Science*, pp. 5348–5359, Springer, New York, NY, USA, 2009.

[7] H. Liu, H. Motoda, R. Setiono, and Z. Zhao, "Feature selection: an ever evolving frontier in data mining," in *Proc.4thWorkshop Feature Selection Data Mining*, pp. 4–13, Hyderabad, India, 2010.

[8] A. Zarshenas and K. Suzuki, "Binary coordinate ascent: an efficient optimization technique for feature subset selection for machine learning," *Knowledge-Based Systems*, vol. 110, pp. 191–201, 2016.

[9] E.-G. Talbi, *Metaheuristics: from design to implementation*, vol. 74, Wiley, Hoboken, NJ, USA, 2009.

[10] H. Liu and H. Motoda, *Feature extraction, construction and selection: a data mining perspective*, vol. 453, Springer, 1998.

[11] R. Bello, Y. Gomez, A. Nowe, and M. M. Garcia, "Two-step particle swarm optimization to solve the feature selection problem," in *Seventh International Conference on Intelligent Systems Design and Applications (ISDA 2007)*, pp. 691–696, Rio de Janeiro, Brazil, 2007.

[12] E. Emary, H. M. Zawbaa, and A. E. Hassanien, "Binary grey wolf optimization approaches for feature selection," *Neurocomputing*, vol. 172, pp. 371–381, 2016.

[13] Q. Al-Tashi, H. Rais, and S. Jadid, "Feature selection method based on grey wolf optimization for coronary artery disease classification," in *Proc. Int. Conf. Reliable Inf. Commun. Technol.*, pp. 257–266, Kuala Lumpur, Malaysia, 2018.

[14] M. M. Kabir, M. Shahjahan, and K. Murase, "A new local search based hybrid genetic algorithm for feature selection," *Neurocomputing*, vol. 74, no. 17, pp. 2914–2928, 2011.

[15] S. Kashef and H. Nezamabadi-pour, "An advanced ACO algorithm for feature subset selection," *Neurocomputing*, vol. 147, pp. 271–279, 2015.

[16] E. ZorarpacI and S. A. Özel, "A hybrid approach of differential evolution and artificial bee colony for feature selection," *Expert Systems with Applications*, vol. 62, pp. 91–103, 2016.

[17] S. Salesi and G. Cosma, "A novel extended binary cuckoo search algorithm for feature selection," in *2017 2nd International Conference on Knowledge Engineering and Applications (ICKEA)*, pp. 6–12, London, UK, 2017.

[18] M. M. Mafarja, D. Eleyan, I. Jaber, A. Hammouri, and S. Mirjalili, "Binary dragonfly algorithm for feature selection," in *2017 International Conference on New Trends in Computing Sciences (ICTCS)*, pp. 12–17, Amman, Jordan, 2017.

[19] A. H. Gandomi and X.-S. Yang, "Chaotic bat algorithm," *Journal of Computational Science*, vol. 5, no. 2, pp. 224–232, 2014.

[20] N. Abd-Alsabour, "A review on evolutionary feature selection," in *2014 European Modelling Symposium*, Pisa, Italy, 2014.

[21] S. Mirjalili, "SCA: a sine cosine algorithm for solving optimization problems," *Knowledge-Based Systems*, vol. 96, pp. 120–133, 2016.

[22] H. Ma and D. Simon, "Blended biogeography-based optimization for constrained optimization," *Engineering Applications of Artificial Intelligence*, vol. 24, no. 3, pp. 517–525, 2011.

[23] N. Almugren and H. Alshamlan, "A survey on hybrid feature selection methods in microarray gene expression data for cancer classification," *IEEE Access*, vol. 7, pp. 78533–78548, 2019.

[24] R. Sindhu, R. Ngadiran, Y. M. Yacob, N. A. Hanin Zahri, M. Hariharan, and K. Polat, "A hybrid SCA inspired BBO for feature selection problems," *Mathematical Problems in Engineering*, vol. 2019, Article ID 9517568, 18 pages, 2019.

[25] P. J. Gaidhane and M. J. Nigam, "A hybrid grey wolf optimizer and artificial bee colony algorithm for enhancing the performance of complex systems," *Journal of Computer Science*, vol. 27, pp. 284–302, 2018.

[26] H. M. Zawbaa, E. Emary, C. Grosan, and V. Snasel, "Large-dimensionality small-instance set feature selection: a hybrid bio- inspired heuristic approach," *Swarm and Evolutionary Computation*, vol. 42, pp. 29–42, 2018.

[27] A. A. Alomoush, A. A. Alsewari, H. S. Alamri, K. Aloufi, and K. Z. Zamli, "Hybrid harmony search algorithm with grey wolf optimizer and modified opposition-based learning," *IEEE Access*, vol. 7, pp. 68764–68785, 2019.

[28] S. Arora, H. Singh, M. Sharma, S. Sharma, and P. Anand, "A new hybrid algorithm based on grey wolf optimization and crow search algorithm for unconstrained function optimization and feature selection," *IEEE Access*, vol. 7, pp. 26343–26361, 2019.

[29] B. N. Gohil and D. R. Patel, "A hybrid GWO-PSO algorithm for load balancing in cloud computing environment," in *2018 Second International Conference on Green Computing and Internet of Things (ICGCIoT)*, pp. 185–191, Bangalore, India, 2018.

[30] A. Zhu, C. Xu, Z. Li, J. Wu, and Z. Liu, "Hybridizing grey wolf optimization with differential evolution for global optimization and test scheduling for 3D stacked SoC," *Journal of Systems Engineering and Electronics*, vol. 26, no. 2, pp. 317–328, 2015.

[31] Z. Yue, S. Zhang, and W. Xiao, "A novel hybrid algorithm based on grey wolf optimizer and fireworks algorithm," *Sensors*, vol. 20, no. 7, article 2147, 2020.

[32] J. Cheng, L. Wang, and Y. Xiong, "Ensemble of cuckoo search variants," *Computers & Industrial Engineering*, vol. 135, pp. 299–313, 2019.

[33] W. Long, S. Cai, J. Jiao, M. Xu, and T. Wu, "A new hybrid algorithm based on grey wolf optimizer and cuckoo search for parameter extraction of solar photovoltaic models," *Energy Conversion and Management*, vol. 203, article 112243, 2020.

[34] Z. Zhang, S. Ding, and W. Jia, "A hybrid optimization algorithm based on cuckoo search and differential evolution for solving constrained engineering problems," *Engineering Applications of Artificial Intelligence*, vol. 85, pp. 254–268, 2019.

[35] X. Zhang, X. . Li, and M. . Yin, "Hybrid cuckoo search algorithm with covariance matrix adaption evolution strategy for global optimisation problem," *International Journal of Bio-Inspired Computation*, vol. 13, no. 2, pp. 102–110, 2019.

[36] G. H. Davies, *The life of birds, parenthood*, 1970.

[37] K. Khan and A. Sahai, "Neural-based cuckoo search of employee health and safety (hs)," *International Journal of Intelligent Systems and Applications*, vol. 5, no. 2, pp. 76–83, 2013.

[38] X.-S. Yang and S. Deb, "Cuckoo search via lévy flights," in *2009 World Congress on Nature & Biologically Inspired Computing (NaBIC)*, pp. 210–214, Coimbatore, India, 2009.

[39] X.-S. Yang and L. Press, *Nature-Inspired Metaheuristic Algorithms Second Edition*, Luniver Press, 2010.

[40] C. T. Brown, L. S. Liebovitch, and R. Glendon, "Lévy flights in dobe ju/'hoansi foraging patterns," *Human Ecology*, vol. 35, no. 1, pp. 129–138, 2007.

[41] I. Pavlyukevich, "Levy flights, non-local search and simulated annealing," *Journal of Computational Physics*, vol. 226, no. 2, pp. 1830–1844, 2007.

[42] I. Pavlyukevich, "Cooling down lévy flights," *Journal of Physics A: Mathematical and Theoretical*, vol. 40, no. 41, pp. 12299–12313, 2007.

[43] 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, p. e354, 2007.

[44] M. F. Shlesinger, G. M. Zaslavsky, and U. Frisch, *Lévy flights and related topics in physics*, Springer Berlin Heidelberg, 1995.

[45] M. F. Shlesinger, "Search research," *Nature*, vol. 443, no. 7109, pp. 281-282, 2006.

[46] X.-S. Yang and S. Deb, "Engineering optimisation by cuckoo search," 2010, https://arxiv.org/abs/1005.2908.

[47] L. Wang, Y. Yin, and Y. Zhong, "Cuckoo search with varied scaling factor," *Frontiers of Computer Science*, vol. 9, no. 4, pp. 623–635, 2015.

[48] M. Reda, A. Y. Haikal, M. A. Elhosseini, and M. Badawy, "An innovative damped cuckoo search algorithm with a comparative study against other adaptive variants," *IEEE Access*, vol. 7, pp. 119272–119293, 2019.

[49] X.-S. Yang and S. Deb, "Cuckoo search: recent advances and applications," *Neural Computing and Applications*, vol. 24, no. 1, pp. 169–174, 2014.

[50] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014.

[51] Q. Al-Tashi, S. J. A. Kadir, H. M. Rais, S. Mirjalili, and H. Alhussian, "Binary optimization using hybrid grey wolf optimization for feature selection," *IEEE Access*, vol. 7, pp. 39496–39508, 2019.

[52] M. M. Mafarja and S. Mirjalili, "Hybrid whale optimization algorithm with simulated annealing for feature selection," *Neurocomputing*, vol. 260, pp. 302–312, 2017.

[53] M. Črepinšek, S.-H. Liu, and M. Mernik, "Exploration and exploitation in evolutionary algorithms: a survey," *ACM Computing Surveys (CSUR)*, vol. 45, no. 3, p. 35, 2013.

[54] J. Wei and Y. Yu, "An effective hybrid cuckoo search algorithm for unknown parameters and time delays estimation of chaotic systems," *IEEE Access*, vol. 6, pp. 6560–6571, 2018.

[55] J. Huang, X. Li, and L. Gao, "A new hybrid algorithm for unconstrained optimisation problems," *International Journal of Computer Applications in Technology*, vol. 46, no. 3, pp. 187–194, 2013.

[56] R. Chi, Y. Su, Z. Qu, and X. Chi, "A hybridization of cuckoo search and differential evolution for the logistics distribution center location problem," *Mathematical Problems in Engineering*, vol. 2019, Article ID 7051248, 16 pages, 2019.

[57] W. Long, J. Jiao, X. Liang, and M. Tang, "An exploration-enhanced grey wolf optimizer to solve high-dimensional numerical optimization," *Engineering Applications of Artificial Intelligence*, vol. 68, pp. 63–80, 2018.

[58] C. K. Alexander and M. N. O. Sadiku, *Fundamentals of Electric Circuits*, McGraw-Hill Education, New York, England, 6th edition, 2017.

[59] M. K. Naik and R. Panda, "A novel adaptive cuckoo search algorithm for intrinsic discriminant analysis based face recognition," *Applied Soft Computing*, vol. 38, pp. 661–675, 2016.

[60] N. Mittal, U. Singh, and B. S. Sohi, "Modified grey wolf optimizer for global engineering optimization," *Applied Computational Intelligence and Soft Computing*, vol. 2016, Article ID 7950348, 16 pages, 2016.

[61] S. Mirjalili and S. Z. M. Hashim, "BMOA: binary magnetic optimization algorithm," *International Journal of Machine Learning and Computing*, vol. 2, no. 3, pp. 204–208, 2012.

[62] M. Lichman, "UCI machine learning repository," 2013, http://archive.ics.uci.edu/ml.

[63] D. Segera, M. Mbuthia, and A. Nyete, "Particle swarm optimized hybrid kernel-based multiclass support vector machine for microarray cancer data analysis," *BioMed Research International*, vol. 2019, Article ID 4085725, 11 pages, 2019.