1-1-2022

Improved Reptile Search Optimization Algorithm using Chaotic map and Simulated Annealing for Feature Selection in Medical Filed

Zenab Elgamal
University of Malaya

Aznul Qalid Md Sabri
University of Malaya

Mohammad Tubishat
Zayed University, mohammad.tubishat@zu.ac.ae

Dina Tbaishat
Zayed University, dina.tbaishat@zu.ac.ae

Sharif Naser Makhadmeh
Ajman University of Science and Technology

See next page for additional authors

Author First name, Last name, Institution
Zenab Elgamal, Aznul Qalid Md Sabri, Mohammad Tubishat, Dina Tbaishat, Sharif Naser Makhadmeh, and Osama Ahmad Alomari

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2022.3174854, IEEE Access

IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

Improved Reptile Search Optimization Algorithm using Chaotic map and Simulated Annealing for Feature Selection in Medical Filed

Zenab Elgamal [1], Aznul Qalid Md Sabri [1], Mohammad Tubishat [2], Dina Tbaishat[2], Sharif Naser Makhadmeh[3], Osama Ahmad Alomari[4]

[1] Faculty of Computer Science and Information Technology, University of Malaya 50603, Kuala Lumpur, Malaysia
[2] College of Technological Innovation, Zayed University, Abu Dhabi, United Arab Emirates
[3]Artificial Intelligence Research Center (AIRC), College of Engineering and Information Technology, Ajman University, Ajman, United Arab Emirates
[4]MLALP Research Group, University of Sharjah, Sharjah P.O. Box 27272, United Arab Emirates
[*] Corresponding author: Aznul Qalid Md Sabri 1 , Zenab Elgamal 1 aznulqalid@um.edu.my, zenabelgamal@siswa.um.edu.my

ABSTRACT The increased volume of medical datasets has produced high dimensional features, negatively affecting machine learning (ML) classifiers. In ML, the feature selection process is fundamental for selecting the most relevant features and reducing redundant and irrelevant ones. The optimization algorithms demonstrate its capability to solve feature selection problems. Reptile Search Algorithm (RSA) is a new nature-inspired optimization algorithm that stimulates Crocodiles' encircling and hunting behavior. The unique search of the RSA algorithm obtains promising results compared to other optimization algorithms. However, when applied to high-dimensional feature selection problems, RSA suffers from population diversity and local optima limitations. An improved metaheuristic optimizer, namely the Improved Reptile Search Algorithm (IRSA), is proposed to overcome these limitations and adapt the RSA to solve the feature selection problem. Two main improvements adding value to the standard RSA; the first improvement is to apply the chaos theory at the initialization phase of RSA to enhance its exploration capabilities in the search space. The second improvement is to combine the Simulated Annealing (SA) algorithm with the exploitation search to avoid the local optima problem. The IRSA performance was evaluated over 20 medical benchmark datasets from the UCI machine learning repository. Also, IRSA is compared with the standard RSA and state-of-the-art optimization algorithms, including Particle Swarm Optimization (PSO), Genetic Algorithm (GA), Grasshopper Optimization algorithm (GOA) and Slime Mould Optimization (SMO). The evaluation metrics include the number of selected features, classification accuracy, fitness value, Wilcoxon statistical test (*p*-value), and convergence curve. Based on the results obtained, IRSA confirmed its superiority over the original RSA algorithm and other optimized algorithms on the majority of the medical datasets.

INDEX TERMS Reptile Search Algorithm (RSA); Feature Selection (FS); Optimization Algorithm; Chaos Theory; Simulated Annealing (SA).

I. INTRODUCTION

Disease detection and diagnosis critically depend on the classification of biomedical datasets. Classifying such datasets can detect complex diseases such as COVID-19, Tumors, etc. The early detection of such diseases increases the survival rate [1]. In biomedical sciences, the disease categorized are classified based on various features [2], [3], [4]. The biomedical datasets are rapidly growing, resulting in high dimensional features [5]. In some cases, these features are redundant, inefficient, or embedding the same classification effect as others [6]. A robust ML classifier is required to reduce the complexity and the time taken to classify these features [7]. The ML classifier is suffer from the redundant, inefficient and biased features[8]. Thus, FS is an important component of the ML processes [9].

Feature selection (FS) has an important role in ML as a pre-processing phase, pruning the redundant and irrelevant features and selecting the most relevant ones. This process can be accomplished by excluding the features that may negatively impact classifier performance, such as unrelated, redundant, and less-informative features [10]. FS has been applied widely in many applications, image segmentation [11], image processing [12], medical diagnosis [13], cancer detection [14], text recognition [15] and more. Based on the literature, the FS technique has four basic steps, including (1) creating the feature subset, (2) the evaluating feature subset, (3) defining the stop condition, and (4) validating the selected subset [16]. According to the evaluation criteria, FS techniques are divided into two main Approaches: Filter Based Approach (FBA) and Wrapper Based Approach (WBA).

The FBA is an approach to filter the feature subsets based on static evaluation tests. The filtration processes of the subset features are independent of the ML classifier [17]. [18]. The Pearson's Correlation, Chi-squared test, and Linear Discriminant Analysis (LDA) are examples of FBA approaches, where filtering is performed before the application ML classifier with no direct contact with the classifier [19]. Unlike, the Wrapper-Based Approach (WBA) which is connected directly to the classifier [20]. The WBA is an approach that evaluates the subsets of features to find the possible correlation between the features based on the applied ML classifier [5]. A WBA is computationally expensive, but it has better results when compared to FBA [21], [22].

Commonly, WBA is used for FS problems because it considers the classification performance, and the feature reduction conditions, in addition to its ability to interact directly with the classifier. Furthermore, WBT minimizes the search area; as a result, the classification performance improves, and the selected features decline, as illustrated in [23]. In WBA, the fitness function is applied to evaluate the FS process depending on the classification accuracy [24]. Based on the literature, the WBA is commonly categorized into three main groups: Forward Feature Selection (FFS), Backward Feature Elimination (BFE), and Recursive Feature Elimination (RFE) [25] . The FFS is an iterative process in which the model starts with no features, then in each iteration, new features are added until the performance no longer improves the model. BFE is a backward elimination that starts with all features and eliminates the lowest significant feature in each iteration; as a result, the model performance improves. Finally, the RFE is a greedy optimization algorithm that repetitively builds models and keeps aside the best or the worst performing feature at each iteration. It then creates the new model with the remaining features until all the features are consumed. After that, features are classified based on the order of their elimination. Several researchers have been using WBA methods in optimization algorithms to solve the problem of feature selection [24], [5], [9]. However, the typical inclusive search aimed to find all possible combinations of features from the total set of features, is considered time-consuming search and is referred to as Nondeterministic Polynomial problem, known as an NP-hard problem [26]. The above reasons along with the powerful WBA characteristics urged this study to utilize WBAs for feature selection problems.

Based on the literature, optimization algorithms have been used to solve FS problem based on WBA, such as the Chimp Optimization Algorithm (COA) was improved in wrapper-mode for feature selection [5], the Dragonfly Algorithm (DA) with Evolutionary Population Dynamics and Adaptive crossover was developed in wrapper-mode for Feature Selection [27], the butterfly optimization algorithm (BOA) was developed in wrapper mode for feature selection [28], the particle swarm optimization was improved in wrapper mode for feature selection [29], and the Whale Optimization Algorithm (WOA) was combine with simulated annealing in wrapper mode for feature selection [30]. The main purpose of using optimization algorithms in FS is to find the optimal features combination or those close to the optimal features within a reasonable time. The wrapper mode helps to evaluate the classification accuracy based on the classifier [20], in this work KNN classifier is used.

However, optimization algorithms suffer from local optima and population diversity problems when dealing with high-dimensional problems, such as the FS problem [10], [30], [31], [32]. Additionally, according to "No-Free Lunch" (NFL) theorems, some algorithms achieve high performance in a particular problem and display low performance in another [33], [34], [35]. Therefore, designing new optimization algorithms and developing existing ones is one of the great interests of researchers in this field of study. Reptile Search Algorithm (RSA) is one of the newest optimization algorithms [36]. RSA is a wildlife-inspired metaheuristic algorithm that mimics Crocodiles' encircling and hunting behavior. RSA's unique search strategies demonstrated superior results over other optimization algorithms. However, RSA is limited by the problem of population diversity and local optima when applied to high-dimensional feature selection. The reasons cited above, and RSA characteristics motivated the researchers of this study to improve RSA in wrapper mode for feature selection problems.

This research proposes a novel algorithm named Improved Reptile Search Algorithm (IRSA). The goal of IRSA is to improve classification performance for feature selection problems in medical datasets and solve the limitation of the standard RSA algorithm. To solve the weaknesses of the standard RSA algorithm and adapt it to FS problem, the following improvements are introduced to RSA algorithm. In the initialization phase of IRSA, the chaotic map algorithm is used to initialize the solutions (search agents). IRSA is expected to achieve a faster convergence rate and generate a wider range of solutions due to the proposed version. Furthermore, to avoid local optima and improve RSA exploitation ability, IRSA combined the SA algorithm with the local search capabilities of the RSA. A number of hybrid optimization algorithms have been presented in the literature to solve feature selection problems. However, to the best of the authors' knowledge, there is no previously published work on improving RSA with a chaotic map and the SA algorithm for feature selection problems. The contributions of this work are summarized as follows:

1) IRSA: a modified variant of the RSA algorithm intended to solve its weaknesses and provide better performance in feature selection.
2) The standard RSA has been improved in two main ways, including:

- The chaotic maps are used in the initialization phase of RSA to improve its solutions diversity.
- Improve the exploitation and avoid local optima, simulated annealing (SA) is combined with RSA.

3) The IRSA algorithm is developed in wrapper mode for feature selection problem.
4) To evaluate the performance of the IRSA algorithm, the experiments are conducted on 20 UCI medical datasets with various dimensionalities. In addition, IRSA results are compared with original RSA and four well-known optimization algorithms including: Particle Swarm Optimization (PSO), Genetic Algorithm (GA), Grasshopper Optimization Algorithm (GOA) and Slime Mould Optimization (SMO). The number of features, classification accuracy, fitness values, P-value, and convergence rate are used as evaluation metrics.

The rest of the article is organized as follows: Section 2 presents a review of related works. Section 3 provides a brief description of the RSA, Chaotic Maps (CM), and Simulated Annealing (SA), The proposed algorithm IRSA is illustrated in Section 4. Section 5 describes the datasets used and experimental details, and Section 6 illustrates the experimental results and discussion. Finally, Section 7 concludes the article.

II. RELATED WORK

A meta-heuristic algorithm is a higher-level sequence of programmable instructions that performs a specific task and provides a sufficiently good solution to an optimization problem within a reasonable time [37]. The meta-heuristic optimization algorithms contain two main phases: (1) exploration (global search) and (2) exploitation (local search). Exploration is the ability to search for solutions in the search space globally. Its ability is associated with escaping and preventing being trapped in local optima. The exploitation is the ability to search locally for a more optimal solution. Good performance is obtained by achieving an optimal balance between these two phases. All population-based algorithms use these features but with different operators and structures [38]. Meta-heuristics are categorized into three main classes: swarm intelligence optimization algorithm, evolutionary optimization algorithm, and physics-based optimization algorithm. The RSA is a new swarm intelligence optimization algorithm. The Swarm Intelligence Optimization algorithm (SIO) is a meta-heuristic algorithm that mimics animals' social behavior in groups (e.g., Crocodiles, Whales, Wolves, etc.). The main feature of SIO is the ability to share the information from multiple sources during the optimization process [39]. The most popular algorithm that belong to this class is the PSO algorithm which was developed by Kennedy & Eberhart in 1995 [40]. PSO simulates the behavior of birds flying together in flocks. Other examples of this type include Whales Optimization Algorithm (WOA) [41], Grey Wolf Optimizer [42], Harris Hawks Optimization (HHO) [43], Salp Swarm Algorithm [44] and others.

Recently, Optimization Algorithm (OA) has been applied in various applications to solve high-dimensional feature selection problems. OA achieved significant improvement in classification accuracy and reduced the number of selected features in various applications. Examples of these recent applications are WOA developed in wrapper mode for feature selection problem [45], Also WOA improved for feature selection in Arabic sentiment analysis [15], Butterfly Optimization Approaches (BOA) developed in binary mode for feature selection. [46], Salp Swarm Algorithm (SSA) is developed based on opposition and new local search mechanism for feature selection [23], Antlion optimization (ALO) similarly developed in wrapper mode for feature selection [47], moreover, PSO is hybrid with spiral shaped algorithm for feature selection [48], GOA was improved using opposition-based learning for feature selection [49], Equilibrium Optimization Algorithm (EOA) was improved using Elite Opposition-Based Learning method and new local search strategy for feature selection [20] and many more. Although each optimization algorithm embraces its unique structure, there are some common characteristics: the search agent initialize a random population (solutions) as the primary process and set the best solution so far, then on each iteration the new solutions are evaluated based on the defined fitness function, after that, the best solution is chosen based on a termination criterion [50]. All optimization algorithms perform exploration and exploitation phases. The imbalanced trade-off between exploration and exploitation slows the convergence speed towards the optimal solution [51]. The original RSA may still not achieve an optimal balance between local and global search, especially when applied for feature selection in high dimensional datasets. The algorithm's imbalanced behavior causes slow convergence and quickly falls into local optima problems. Thus, two main improvements need to be applied in RSA. The first improvement is to enhance the population diversity of the algorithm by applying a Chaotic map to the initial solution. The second improvement is improving the local search by combining SA with the local search strategy in RSA.

The Chaotic Map (CM) is a dynamic system [52]. This system is one of the modern methods used in the literature to solve the population diversity problem and low convergence speed in the optimization algorithm. It is a useful method for searching for global optimum solutions in a search space [53]. Chaos Optimization Algorithm (COA) uses the benefit of the chaotic structures in several applications as reported [54]. It had been proven that changing the random parameter values with a chaotic system can enhance classification [55]. Therefore, several efforts contributing to optimization algorithms have involved chaos theory to improve performance and adjust specific parameters. Examples of these implementations are the Harris Hawks Optimization (HHO) [56], where the chaotic map was applied to improve the initial solution of HHO. Also, Chaotic Crow Search Optimization (CCSA) [53], where a chaotic maps was also applied to improve the convergence speed and prevent the local optima problem. Additionally, Chaotic Grasshopper Optimization Algorithm (CGOA) to accelerate the global convergence speed of GOA algorithm [57]. As well as the Chaotic Whale Optimization Algorithm (CWOA) using the chaos maps to improve the global convergence rate and enhance the algorithm performance of WOA algorithm [58]. Similarly, Chaotic Salp Swarm Algorithm (CSSA) algorithm examined a chaotic map to improve the local optima problem and low convergence. Chaotic Gray Wolf Optimization (CGWO) where the chaotic system was applied to accelerate the global convergence rate [55]. These algorithms have all embedded chaos maps to improve the global optimization, used in different fields and applications. The reported results verified noticeable improvements after integrating the chaos maps to these algorithms.

All of these have encouraged our research to explore the effect of combining chaos maps with RSA to improve population diversity. In this work, Circle chaotic map value replaced the randomly generated values for initializing the Reptile positions at the initialization phase. It is worth mentioning that different types of chaotic maps were applied to the optimization algorithm [56]. Examples of these maps are Singer, Sinusoidal, Chebyshev, Circle, Tent, Sine, Piecewise, Logistic, Iterative, and Gauss/mouse. These maps, with their statistical equations, are used in several applications. These maps significantly increase the convergence rate and the fitness performance of the algorithms, as reported in several studies [59], [60], [61], [62]. However, the circle map outperforms other chaotic maps in several studies [63], [64]. In addition, the Circle map provided the high stability with high classification performance and a small number of features [59],[65], [66], [67] . Therefore, we utilized Circle chaotic map to improve the diversity of solution at the initialization phase of RSA.

On the contrary, the next phase intends to enhance the search process for local regions rather than all feature spaces. Usually, exploitation is performed after the exploration phase [68]. In most complex applications, optimization algorithms are trapped in local optima due to the incorrect balance between the exploitation and exploration and the randomization nature of the initialization process. Based on the literature, it has been found that many optimization algorithms use the Simulated Annealing (SA) algorithm to enhance the local search strategy. In our work, SA is proposed to solve the RSA local optima problem, specifically for high dimensional FS. SA was presented in 1983 by Kirkpatrick [69]. It is considered a hill-climbing method that enhances the candidate solution for the objective function. SA algorithm was used to improve the exploitative capability of the algorithm and prevent local optima problems. Many optimization algorithms used SA to enhance the local search strategy. Examples of these implementations such as: the hybridization of PSO with SA for feature selection [70]. The hybridization of SA algorithm with Moth-Flame Optimization to increase the advantage to improve its exploitation capability [71]. Another example is the hybridization of Whale Optimization Algorithm with SA to improve the WOA exploitation for feature selection [72]. Also, the hybridization of the Salp Swarm Algorithm (SSA) with SA Algorithm to adjust the balance between exploration and exploitation of SSA algorithm [73]. Finally, Monarch Butterfly Optimization (MBO) with SA strategy to improve the convergence speed of MBO algorithm. The unique structure and performance obtained by employing the SA in these previous studies inspired this research to include the SA algorithm in the iteration process to enhance the RSA local search. SA is proposed to solve the RSA local optima problem.

Reptile Search Algorithm (RSA) is a new natural-inspired meta-heuristic optimizer [36]. This algorithm is inspired by Crocodiles' encircling and hunting behaviours in the wild. The key difference between the RSA algorithm and other optimization algorithms is that RSA has a unique method to update the search-agent locations using four new methods. For instance, the act of surrounding is conducted by high-walking or belly-walking, and the Crocodiles communicate or collaborate to perform hunting. RSA attempt to generate powerful search methods that can produce better quality results and get new solutions that can help solve complex real-life issues. However, as reported by the author, RSA successfully solves Artificial Landscapes Functions (ALF) and real-world engineering problems compared to other popular optimization algorithms. The ALF are benchmark mathematical functions used to evaluate the performance of optimization algorithms. Furthermore, although RSA is considered to be a random population optimization algorithm, it is prone to issues such as population diversity and local optima when dealing with high-dimensional features. These reasons and the RSA characteristics motivated this study to improve the performance of the RSA to adapt for the feature selection problem. The following section provides an overview and background about the RSA algorithm.

III. BASICS AND BACKGROUND

A. REPTILE SEARCH ALGORITHM (RSA)

RSA is a novel optimization algorithm developed by Abualigah et al. in 2022 [36], which mimics the Crocodiles encircling and hunting behaviour. The Crocodiles are semiaquatic reptiles with unique physical characteristics such as lined body shape, the ability to raise their legs to the side when they walk, the belly walk, and the swim. These characteristics allow them to become powerful hunters in the wild. This section describes the exploration and exploitation capabilities of the RSA, which is based on the smart encircling and hunting of the prey. Furthermore, the mathematical functions and Pseudo-code of the algorithm are covered. The RSA is a population-based and gradient-free method that can solve complex and simple optimization problems subject to specific constraints.

1) INITIALIZATION PHASE

In this phase, the initial candidate solutions are generated based on chaotic maps as in Eq. (1). Also, the search-space and the objective function are defined. As well, all parameter values are set before computation.

$$X=\begin{bmatrix} x_{1,1} & \cdots & x_{1,j} & x_{1,n-1} & x_{1,n} \\ x_{2,1} & \cdots & x_{2,j} & x_{2,n-1} & x_{2,n} \\ \cdots & \cdots & x_{i,j} & x_{i,n-1} & x_{i,n} \\ \vdots & & \vdots & \vdots & \vdots \\ x_{N-1,1} & \cdots & x_{N-1,j} & \cdots & x_{N-1,n} \\ x_{N,1} & \cdots & x_{N,j} & x_{N-1,n} & x_{N,n} \end{bmatrix} \quad (1)$$

where X is a represent the candidate solutions produced by using Eq. (2), and $x_{i,j}$ indicate the jth search-agent position of the ith solution, and N is the number of potential solutions, n indicates the size of the problem.

$$x_{ij}=rand\ (UB-LB)+LB, j=1,2,\dots,n \quad (2)$$

where the $rand$ is an initiation value. Also, the UB and LB are defined, which specify the upper and lower bounds of the given problem, respectively.

2) EXPLORATION PHASE (ENCIRCLING)

In this phase, the exploratory behaviour (encircling) of RSA is discussed. Two strategies Crocodiles perform during their encircling process: high walking and belly walking. These movements refer to different approaches, which are committed to representing the algorithm's exploration capabilities (global search). Crocodile movements (high walk and belly walk) prevent them from catching the prey due to their noise unless they employ another search mechanism (exploration phase). Hence, the exploration search discovers a wide search space; it can find the promising area maybe after several searches.

The RSA balanced exploration (encircling) and exploitation (hunting) search according to four conditions; break the total number of iterations into four parts. Exploration mechanisms in RSA concentrate on two major search strategies (high walking and belly walking) to explore the search space and find a better solution. The high walk strategy is defined by $t\le\frac{T}{4}$, and the belly walk motion strategy is defined by $t\le 2\frac{T}{4}$ and $t\le\frac{T}{4}$. This means the condition will be met for almost the half number of exploration iterations (High walk) and another half for the (Belly walk). The position updating formula is presented for the exploration phase as shown in Eq. (3).

$$x_{(i,j)}(t+1) = \begin{cases} Best_j(t)\times -\eta_{(i,j)}(t)\times\ \beta-\ \boldsymbol{R}_{(i,j)}(t)\times rand, & t\le\frac{T}{4} \\ Best_j(t)\times\ x_{r1,j}\ \times \boldsymbol{ES}(t)\times rand, & t\le 2\frac{T}{4}\ and\ t\le\frac{T}{4} \end{cases} \quad (3)$$

where $Best_j(t)$ presents the jth position in the best-achieved solution so far, $rand$ refers to an integer between 0 and 1, t is the current iteration number, and T stands for the maximum number of iterations. $\eta_{(i,j)}$ identifies the exploration operator of the jth position in the ith solution, calculated by Eq. (4). β is a critical parameter, that guides the exploration accuracy for the encircling time through iterations, which is set to 0.1 value. $\boldsymbol{R}_{(i,j)}$ is an amount applied to reduce the search area, calculated by Eq. (5). $r1$ is a random number between $[1,N]$, and $x_{r1,j}$ refer to a random position of the ith solution. Evolutionary Sense $\boldsymbol{ES}(t)$ is a random ratio between $[2,-2]$ describe the probability of decreasing values throughout the iterations, calculated by Eq. (6).

$$\eta_{(i,j)}=Best_j(t)\times\ P_{(i,j)} \quad (4)$$

$$\boldsymbol{R}_{(i,j)}=\frac{Best_j\,(t)-x_{(r_2,j)}}{Best_j(t)+\epsilon} \quad (5)$$

$$\boldsymbol{ES}(t)=2\times r_3\times\left(1-\frac{1}{T}\right), \quad (6)$$

where r_2 is a random number between $[1,N]$ and ϵ a small amount. In Eq. (6), 2 is the correlation value used to give values between 2 and 0, r_3 which implies to a random integer number between $[1,-1]$. $P_{(i,j)}$ corresponding to the difference between the jth position of the best-obtained solution and the jth position of the current solution, calculated by Eq. (7).

$$P_{(i,j)}=\alpha\ +\frac{x_{(i,j)}-M(x_i)}{Best_j(t)\times\left(UB_{(j)}-LB_{(j)}\right)+\ \epsilon} \quad (7)$$

where $M(x_i)$ stands to the average positions of the ith solution, calculated by Eq. (8). $UB_{(j)}$ and $LB_{(j)}$ are the boundaries of the jth position, respectively. α is a critical parameter, guides also the exploration accuracy for the hunting cooperation over the course of iterations, which set to 0.1 value in this work.

$$M(x_i)=\frac{1}{n}\sum_{j=1}^{n}x_{(i,j)} \quad (8)$$

3) EXPLOITATION PHASE (HUNTING)

In this phase, the exploitative behaviour (hunting) of RSA is introduced. Two strategies Crocodiles perform during their hunting process: cooperation and coordination. These strategies simulate the exploitation search (Local search), formulated as in Eq. (9). The strategy for hunting coordination in this phase is conditioned by $t\le 3\frac{T}{4}$ and $t2\frac{T}{4}$, or else the hunting cooperation strategy is executed. In the original RSA the position updating formula for the exploitation are presented in Eq. (9)

$$x_{(i,j)}(t+1) = \begin{cases} Best_j(t) \times \boldsymbol{P}_{(i,j)}(t) \times rand, & t \leq 3\frac{T}{4} \text{ and } t > 2\frac{T}{4} \\ Best_j(t) - \eta_{(i,j)}(t) \times \epsilon - \boldsymbol{R}_{(i,j)}(t) \times rand, & t \leq T \text{ and } t > 3\frac{T}{4} \end{cases} \quad (9)$$

where $Best_j(t)$ is the jth position in the best-found solution so far, $\eta_{(i,j)}$ implies to the hunting parameter for the jth position in the ith solution, calculated by Eq. (3). $\boldsymbol{P}_{(i,j)}$ is the difference between the jth position of the best-found solution and the jth position of the current solution, calculated by Eq. (6). $\eta_{(i,j)}$ implies to the hunting parameter for the jth position in the ith solution, which is calculated using Eq. (3). $\boldsymbol{R}_{(i,j)}(t)$ is an amount applied to reduce the search area in the current iteration, calculated by Eq. (4).

B. CIRCLE CHAOTIC MAP

Chaos theory is commonly used in optimization algorithms to optimize the diversity of initialized solutions. The improvement of initialized solutions using chaotic map increases the performance of algorithms. Moreover, chaos theory can explore the search space more thoroughly than random search [74]. However, in order to make the initial population as effective as possible, it is important to leverage solution space as much as possible. This work applies Chaos theory's Circle Map (CM) to initialize the IRSA to improve population diversity. The Circle map is a one-dimensional function extracted from the circle itself. Mathematically, it is equivalent to a point in the circle line, assumed as starting point x that calculated modulo 2π, to identify, the angle of the point in the circle [75]. The modulo of two numbers are given, a similar remainder when divided by same number. When the modulo is taken with a value other than 2π the result still represents an angle but must be normalized so that the whole range between $[0,2\pi]$ as proofed by [75]. In this implementation, the CM control variables are set to $a = 0.5$ and $b = 0.2$. The mathematical model of the CM is computed as in Eq. (10).

$$ChaosCircleMap = x_{n+1} = x_{ij} + b - \left(\frac{a}{2\pi}\right)\sin\left(2\pi x_{ij}\right) mod(1), \quad (0,1) \quad (10)$$

where n refers to the symbol of chaotic sequence $x,$ and x_n is the nth chaotic number of chaotic sequence. As defined earlier, the b and a are controlling variables that help identify the chaotic performance. The CM value replaced the Crocodiles random initial position's (search-agent) values in the IRSA.

C. SIMULATED ANNEALING

The Simulated Annealing (SA) algorithm was used by several optimization algorithms to improve exploitative capability and to prevent local search problems, As illustrated in the literature review. In this work, to avoid the local optima stagnation problem of original RSA, the SA is applied at the end of each RSA iteration to improve the best solution. Where the best solution will be accepted, and the worst solution will be taken with a well-defined probability to avoid local optima. The Boltzmann probability function determines the likelihood of choosing a worse solution as in Eq. (12).

$$P = e - T(Generated_{Sol} - Best_{Sol}) \quad (12)$$

were e is the energy of the system, T is a parameter (named temperature) that periodically decreases throughout the search process the decreasing rate is $\alpha = 0.99$, thus in next iteration $T = T - \alpha$. The ratio of probabilities of two states is known as the Boltzmann factor, which is computed by the fitness function between the best solution ($Best_{Sol}$) and the generated solution ($Generated_{Sol}$). In this experiment, all SA parameters are based on the cooling schedule [76] and adopted it as in Yarpiz.com [77].

IV. THE PROPOSED IMPROVED REPTILE SEARCH ALGORITHM (IRSA)

In this study, a novel IRSA for feature selection is proposed. The proposed IRSA is a hybrid of the original RSA with chaos theory and SA algorithm. The aim of this improvement is to increase the classification accuracy and decrease the number of selected features. However, the original RSA has two noteworthy drawbacks when used to solve high-dimensional problem, such as feature selection. These drawbacks include the diversity of initial solutions and local optima problems. Therefore, two modifications are suggested to the RSA to overcome the feature selection problem. The first improvement includes integrating the chaotic maps, specifically, Circle Map (CM) at the initialization phase to improve RSA solutions diversity. The second improvement is combining the SA algorithm to the exploitation phase of the RSA to improve the local search. The details of these improvements are presented in this section as follows.

In the IRSA algorithm, the CM value will replace the stochastic values of initializing the RSA population positions at the initialization phase. The chaotic values are generated from Circle chaotic map. This map notably increases the convergence speed and the fitness performance of the RSA, as will be presented later in the experimental result and discussion section.

Furthermore, the second improvement is to combine the SA in the IRSA to enhance its exploitation cababilites. After implementing CM and find best solution, SA is used to improve the current best solution at the end of each RSA iteration. The pseudocode of the proposed CHHO algorithm is illustrated in Algorithm 1.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2022.3174854, IEEE Access

Algorithm 1: Pseudo-code of IRSA algorithm

1: **Initialization phase**:
2: Initialize RSA parameters α, β, etc.
3: **Initialize the solutions' positions based Chaotic Circle map.** X (i = 1,2, …, N)
4: **while** (fitness value != stopping criteria) **do**
5: Calculate the Fitness value for the candidate solutions (X).
6: Find the Best solution so far.
7: Update the $\boldsymbol{ES}$ using Equations (6).
8: **The starting of the RSA**
9: **for** ($i = 1$ to N) **do**
10: **for** ($j = 1$ to N) **do**
11: Update the η, R, P values using Equations (4), (5)and (7), respectively.
12: **if** ($t \leq \frac{T}{4}$) **then**
13: $x_{(i,j)}(t+1) = Best_j(t) \times -\eta_{(i,j)}(t) \times \beta - \boldsymbol{R}_{(i,j)}(t) \times rand,$ ⊳ {High walking}
14: **else if** ($t \leq 2\frac{T}{4}$ and $t \leq \frac{T}{4}$) **then**
15: $x_{(i,j)}(t+1) = Best_j(t) \times x_{r1,j} \times \boldsymbol{ES}(t) \times rand,$ ⊳ {Belly walking}
16: **else if** ($t \leq 3\frac{T}{4}$ and $t > 2\frac{T}{4}$)**then**
17: $x_{(i,j)}(t+1) = Best_j(t) \times \boldsymbol{P}_{(i,j)}(t) \times rand,$ ⊳ {Hunting coordination}
18: **else**
19: $x_{(i,j)}(t+1) = Best_j(t) - \eta_{(i,j)}(t) \times \epsilon - \boldsymbol{R}_{(i,j)}(t) \times rand,$ ⊳ {Hunting cooperation}
20: **end if**
21: **end for**
22: **end for**
23: **Apply SA**
24: t = t + 1
25: **end while**
26: Return the best solution $Best(X)$.

A. FITNESS FUNCTION

In this work, the proposed fitness function is used to calculate the classification accuracy of each solution as well as the number of selected features. Each solution is computed according to a proposed fitness function that depends on a K-Nearest Neighbor (KNN) classifier in wrapper mode (Altman, 1992). However, after the candidate solution is initialized, the fitness value is calculated to be saved as the best solution so far. Then, in each iteration, a fitness function is computed following the exploration and exploitation of the current best position. It is assumed that the fitness value of the new position (solution) is better than the current position. As a result, the best solution is replaced by the improved solution, and a neighbourhood search is performed. This process is repeated until stopping criteria is performed. The proposed fitness function is utilized as in Eq. (13)

$$Fitness = \alpha\gamma_R(D) + \beta\frac{R}{N} \qquad (13)$$

where $\alpha\gamma_R(D)$ refer to the classification error rate of the used classier KNN. Furthermore, R is a number of the selected subset, and N is the total number of features in the dataset, α, and β are two parameters corresponding to the importance of classification quality and subset length, $\alpha \in [0,1]$ and $\beta = (1-\alpha)$ approved in [78] and [72]. The Pseudo-code of the proposed IRSA algorithm is explained in Algorithm 1. Additionally, the flowchart of the proposed IRSA is presented in Figure 1.

B. COMPUTATIONAL COMPLEXITY

IRSA complexity is determined by three main parameters: initialization, fitness evaluation, and updating of the candidate solutions processes. First, the computational complexity of the initialization process is $O(N)$, for all possible solutions N. Second, the computational complexity of the updating processes $O(T*N) + O(T*N*Dim)$, which is contained in the searching of the best location and updating the location vector of all solutions, where T indicates the maximum number of iterations and Dim is the dimension of the search space. However, the computational complexity of applying SA local search is defined as $O(T*I*S)$, where I is the number of SA iteration, and S is the search strategy. Accordingly, the computational complexity of the proposed IRSA is formulated as in Eq. (14)

$$O(IRSA) = O\big(N \times (T \times Dim + 1) + (TIS)\big) \qquad (14)$$

where, T is the number of iterations, N presents the number of solutions, and Dim refers to the solution size.

V. EXPERIMENTAL RESULTS AND DISCUSSION

The experimental details will be discussed in this section. In addition, this section presents the evaluation performance and validation criteria of the proposed IRSA. In this context, the IRSA algorithm was compared with some well-known and new optimization algorithms, including PSO, GA, GOA, and SMO. The experiments were conducted over 20 benchmark medical datasets from the UCI machine learning repository. In the following steps, the datasets and experiment details are presented.

A. DATASETS DETAILS

The experiment was conducted on PC with setting as Table I. In addition, all experiment performaned on 20 medical benchmark datasets from the UCI repository. The Details of the used datasets are presented in Table II.

TABLE I
PC DESCRIPTIONS

Name	Detailed settings
CPU	Core(TM) i7 1.80GHz
RAM	16.0GB
OS	Windows11
Language	MATLAB R2020a

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2022.3174854, IEEE Access

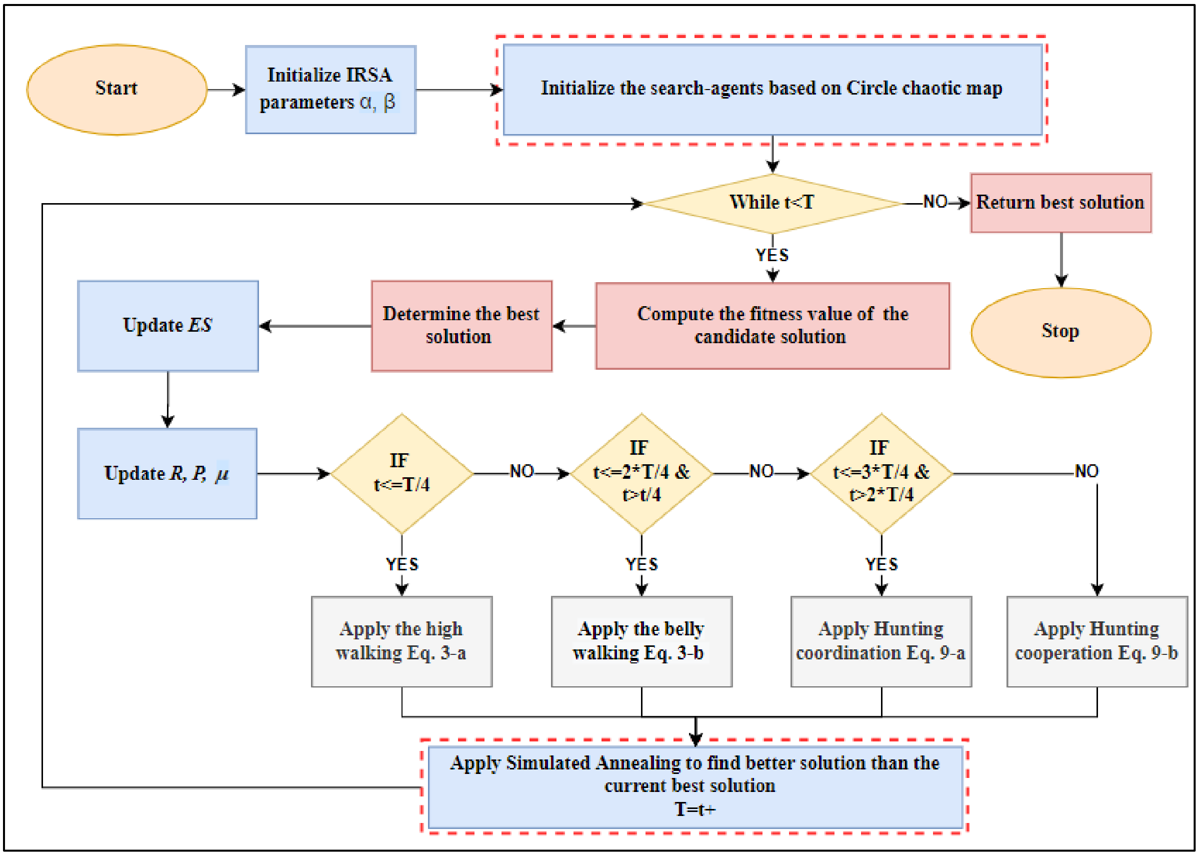


FIGURE 1. The flowchart of the proposed CHHO algorithm using chaotic maps and SA techniques.

TABLE II
THE UCI MEDICAL DATASETS DETAILS

Dataset	No. of Features	No. of Instances
Primry_Tumer	17	339
Hepatitis	20	155
Lymphography	19	148
Breast_Cancer	10	699
Echocardiogram	12	132
Fertility	10	100
Leaf	16	340
Lung_Cancer	57	32
Diabetic	20	1151
ILPD	11	583
Cortex_Nuclear	82	1080
Promoter-gene	58	116
WDBC	31	569
Cervical cancer	36	858
Arrhythmia	279	452
Dermatology	35	366
Heart Disease	75	303
HCV	29	1385
Parkinson	29	1040
HCC	50	165

B. ALGORITHMS AND EXPERIMENTS PARAMETER SETTING

A KNN classifier based on a wrapper method (k-fold cross-validation) was used to validate the fitness performance of the proposed algorithm. The validation technique utilizes k-1 folds to train and one fold to test. The parameter settings of the baseline optimization algorithms PSO, GA, GOA, and SMA are also considered as in Table III. Furthermore, for all algorithms, the search agent was set to 10, and the maximum number of iterations was set to 100. The classification accuracy was selected as a critical metric for evaluating and validating the optimization algorithms performance. In addition, the statistical measures are computed for each algorithm after performing 30 runs. Also, the parameters of the RSA are specified as α is set to 0.1 and β is set to 0.005 by experiments.

TABLE III
THE PARAMETER SETTING OF THE OPTIMIZATION ALGORITHMS

Algorithm	Parameter	Ref.
PSO	Acceleration _constants (C1=2 , C2=2)	[4]
GA	Inertia _Weights (W1=0.9, W2=0.4	[79]
GOA	Crossover _ratio = 0.9	[49]
SMA	Mutation _ratio = 0.1	[80]

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2022.3174854, IEEE Access

C. RESULTS AND DISCUSSION

This section demonstrates the effectiveness of the proposed IRSA by performing two main experiments. The first experiment included the comparison of the proposed IRSA with the standard RSA. The second experiment involved the comparison of IRSA with state-of-the-art algorithms, such as PSO, GA, GOA, and SMA. In all conducted experiments, each algorithm was utilized on all the datasets to verify the solidity of the algorithm within feature dimensionalities. Additionally, the reported results are based on computing the average of 30 runs for every experiment.

1) THE COMPARISON OF RSA AND IRSA

In this section, the proposed IRSA is compared to the original RSA. There are four metrics used in this comparison: classification accuracy, number of selected features, fitness value, and Wilcoxon statistical test (*p*-value). Table IV. displays the experimental results of IRSA in comparison to the original RSA algorithm, the best results are underlined. To determine whether the classification accuracy of IRSA is statistically improved, the *p*-value is computed, where the improvement is considered statistically significant if the *p*-value is smaller than 0.05; otherwise, it is not.

The results show that IRSA has a higher classification accuracy than RSA for the majority of the datasets, while it provided similar accuracy to RSA in one dataset, as illustrated in Table IV. Accordingly, there is no doubt that the application of CM and SA to IRSA enhances its classification performance. In terms of the number of selected features, IRSA outperformed the original RSA by reducing the number of selected features by 61.18 % across all datasets. In addition, IRSA performed better than RSA in all datasets in terms of fitness value. According to the classification accuracy the IRSA significantly outperforms the RSA in 16 datasets. The overall results of classification accuracy, feature selection, and fitness values and *p*-value on most datasets indicate the remarkable improvement accomplished by IRSA.

TABLE IV
THE EXPERIMENTAL RESULTS OF THE IRSA IN COMPARISON TO THE ORIGINAL RSA IN TERMS OF CLASSIFICATION ACCURACY, NUMBER OF SELECTED FEATURES, THE FITNESS VALUE, AND P-VALUE.

Dataset	Classification Accuracy		Selected Features		Fitness		RSA *p*-value
	RSA	IRSA	RSA	IRSA	RSA	IRSA	
Primry_Tumer	0.80348	<u>0.83443</u>	9.7164	<u>8.1656</u>	0.20043	<u>0.1688</u>	<u>0.0092704</u>
Hepatitis	0.75591	<u>0.81935</u>	6.6404	<u>5.7315</u>	0.24475	<u>0.18189</u>	<u>1.57E-07</u>
Lymphography	0.54938	<u>0.62991</u>	7.6745	<u>7.0244</u>	0.45015	<u>0.37031</u>	<u>0.00069002</u>
Breast_Cancer	0.98094	<u>0.98618</u>	4.2158	<u>3.9165</u>	0.023756	<u>0.017941</u>	0.22286
Echocardiogram	0.95318	<u>0.97977</u>	3.5394	<u>2.7711</u>	0.049229	<u>0.02245</u>	<u>0.015196</u>
Fertility	0.92833	<u>0.93333</u>	3.4895	<u>2.9067</u>	0.074357	<u>0.069</u>	0.71347
Leaf	0.63943	<u>0.66356</u>	9.9641	<u>6.0911</u>	0.36366	<u>0.33718</u>	<u>0.00016498</u>
Lung_Cancer	0.74286	<u>0.77143</u>	7.7524	<u>4.9873</u>	0.057637	<u>0.029024</u>	0.21318
Cortex_Nuclear	0.86919	<u>0.93687</u>	22.4803	<u>19.0949</u>	0.1333	<u>0.065905</u>	<u>0.00039216</u>
Promoter-gene	0.87468	<u>0.94069</u>	17.3141	<u>15.69</u>	0.1268	<u>0.061387</u>	<u>1.19E-06</u>
WDBC	0.95457	<u>0.96395</u>	7.9386	<u>4.5843</u>	0.047365	<u>0.037179</u>	<u>0.054654</u>
Cervical cancer	0.96212	<u>0.97552</u>	9.0442	<u>5.9581</u>	0.039793	<u>0.025909</u>	<u>1.69E-05</u>
Dermatology	0.98682	<u>0.99636</u>	21.6855	<u>11.7993</u>	0.019558	<u>0.0070846</u>	<u>0.099005</u>
Heart Disease	0.85309	<u>0.89444</u>	6.0813	<u>4.8087</u>	0.15003	<u>0.10824</u>	<u>0.00012162</u>
HCV	0.30469	<u>0.31649</u>	10.0079	<u>5.917</u>	0.69166	<u>0.67886</u>	<u>5.12E-06</u>
Parkinson	<u>1</u>	<u>1</u>	2.0951	<u>1.6885</u>	0.00038095	<u>0.00035714</u>	0
Arrhythmia	0.66989	<u>0.70422</u>	124.0321	<u>107.545</u>	0.33113	<u>0.29663</u>	<u>0.00018468</u>
Diabetic	0.71126	<u>0.74556</u>	7.5959	<u>5.7983</u>	0.28957	<u>0.25508</u>	<u>1.67E-08</u>
ILPD	0.74762	<u>0.77702</u>	4.1177	<u>3.8117</u>	0.25356	<u>0.22455</u>	<u>6.32E-08</u>
HCC	0.80606	<u>0.8596</u>	13.7395	<u>9.6556</u>	0.19454	<u>0.14093</u>	<u>0.0069343</u>
Average	0.64 % ⇧		61.18 % ⇩		0.64 % ⇩		

In addition, the results displayed in Table IV, show that the enhancement introduced in the initialization phase using the CM method, improved the candidate solution, instead of using the random solution in the original RSA. The possible

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2022.3174854, IEEE Access

reason is that the improved population diversity from random solutions to chaotic solutions using CM balances the convergence speed towards the optimal solution. Also, the enhancement in the exploitation phase with SA provided a better solution. These superiority results prove the IRSA algorithm capability of avoiding the local optima problem and solving the feature selection problem.

2) COMPARISON OF IRSA ALGORITHM WITH OTHER OPTIMIZATION ALGORITHMS

Prior experiments have demonstrated the superiority of IRSA, especially in terms of classification accuracy and fitness value, over the original RSA. This advantage is the result of improving population diversity and maintaining an appropriate balance between exploration and exploitation to prevent local optima. Therefore, to validate the advantage of IRSA, an extended comparison was performed between IRSA and well-known and recent optimization algorithms like PSO, GA, GOA and SMA. To compare the performance of IRSA to the other optimization algorithms, the same evaluation metrics were also used. First, the classification performance was evaluated for the considered algorithms, as illustrated in Table V. Based on the results achieved, IRSA outperformed the other optimization algorithms over all datasets in terms of classification accuracy. The significant results are bolded, while the GOA obtained the last accuracy, PSO ranked a second higher classification accuracy after IRSA with less accuracy 0.59 %, then followed by GA, SMO, GOA with less accuracy respectively. The classification accuracy results of IRSA and all compared algorithm presented in Table V.

TABLE V
IRSA COMPARISON WITH OTHER ALGORITHMS BASED ON AVERAGE CLASSIFICATION ACCURACY IN 30 RUNS

Dataset	**IRSA**	PSO	GA	GOA	SMO
Primry_Tumer	**0.83443**	0.80489	0.78817	0.74983	0.77148
Hepatitis	**0.81935**	0.75484	0.73118	0.68817	0.70968
Lymphography	**0.62991**	0.55072	0.52013	0.4833	0.50195
Breast_Cancer	**0.98618**	0.98023	0.97713	0.97379	0.97761
Echocardiogram	**0.97977**	0.95066	0.92783	0.91662	0.93443
Fertility	**0.93333**	0.90333	0.91	0.88833	0.90333
Leaf	**0.66356**	0.62638	0.6138	0.58508	0.62001
Lung_Cancer	**0.77143**	0.72698	0.70952	0.66349	0.67937
Cortex_Nuclear	**0.93687**	0.91486	0.84856	0.75765	0.77266
Promoter-gene	**0.94069**	0.91876	0.82951	0.79517	0.76688
WDBC	**0.96395**	0.95282	0.94989	0.94462	0.94726
Cervical cancer	**0.97552**	0.96465	0.95475	0.94951	0.94602
Dermatology	**0.99636**	0.98865	0.98365	0.96093	0.98043
Heart Disease	**0.89444**	0.8679	0.82778	0.7963	0.80926
HCV	**0.31649**	0.30048	0.30132	0.28809	0.28616
Parkinson	**1**	0.99407	0.99054	0.96314	0.96651
Arrhythmia	**0.6928**	0.69025	0.67493	0.63576	0.64056
Diabetic	**0.74556**	0.71604	0.70779	0.68128	0.68911
ILPD	**0.77702**	0.74389	0.74786	0.72332	0.7359
HCC	**0.8596**	0.77778	0.76667	0.7303	0.74646
Average	16.72868	16.13718	15.77074	15.18522	15.39527
Rank	1	2	3	5	4

The second evaluation metrics used to evaluate the IRSA performance is the average number of selected features. The best results are bolded in Table VI. Based on the results achieved, IRSA outperformed the other optimization algorithms with the lowest number of selected features in 16 datasets, while GA ranked as second-best performance successful in 4 datasets. The overall ranked results POS, GOA, and SMO show increasing numbers of selected features with 2.85%, 4.15%, 4.8%, respectively.

TABLE VI
IRSA COMPARISON WITH OTHER ALGORITHMS BASED ON AVERAGE NUMBER OF SELECTED FEATURES IN 30 RUNS

Dataset	**IRSA**	PSO	GA	GOA	SMO
Primry_Tumer	8.1656	8.3972	**7.6333**	8.6717	10.4333
Hepatitis	**5.7315**	8.5974	7.6	9.7722	10.8667
Lymphography	**7.0244**	8.2345	8.7667	9.0383	11.3
Breast_Cancer	**3.9165**	4.3259	4.3667	4.6	5.3333
Echocardiogram	**2.7711**	4.9193	3.9	5.094	5.7
Fertility	2.9067	3.611	**2.2**	3.667	3.3333
Leaf	**6.0911**	7.4332	7.4667	7.6563	11.6667
Lung_Cancer	**4.9873**	19.1767	16.3333	20.3368	21.3667
Cortex_Nuclear	**19.0949**	26.2835	25	26.7407	32.6
Promoter-gene	**15.69**	26.9509	26.5667	27.5718	31.1333
WDBC	**4.5843**	13.6194	10.1	14.9618	15.5333
Cervical cancer	**5.9581**	15.2574	11.9	16.5317	15.6667
Dermatology	**11.7993**	16.5665	16.7	17.0585	24.4
Heart Disease	4.8087	6.1018	**4.7**	6.2462	7.1
HCV	**5.917**	12.7761	9.3333	13.8942	15.8333
Parkinson	**1.6885**	12.4453	9.6667	13.9993	14.8333
Arrhythmia	**112.5647**	134.4364	129.9	139.4435	174.7333
Diabetic	**5.7983**	9.0117	7.9333	9.36	10.6667
ILPD	3.8117	4.6142	**3.7333**	4.7671	4.7333
HCC	**9.6556**	21.7416	18.4667	24.5807	26.8667
Average	11.89728	18.25662	16.513335	19.21981	22.009995
Rank	1	3	2	4	5

The third evaluation metrics used to evaluate the IRSA performance is the average fitness value. The fitness function is calculated based on the KNN classifier. The fitness value calculated is based on the classification error rate of the KNN classier, number of selected features and original number of features as presented in Eq. (13). Low fitness value means that the proposed solution obtains good results towards optimal solutions, as this research aims to minimize the features not maximize. The results show that IRSA outperforms all other optimization algorithms in all selected datasets. The PSO ranked as second-best fitness value followed by GA, SMO, GOA respectively. The results presented in Table VII.

TABLE VII
IRSA COMPARISON WITH OTHER ALGORITHMS BASED ON AVERAGE FITNESS VALUE IN 30 RUNS

Dataset	**IRSA**	PSO	GA	GOA	SMO
Primry_Tumer	**0.1688**	0.19761	0.2142	0.25273	0.23237
Hepatitis	**0.18189**	0.24571	0.27013	0.31366	0.29314
Lymphography	**0.37031**	0.44847	0.47995	0.51642	0.49935
Breast_Cancer	**0.017941**	0.023871	0.027491	0.031318	0.028093
Echocardiogram	**0.02245**	0.051812	0.074998	0.086699	0.070101
Fertility	**0.069**	0.097959	0.091544	0.11333	0.099404
Leaf	**0.33718**	0.37444	0.38731	0.41611	0.38397
Lung_Cancer	**0.029024**	0.074048	0.092488	0.13843	0.12324
Cortex_Nuclear	**0.065905**	0.088054	0.15439	0.24443	0.23089
Promoter-gene	**0.061387**	0.084212	0.17345	0.20742	0.23625
WDBC	**0.037179**	0.049257	0.052972	0.059459	0.057391
Cervical cancer	**0.025909**	0.037336	0.048198	0.054204	0.057919

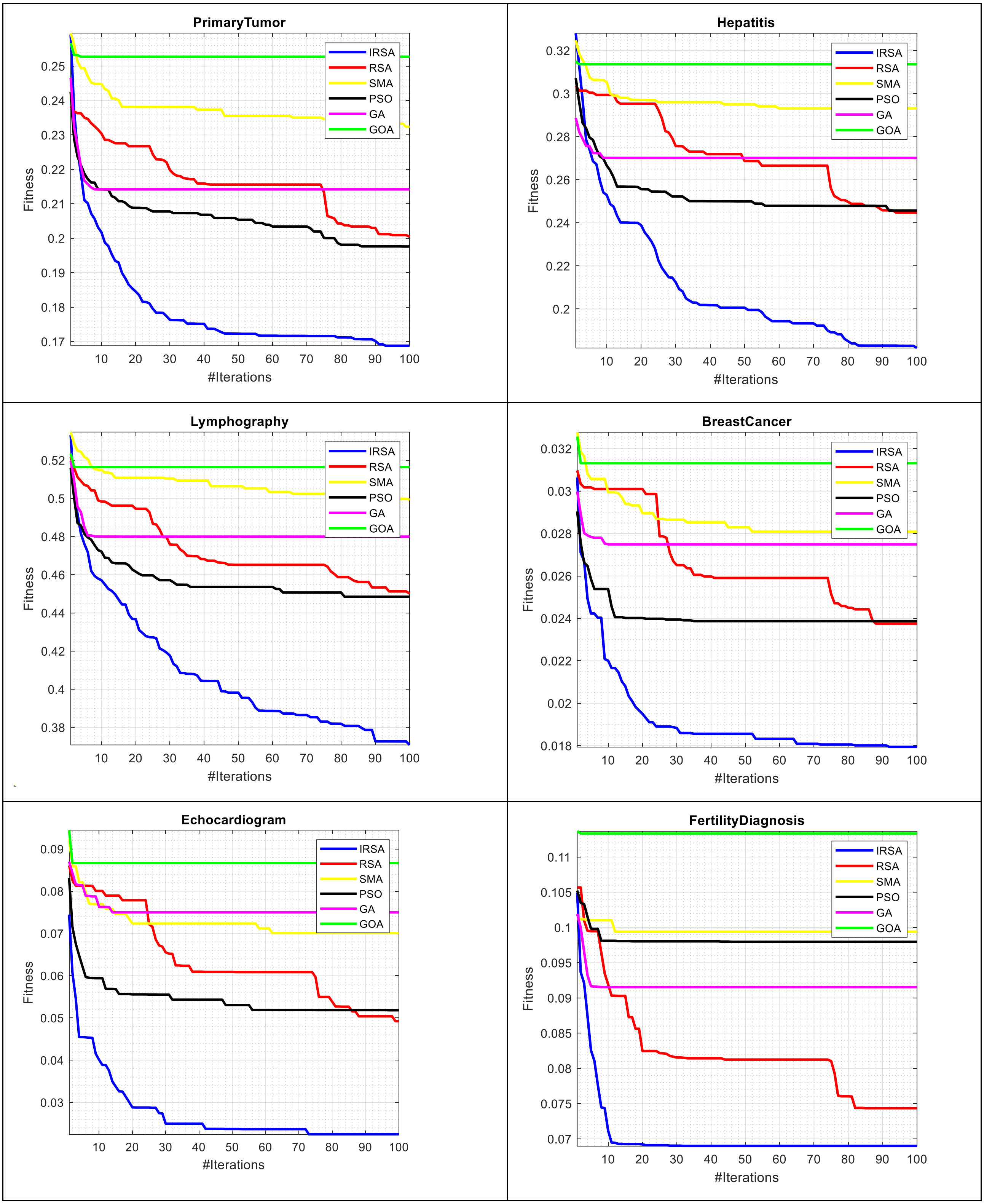
Dermatology	**0.0070846**	0.015152	0.0211	0.043626	0.026548
Heart Disease	**0.10824**	0.13493	0.17412	0.2061	0.19429
HCV	**0.67886**	0.69521	0.69502	0.70944	0.71236
Parkinson	**0.00035714**	0.0082511	0.012813	0.041228	0.038456
Arrhythmia	**0.30818**	0.31075	0.32648	0.36545	0.36211
Diabetic	**0.25508**	0.28478	0.29346	0.32002	0.3134
ILPD	**0.22455**	0.25719	0.25335	0.27815	0.26619
HCC	**0.14093**	0.22239	0.23477	0.27171	0.25648
Average	3.09870674	3.6924921	4.068524	4.659534	4.471352
Rank	**1**	2	3	5	4

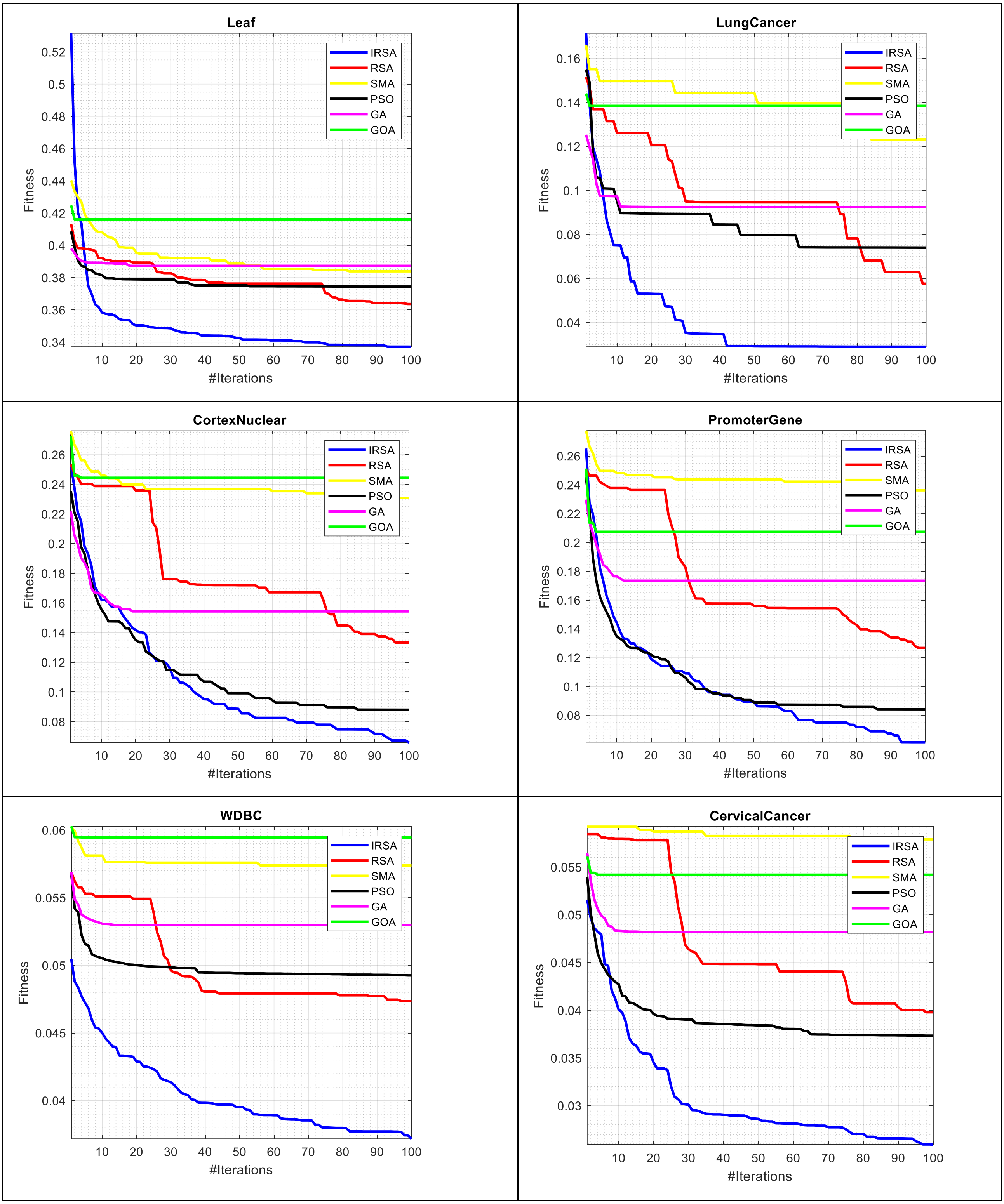
The fourth evaluation metrics used to evaluate the IRSA performance is the Wilcoxon statical test or p-value. The Wilcoxon test was applied to verify the significance of classification accuracy, as displayed in Table VIII, the best results are bolded. The significant results were verified, with a p-value < 0.05. IRSA show significant improvement over all selected algorithm and on the majority of datasets. IRSA outperform the GOA and SMO in all datasets, while it performed significantly in 18 datasets over GA algorithm and 14 datasets over PSO algorithm. The significant results are presented in Table VIII, with bold font. These significant results proved the superiority of IRSA over all the other algorithms. The results signify the capability of IRSA to balance exploration and exploitation. Moreover, it has a better chance of avoiding the trap of local optima, which ultimately leads to a significant improvement in the classification accuracy of IRSA.

TABLE VIII
IRSA COMPARISON WITH OTHER ALGORITHMS BASED ON WILCOXON TEST IN 30 RUNS (P-VALUE $\geq$ 0.05 ARE BOLD)

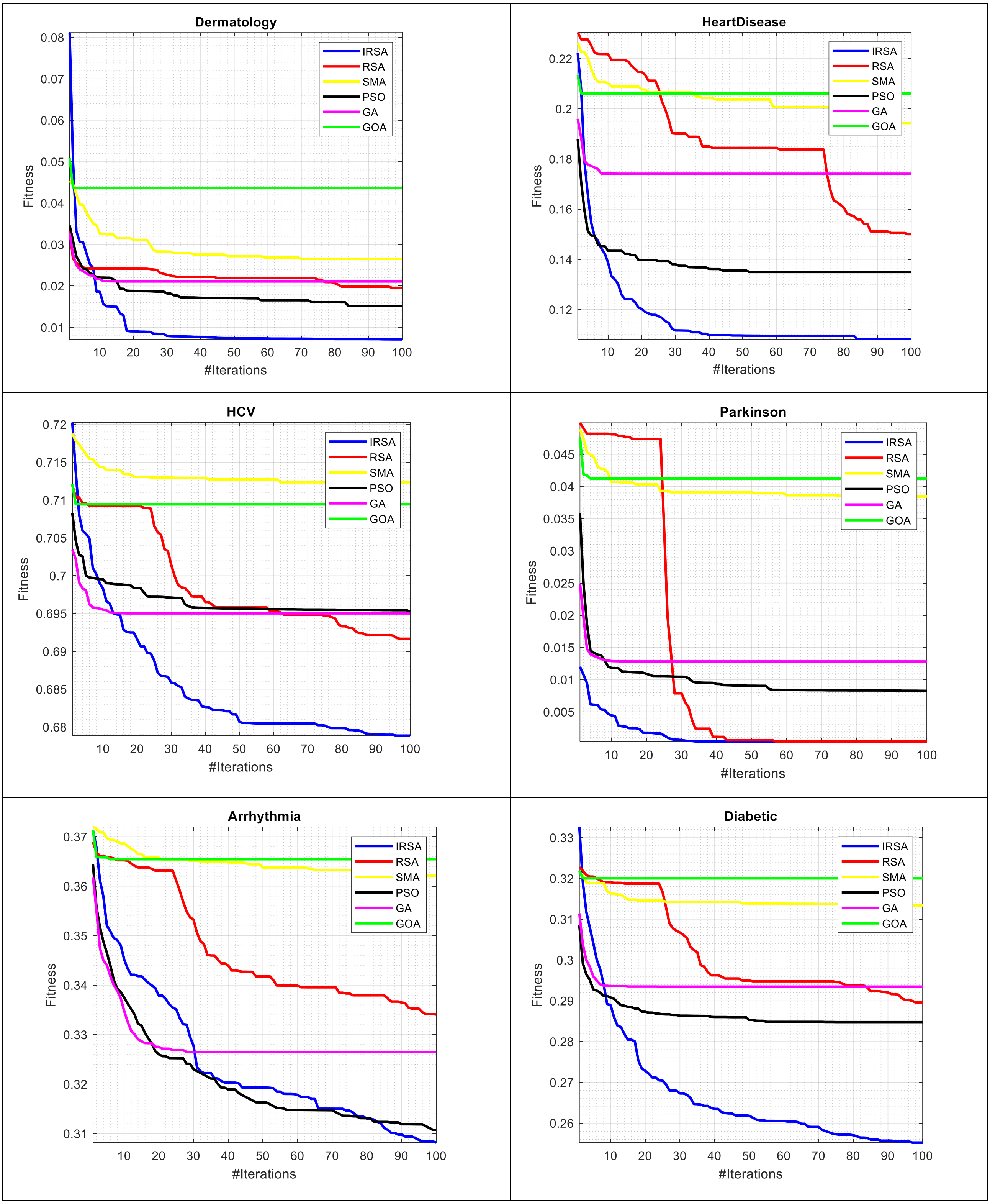
Dataset	PSO	GA	GOA	SMO
Primry_Tumer	**0.0010749**	**5.81E-05**	**7.86E-08**	**2.02E-06**
Hepatitis	**3.78E-07**	**7.45E-09**	**3.47E-11**	**5.38E-11**
Lymphography	**0.0023991**	**3.08E-05**	**8.00E-07**	**4.81E-06**
Breast_Cancer	0.1747	**0.027998**	**0.0048479**	**0.022219**
Echocardiogram	**0.035663**	**1.51E-05**	**1.23E-06**	**0.0024065**
Fertility	**0.056272**	0.10395	**0.0068001**	**0.042474**
Leaf	**3.96E-06**	**4.12E-10**	**6.24E-10**	**7.53E-08**
Lung_Cancer	0.097707	**0.053981**	**0.0004969**	**0.00094845**
Cortex_Nuclear	0.30516	**0.00031061**	**1.61E-08**	**1.86E-06**
Promoter-gene	0.073244	**8.53E-10**	**8.25E-10**	**2.71E-11**
WDBC	**0.036333**	**0.0093197**	**0.00034395**	**0.0037359**
Cervical cancer	**0.00098298**	**8.78E-08**	**2.05E-08**	**8.70E-11**
Dermatology	0.083498	**0.0032166**	**7.72E-08**	**0.0015815**
Heart Disease	**0.0031871**	**9.92E-09**	**1.39E-09**	**1.28E-09**
HCV	**9.01E-07**	**6.01E-07**	**1.64E-10**	**9.34E-11**
Parkinson	**6.54E-05**	**8.71E-07**	**1.18E-12**	**1.58E-11**
Arrhythmia	0.92913	0.069554	**3.34E-06**	**8.29E-06**
Diabetic	**1.63E-07**	**2.29E-09**	**7.00E-11**	**2.85E-11**
ILPD	**2.65E-08**	**8.46E-07**	**5.91E-11**	**2.67E-09**
HCC	**7.05E-05**	**8.95E-06**	**7.74E-08**	**3.11E-08**

Furthermore, the IRSA performance was evaluated based on convergence curves. The convergence curves measure the average fitness value among the iterations. Graphical representation of the convergence curves among all selected optimization algorithms and datasets are illustrated in Figure 2. Based on the results obtained, it is observed that the IRSA is outperformed all other algorithms in convergence curves. Also, it is observed that the performance of PSO is ranked as second-best convergence curves among the datasets. This superiority came from the improvement implemented in the initialization and exploitation phases. The enhancement is done in the initialization phase by applying the chaotic map to accelerate the convergence speed among all iterations. The improved population diversity from random solutions to chaotic solutions balances the convergence speed towards the optimal solution. Also, the enhancement in the exploitation phase provided a high fitness value. These superiority results are a clue of the higher algorithm capability to avoid the local optima problem and solve the feature selection problem.




Leaf
LungCancer
CortexNuclear
PromoterGene
WDBC
CervicalCancer
IRSA
RSA
SMA
PSO
GA
GOA
Fitness
#Iterations

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2022.3174854, IEEE Access



This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2022.3174854, IEEE Access

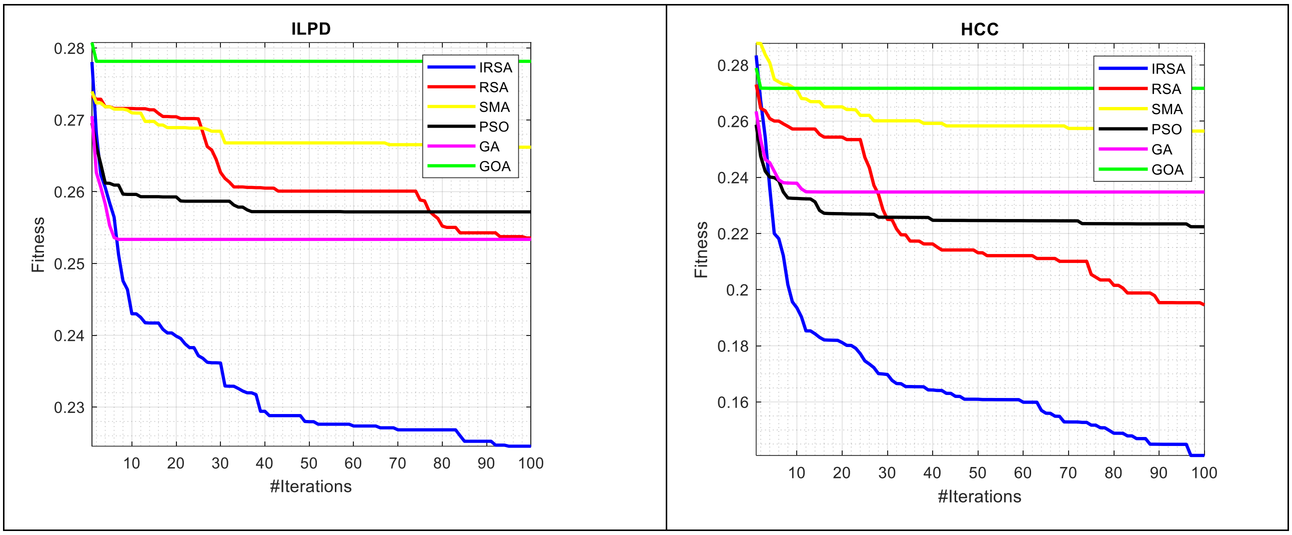


FIGURE 2. Graphical Representation of the Convergence-Curves, Considered to Evaluate the Convergence Speed of IRSA Among the selected Optimization Algorithms on 20 datasets

3) THE LIMITATIONS OF CHHO ALGORITHM

The superiority of IRSA comes from the improvements introduced to the RSA algorithm. Improving the exploration phase (global search) controls the algorithm's population diversity. At the same time, the improvement of the exploitation phase (local search) prevents the local search problem. However, this has some limitations; applying the SA algorithm in each iteration to select the best solution and avoid the local optima problem increases the execution time of the algorithm. As the results show, the average time of algorithm run reaches 6.4 % higher than the second-best algorithms PSO. It is worth mentioning that the choice of optimization algorithm (and its parallelization) highly depends on the properties of the objective function and constraints.

VI. CONCLUSION AND FUTURE DIRECTIONS

The Reptile Search Algorithm (RSA) is a novel population-based optimization algorithm. RSA is inspired by the swarm-based comparison meta-heuristic algorithm that mimics the Crocodiles' encircling and hunting behavior in the wild. This study proposes an improved version of RSA, named IRSA, which adds two main improvements to the original RSA: (1) applying the chaos theory at the initialization phase of RSA to enhance its exploration capabilities in the search space. And (2) combining the Simulated Annealing (SA) algorithm with the exploitation process to avoid the local optima problem. These two improvements substantially increased the exploration and exploitation search capability of IRSA. Specifically, the use of Circle chaotic map improves the population diversity, whereas SA algorithm avoids trapping in local optima. Additionally, these two improvements to IRSA provides a good balance when transferring between exploration and exploitation search. The performance of IRSA was evaluated over 20 medical benchmark datasets from the UCI repository. Moreover, IRSA was compared with other well-known and recent optimization algorithms, including PSO, GA, GOA, and SMA. Four evaluation metrics were used in the comparison: classification accuracy, fitness value, number of selected features, and p-value. According to these metrics, IRSA is superior to all other algorithms. Furthermore, the results also indicated that IRSA was capable of improving the computational accuracy and accelerating the convergence rate. In addition, the results showed that IRSA was able to minimize the number of features selected for the majority of the datasets. Based on the obtained results, IRSA can be employed as a technique for real-world application. For future work, IRSA could be further developed based on the filter feature selection method used in conjunction with IRSA to deal with real-world datasets. finally, IRSA could possibly be applied to developing other optimization algorithms.

ACKNOWLEDGMENT

The contributors would like to acknowledge the editors, reviewers, and Prof.Mohammad Tubishat for his valuable comments.

REFERENCES

[1] R. L. Siegel, K. D. Miller, H. E. Fuchs, and A. Jemal, “Cancer statistics, 2022,” *CA. Cancer J. Clin.*, vol. 72, no. 1, pp. 7–33, 2022, doi: 10.3322/caac.21708.

[2] B. Remeseiro and V. Bolon-Canedo, “A review of feature selection methods in medical applications,” *Comput. Biol. Med.*, vol. 112, no. February, 2019, doi: 10.1016/j.compbiomed.2019.103375.

[3] P. M. Shakeel, A. Tolba, Z. Al-Makhadmeh, and M. M. Jaber, “Automatic detection of lung cancer from biomedical data set using discrete AdaBoost optimized ensemble learning generalized neural networks,” *Neural Comput. Appl.*, vol. 32, no. 3,

pp. 777–790, 2020, doi: 10.1007/s00521-018-03972-2.
[4] T. A. Khan, K. Zain-Ul-Abideen, and S. H. Ling, "A modified particle swarm optimization algorithm used for feature selection of UCI biomedical data sets," *60th Int. Sci. Conf. Inf. Technol. Manag. Sci. Riga Tech. Univ. ITMS 2019 - Proc.*, pp. 1–4, 2019, doi: 10.1109/ITMS47855.2019.8940760.
[5] E. Pashaei and E. Pashaei, "An efficient binary chimp optimization algorithm for feature selection in biomedical data classification," *Neural Comput. Appl.*, vol. 34, no. 8, pp. 6427–6451, 2022, doi: 10.1007/s00521-021-06775-0.
[6] J. Park, M. W. Park, D. W. Kim, and J. Lee, "Multi-population genetic algorithm for multilabel feature selection based on label complementary communication," *Entropy*, vol. 22, no. 8, 2020, doi: 10.3390/E22080876.
[7] M. Alweshah, S. Al Khalaileh, B. B. Gupta, A. Almomani, A. I. Hammouri, and M. A. Al-Betar, "The monarch butterfly optimization algorithm for solving feature selection problems," *Neural Comput. Appl.*, vol. 0, 2020, doi: 10.1007/s00521-020-05210-0.
[8] M. H. Waseem *et al.*, "On the Feature Selection Methods and Reject Option Classifiers for Robust Cancer Prediction," *IEEE Access*, vol. 7, pp. 141072–141082, 2019, doi: 10.1109/access.2019.2944295.
[9] S. S. Shreem, H. Turabieh, S. Al Azwari, and F. Baothman, "Enhanced binary genetic algorithm as a feature selection to predict student performance," *Soft Comput.*, vol. 26, no. 4, pp. 1811–1823, 2022, doi: 10.1007/s00500-021-06424-7.
[10] G. Hu, B. Du, X. Wang, and G. Wei, "An enhanced black widow optimization algorithm for feature selection," *Knowledge-Based Syst.*, vol. 235, 2022, doi: 10.1016/j.knosys.2021.107638.
[11] S. Mahalakshmi and T. Velmurugan, "Detection of brain tumor by particle swarm optimization using image segmentation," *Indian J. Sci. Technol.*, vol. 8, no. 22, 2015, doi: 10.17485/ijst/2015/v8i22/79092.
[12] A. A. Ewees *et al.*, "Boosting arithmetic optimization algorithm with genetic algorithm operators for feature selection: Case study on cox proportional hazards model," *Mathematics*, vol. 9, no. 18, 2021, doi: 10.3390/math9182321.
[13] A. M. Anter and M. Ali, "Feature selection strategy based on hybrid crow search optimization algorithm integrated with chaos theory and fuzzy c-means algorithm for medical diagnosis problems," *Soft Comput.*, vol. 24, no. 3, pp. 1565–1584, 2020, doi: 10.1007/s00500-019-03988-3.
[14] M. Prabukumar, L. Agilandeeswari, and K. Ganesan, "An intelligent lung cancer diagnosis system using cuckoo search optimization and support vector machine classifier," *J. Ambient Intell. Humaniz. Comput.*, vol. 10, no. 1, pp. 267–293, 2019, doi: 10.1007/s12652-017-0655-5.
[15] M. Tubishat, M. A. M. Abushariah, N. Idris, and I. Aljarah, "Improved whale optimization algorithm for feature selection in Arabic sentiment analysis," *Appl. Intell.*, vol. 49, no. 5, pp. 1688–1707, 2019, doi: 10.1007/s10489-018-1334-8.
[16] O. Rostami and M. Kaveh, "Optimal feature selection for SAR image classification using biogeography-based optimization (BBO), artificial bee colony (ABC) and support vector machine (SVM): a combined approach of optimization and machine learning," *Comput. Geosci.*, vol. 25, no. 3, pp. 911–930, 2021, doi: 10.1007/s10596-020-10030-1.
[17] A. Jović, K. Brkić, and N. Bogunović, "A review of feature selection methods with applications," *2015 38th Int. Conv. Inf. Commun. Technol. Electron. Microelectron. MIPRO 2015 - Proc.*, no. May, pp. 1200–1205, 2015, doi: 10.1109/MIPRO.2015.7160458.
[18] A. M. Usman, "Filter-Based Feature Selection Using Information Theory and Binary Cuckoo Optimisation Algorithm."
[19] J. Wang, J. Xu, C. Zhao, Y. Peng, and H. Wang, "An ensemble feature selection method for high-dimensional data based on sort aggregation," *Syst. Sci. Control Eng.*, vol. 7, no. 2, pp. 32–39, 2019, doi: 10.1080/21642583.2019.1620658.
[20] Z. M. Elgamal, N. M. Yasin, A. Q. M. Sabri, R. Sihwail, M. Tubishat, and H. Jarrah, "Improved equilibrium optimization algorithm using elite opposition-based learning and new local search strategy for feature selection in medical datasets," *Computation*, vol. 9, no. 6, 2021, doi: 10.3390/computation9060068.
[21] M. Tubishat, N. Idris, L. Shuib, M. A. M. Abushariah, and S. Mirjalili, "Improved Salp Swarm Algorithm based on opposition based learning and novel local search algorithm for feature selection," *Expert Syst. Appl.*, vol. 145, 2020, doi: 10.1016/j.eswa.2019.113122.
[22] J. Too and S. Mirjalili, "General Learning Equilibrium Optimizer: A New Feature Selection Method for Biological Data Classification," *Appl. Artif. Intell.*, vol. 00, no. 00, pp. 1–17, 2020, doi: 10.1080/08839514.2020.1861407.
[23] M. Tubishat, N. Idris, L. Shuib, M. A. M. Abushariah, and S. Mirjalili, "Improved Salp Swarm Algorithm based on opposition based learning and novel local search algorithm for feature selection," *Expert Syst. Appl.*, vol. 145, p. 113122, 2020, doi: 10.1016/j.eswa.2019.113122.
[24] M. Allam and M. Nandhini, "Optimal feature selection using binary teaching learning based optimization algorithm," *J. King Saud Univ. -*

Comput. Inf. Sci., vol. 34, no. 2, pp. 329–341, 2022, doi: 10.1016/j.jksuci.2018.12.001.
[25] R. Nasfi and N. Bouguila, "A novel feature selection method using generalized inverted Dirichlet-based HMMs for image categorization," *Int. J. Mach. Learn. Cybern.*, no. 0123456789, 2022, doi: 10.1007/s13042-022-01529-3.
[26] S. Nayeri, R. Tavakkoli-Moghaddam, Z. Sazvar, and J. Heydari, "A heuristic-based simulated annealing algorithm for the scheduling of relief teams in natural disasters," *Soft Comput.*, vol. 26, no. 4, pp. 1825–1843, 2022, doi: 10.1007/s00500-021-06425-6.
[27] J. Li *et al.*, "IBDA: Improved Binary Dragonfly Algorithm with Evolutionary Population Dynamics and Adaptive Crossover for Feature Selection," *IEEE Access*, vol. 8, pp. 108032–108051, 2020, doi: 10.1109/ACCESS.2020.3001204.
[28] A. Tiwari and A. Chaturvedi, "A hybrid feature selection approach based on information theory and dynamic butterfly optimization algorithm for data classification," *Expert Syst. Appl.*, vol. 196, no. November 2021, p. 116621, 2022, doi: 10.1016/j.eswa.2022.116621.
[29] K. Chen, F.-Y. Zhou, and X.-F. Yuan, "Hybrid particle swarm optimization with spiral-shaped mechanism for feature selection," *Expert Syst. Appl.*, vol. 128, pp. 140–156, Aug. 2019, doi: 10.1016/J.ESWA.2019.03.039.
[30] R. Sihwail, K. Omar, K. A. Z. Ariffin, and M. Tubishat, "Improved Harris Hawks Optimization Using Elite Opposition-Based Learning and Novel Search Mechanism for Feature Selection," *IEEE Access*, vol. 8, pp. 121127–121145, 2020, doi: 10.1109/ACCESS.2020.3006473.
[31] T. Alhersh, A. Alorainy, B. B. Samir, H. R. H. Al-Absi, and B. Bouzid, "Species identification using part of DNA sequence: Evidence from machine learning algorithms," *EAI Int. Conf. Bio-inspired Inf. Commun. Technol.*, no. January, 2015, doi: 10.4108/eai.3-12-2015.2262476.
[32] L. Abualigah and A. Diabat, *Improved multi-core arithmetic optimization algorithm-based ensemble mutation for multidisciplinary applications*. Springer US, 2022.
[33] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 67–82, 1997, doi: 10.1109/4235.585893.
[34] A. J. Lockett, "No free lunch theorems," *Nat. Comput. Ser.*, vol. 1, no. 1, pp. 287–322, 2020, doi: 10.1007/978-3-662-62007-6_12.
[35] F. Gul, I. Mir, L. Abualigah, and P. Sumari, "Multi-Robot Space Exploration: An Augmented Arithmetic Approach," *IEEE Access*, vol. 9, pp. 107738–107750, 2021, doi: 10.1109/ACCESS.2021.3101210.
[36] L. Abualigah, M. A. Elaziz, P. Sumari, Z. W. Geem, and A. H. Gandomi, "Reptile Search Algorithm (RSA): A nature-inspired meta-heuristic optimizer," *Expert Syst. Appl.*, vol. 191, no. November 2021, p. 116158, 2022, doi: 10.1016/j.eswa.2021.116158.
[37] A. Faramarzi, M. Heidarinejad, B. Stephens, and S. Mirjalili, "Equilibrium optimizer: A novel optimization algorithm," *Knowledge-Based Syst.*, vol. 191, p. 105190, 2020, doi: 10.1016/j.knosys.2019.105190.
[38] F. A. Hashim and A. G. Hussien, "Snake Optimizer: A novel meta-heuristic optimization algorithm," *Knowledge-Based Syst.*, vol. 242, p. 108320, 2022, doi: 10.1016/j.knosys.2022.108320.
[39] L. Abualigah, A. Diabat, S. Mirjalili, M. Abd Elaziz, and A. H. Gandomi, "The Arithmetic Optimization Algorithm," *Comput. Methods Appl. Mech. Eng.*, vol. 376, p. 113609, 2021, doi: 10.1016/j.cma.2020.113609.
[40] R. Eberhart and J. Kennedy, "A New Optimizer Using Particle Swarm Theory," pp. 39–43.
[41] S. Mirjalili and A. Lewis, "The Whale Optimization Algorithm," *Adv. Eng. Softw.*, vol. 95, pp. 51–67, 2016, doi: 10.1016/j.advengsoft.2016.01.008.
[42] S. Mirjalili, S. Mohammad, and A. Lewis, "Grey Wolf Optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, 2014, doi: 10.1016/j.advengsoft.2013.12.007.
[43] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, "Harris hawks optimization: Algorithm and applications," *Futur. Gener. Comput. Syst.*, vol. 97, pp. 849–872, 2019, doi: 10.1016/j.future.2019.02.028.
[44] S. Mirjalili, A. H. Gandomi, S. Z. Mirjalili, S. Saremi, H. Faris, and S. M. Mirjalili, "Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems," *Adv. Eng. Softw.*, vol. 114, pp. 163–191, 2017, doi: 10.1016/j.advengsoft.2017.07.002.
[45] M. Mafarja and S. Mirjalili, "Whale optimization approaches for wrapper feature selection," *Appl. Soft Comput. J.*, vol. 62, pp. 441–453, 2018, doi: 10.1016/j.asoc.2017.11.006.
[46] S. Arora and P. Anand, "Binary butterfly optimization approaches for feature selection," *Expert Syst. Appl.*, vol. 116, pp. 147–160, 2019, doi: 10.1016/j.eswa.2018.08.051.
[47] H. M. Zawbaa, E. Emary, and B. Parv, "Feature selection based on antlion optimization algorithm," *Proc. 2015 IEEE World Conf. Complex Syst. WCCS 2015*, 2016, doi: 10.1109/ICoCS.2015.7483317.
[48] K. Chen, F. Y. Zhou, and X. F. Yuan, "Hybrid particle swarm optimization with spiral-shaped mechanism for feature selection," *Expert Syst. Appl.*, vol. 128, pp. 140–156, 2019, doi: 10.1016/j.eswa.2019.03.039.
[49] A. A. Ewees, M. Abd Elaziz, and E. H. Houssein,

"Improved grasshopper optimization algorithm using opposition-based learning," *Expert Syst. Appl.*, vol. 112, pp. 156–172, 2018, doi: 10.1016/j.eswa.2018.06.023.

[50] A. A. Ewees, R. R. Mostafa, R. M. Ghoniem, and M. A. Gaheen, *Improved seagull optimization algorithm using Lévy flight and mutation operator for feature selection*, vol. 8. Springer London, 2022.

[51] S. Song *et al.*, "Dimension decided Harris hawks optimization with Gaussian mutation: Balance analysis and diversity patterns," *Knowledge-Based Syst.*, vol. 215, p. 106425, 2021, doi: 10.1016/j.knosys.2020.106425.

[52] G. Kaur and S. Arora, "Chaotic whale optimization algorithm," *J. Comput. Des. Eng.*, vol. 5, no. 3, pp. 275–284, Jul. 2018, doi: 10.1016/J.JCDE.2017.12.006.

[53] G. I. Sayed, A. E. Hassanien, and A. T. Azar, "Feature selection via a novel chaotic crow search algorithm," *Neural Comput. Appl.*, vol. 31, no. 1, pp. 171–188, 2019, doi: 10.1007/s00521-017-2988-6.

[54] Z. Wang and Y. Zhang, "Application of chaos optimization algorithm to nonlinear constrained programming," *2010 Int. Conf. E-Product E-Service E-Entertainment, ICEEE2010*, pp. 1–4, 2010, doi: 10.1109/ICEEE.2010.5660461.

[55] M. Kohli and S. Arora, "Chaotic grey wolf optimization algorithm for constrained optimization problems," *J. Comput. Des. Eng.*, vol. 5, no. 4, pp. 458–472, 2018, doi: 10.1016/j.jcde.2017.02.005.

[56] Z. M. Elgamal, N. B. M. Yasin, M. Tubishat, M. Alswaitti, and S. Mirjalili, "An Improved Harris Hawks Optimization Algorithm With Simulated Annealing for Feature Selection in the Medical Field," *IEEE Access*, vol. 8, pp. 186638–186652, 2020, doi: 10.1109/access.2020.3029728.

[57] S. Arora and P. Anand, "Chaotic grasshopper optimization algorithm for global optimization," *Neural Comput. Appl.*, vol. 31, no. 8, pp. 4385–4405, 2019, doi: 10.1007/s00521-018-3343-2.

[58] G. Kaur and S. Arora, "Chaotic whale optimization algorithm," *J. Comput. Des. Eng.*, vol. 5, no. 3, pp. 275–284, 2018, doi: 10.1016/j.jcde.2017.12.006.

[59] V. Hayyolalam and A. A. Pourhaji Kazem, "Black Widow Optimization Algorithm: A novel meta-heuristic approach for solving engineering optimization problems," *Eng. Appl. Artif. Intell.*, vol. 87, no. July 2019, p. 103249, 2020, doi: 10.1016/j.engappai.2019.103249.

[60] B. A. Hassan, "CSCF: a chaotic sine cosine firefly algorithm for practical application problems," *Neural Comput. Appl.*, vol. 33, no. 12, pp. 7011–7030, 2021, doi: 10.1007/s00521-020-05474-6.

[61] M. Tubishat *et al.*, "Dynamic Salp swarm algorithm for feature selection," *Expert Syst. Appl.*, vol. 164, no. August 2020, p. 113873, 2021, doi: 10.1016/j.eswa.2020.113873.

[62] L. A. Demidova and A. V. Gorchakov, "A study of chaotic maps producing symmetric distributions in the fish school search optimization algorithm with exponential step decay," *Symmetry (Basel).*, vol. 12, no. 5, 2020, doi: 10.3390/SYM12050784.

[63] A. A. Ewees and M. A. Elaziz, "Performance analysis of Chaotic Multi-Verse Harris Hawks Optimization: A case study on solving engineering problems," *Eng. Appl. Artif. Intell.*, vol. 88, no. December 2018, p. 103370, 2020, doi: 10.1016/j.engappai.2019.103370.

[64] Y. Wang, T. Wang, S. Dong, and C. Yao, "An Improved Grey-Wolf Optimization Algorithm Based on Circle Map," *J. Phys. Conf. Ser.*, vol. 1682, no. 1, 2020, doi: 10.1088/1742-6596/1682/1/012020.

[65] F. Kutlu Onay and S. B. Aydemir, "Chaotic hunger games search optimization algorithm for global optimization and engineering problems," *Math. Comput. Simul.*, vol. 192, pp. 514–536, 2022, doi: 10.1016/j.matcom.2021.09.014.

[66] S. B. Aydemir, "A novel arithmetic optimization algorithm based on chaotic maps for global optimization," *Evol. Intell.*, no. 1, 2022, doi: 10.1007/s12065-022-00711-4.

[67] X. Li and J. Wang, "Chaotic arithmetic optimization algorithm," 2022.

[68] A. N. Jadhav and N. Gomathi, "WGC: Hybridization of exponential grey wolf optimizer with whale optimization for data clustering," *Alexandria Eng. J.*, vol. 57, no. 3, pp. 1569–1584, 2018, doi: 10.1016/j.aej.2017.04.013.

[69] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science (80-.).*, vol. 220, no. 4598, pp. 671–680, 1983, doi: 10.1126/science.220.4598.671.

[70] P. Moradi and M. Gholampour, "A hybrid particle swarm optimization for feature subset selection by integrating a novel local search strategy," *Appl. Soft Comput. J.*, vol. 43, pp. 117–130, 2016, doi: 10.1016/j.asoc.2016.01.044.

[71] C. Yu, A. A. Heidari, and H. Chen, "A quantum-behaved simulated annealing algorithm-based moth-flame optimization method," *Appl. Math. Model.*, vol. 87, pp. 1–19, 2020, doi: 10.1016/j.apm.2020.04.019.

[72] M. M. Mafarja and S. Mirjalili, "Hybrid Whale Optimization Algorithm with simulated annealing for feature selection," *Neurocomputing*, vol. 260, pp. 302–312, 2017, doi: 10.1016/j.neucom.2017.04.053.

[73] S. Kassaymeh, M. Al-Laham, M. A. Al-Betar, M. Alweshah, S. Abdullah, and S. N. Makhadmeh, "Backpropagation Neural Network optimization and software defect estimation modelling using a hybrid Salp Swarm optimizer-based Simulated

Annealing Algorithm," *Knowledge-Based Syst.*, vol. 244, p. 108511, 2022, doi: 10.1016/j.knosys.2022.108511.

[74] G. I. Sayed, A. Darwish, and A. E. Hassanien, "A New Chaotic Whale Optimization Algorithm for Features Selection," *J. Classif.*, vol. 35, no. 2, pp. 300–344, 2018, doi: 10.1007/s00357-018-9261-2.

[75] W. M. Zheng, "Kneading plane of the circle map," *Chaos, Solitons and Fractals*, vol. 4, no. 7, pp. 1221–1233, 1994, doi: 10.1016/0960-0779(94)90033-7.

[76] D. Long, J. Viovy, and A. Ajdari, "A comparison of simulated annealing cooling strategies This," *J. physics. Condens. matter*, vol. 8, p. 9471, 1996.

[77] "The Yarpiz Project resource of academic and professional scientific source codes , 2018, [online] Available: http://www.yarpiz.com.," p. 2018, 2018.

[78] E. Emary, H. M. Zawbaa, and A. E. Hassanien, "Binary ant lion approaches for feature selection," *Neurocomputing*, vol. 213, pp. 54–65, 2016, doi: 10.1016/j.neucom.2016.03.101.

[79] M. Ghosh, R. Guha, I. Alam, P. Lohariwal, D. Jalan, and R. Sarkar, "Binary Genetic Swarm Optimization: A Combination of GA and PSO for Feature Selection," *J. Intell. Syst.*, vol. 29, no. 1, pp. 1598–1610, 2019, doi: 10.1515/jisys-2019-0062.

[80] S. Li, H. Chen, M. Wang, A. A. Heidari, and S. Mirjalili, "Slime mould algorithm: A new method for stochastic optimization," *Futur. Gener. Comput. Syst.*, vol. 111, pp. 300–323, 2020, doi: 10.1016/j.future.2020.03.055.

ZENAB ELGAMAL Received her PhD degree in Computer Science (Information System, Bioinformatic) from the University of Malaya in 2022. Furthermore, MSc in Computer Science from AAST, Egypt in 2011. Moreover, BSc degree in Information System from AAST, Egypt in 2008. was working as Lecturer in several universities. The current research interests include Optimization Algorithm., Artificial Intelligence, Machine Learning, Data Science and Bioinformatics. zenabelgamal@siswa.um.edu.my

AZNUL QALID MD SABRI currently holds the position of Senior Lecturer at the Department of Artificial Intelligence, Faculty of Computer Science and Information Technology (FCSIT), University of Malaya, Malaysia. He is a graduate of the prestigious Erasmus Mundus Master in Vision and Robotics (ViBot), a Master program jointly coordinated by three different universities (University of Burgundy, France, University of Girona, Spain and Heriot-Watt University Edinburgh, United Kingdom). He then completed his Master's degree by performing a research internship program at the Commonwealth Scientific Research Organization (CSIRO) in Brisbane, Australia focusing on Medical Imaging. Next, he pursued his PhD on the topic of "Human Action Recognition" (completed with distinction, *très honorable*), under a program jointly offered by a well-known research institution in France, Mines de Douai (a research lab) and the reputable University of Picardie Jules Verne, Amiens, France. He is an active researcher in the field of Artificial Intelligence, having published in multiple international conferences as well as international journals. His main research interest is in the field of Computer Vision, Robotics and Machine Learning. He is part of the pioneering members of FCSIT's COVIRO (Cognitive, Vision and Robotics) research group and is currently the principal investigator of multiple research grants. aznulqalid@um.edu.my

MOHAMMAD TUBISHAT received his PhD degree in Computer Science (Artificial Intelligence – Natural Language Processing) from the University of Malaya in 2019. In addition, MSc in Computer and Information Sciences from Yarmouk University in 2004. Furthermore, BSc degree in Computer Science from Yarmouk University in 2002. His research interests include natural language processing, data mining, artificial intelligence, machine learning, optimization algorithms, data science, and sentiment analysis. He is now working as an assistant professor at Zayed University, UAE. mohammad.tubishat@zu.ac.ae

DINA TBAISHAT has a PhD in Information Science from Aberystwyth University / UK in 2012. She has also a MSc degree in Software Engineering from University of the West of England awarded in 2006. She completed her bachelor degree from University of Jordan in 2004 in Computer Information Systems. While Dina conducted research in business process modelling during her PhD studies, she investigated the use of neural networks in off-line signature recognition for her Master's dissertation. After 7 years teaching at University of Jordan, Dina is currently teaching at Zayed University - College of Technological Innovation. Active and founding member of the iOntoBioethics Research Group. Her research interests include business process modelling, process architecture, requirements engineering, and the latest work included using machine learning for generating ontological bioethics topics from the incrementally developing bioethics publications. dina.Tbaishat@zu.ac.ae

OSAMA AHMAD ALOMARI is currently working as Postdoctoral Researcher in the Machine Learning and Arabic Language Processing research group at University of Sharjah. He received his B.Sc. from Al-al Bayt University, Computer Science, Jordan, in 2005 and MSc in computer science from the National University of Malaysia(UKM) in 2012. He obtained a Ph.D. degree from Universiti Sains Malaysia, Computer Science (Artificial Intelligence), in 2018. In general, his research interests are Optimization, Pattern recognition, Feature selection, Microarray data analysis, Machine, and Deep Learning. oalomari@sharjah.ac.ae

SHARIF NASER MAKHADMEH Full-time research associate (FTRA) at Artificial Intelligence Research Center (AIRC) - Ajman University, United Arab Emirates. He received his B.Sc. in Computer Science from Yarmouk University, Jordan, in 2013 and MSc in Information Technology from the Universiti Utara Malaysia (UUM), Malaysia, in 2015. He obtained a PhD degree in Artificial Intelligence from Universiti Sains Malaysia (USM), Malaysia, in 2020. His research interests include optimization algorithms, artificial intelligence, engineering and scheduling problem, smart home. s.makhadmeh@ajman.ac.ae