Received: January 14, 2021. Revised: April 20, 2021.

INASS

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Population and Global Search Improved Squirrel Search Algorithm for Feature Selection in Big Data Classification

Issa Mohammed Saeed Ali[1]* **Mukunthan Balakrishnan[2]**

[1]*Department of Computer Science, School of Computing, Sri Ramakrishna College of Arts and Science, Coimbatore - 641006, Tamil nadu, India*

* Corresponding author's Email: issamohammed.cs@gmail.com

Abstract: The analytics techniques in Big Data are extensively employed as an alternative to generalized for data mining due to the huge volumes of large-scale high dimensional data. Feature selection techniques eradicate the redundant and inappropriate features to decrease the data dimensionality and increase the classifiers' efficiency. However, the traditional feature selection strategies are dearth of scalability to handle the unique characteristics of large-scale data and extract the valuable features within the restricted time. This article proposed a feature selection algorithm centered on the Population and Global Search Improved Squirrel Search Algorithm (PGS-ISSA) that tackles the problem of local optimum and reduce convergence rate in standard Squirrel Search Algorithm (SSA). The novelty of this proposed PGS-ISSA is the introduction of chaos theory to improve population initialization so that the search space is increased. Then the acceleration coefficients are used in the position update equations to improve the convergence rate in local search process while inertia weight is also applied to optimally balance the exploration and exploitation in SSA. PGS-ISSA employs the fitness function based on the minimum error rate for ensuring the selection of best features that improve the classification accuracy. The proposed PGS-ISSA based feature section algorithm is evaluated by using Support Vector Machine (SVM) classifier implemented in MATLAB tool to address the big data classification problem. The experiments performed on both small and large-scale datasets illustrated that the suggested PGS-ISSA enhances the classification accuracy by 1.7% to 5.4% better than other compared models through effective handling of the big data problems. The results obtained for the bigger Higgs dataset shows that the proposed PGS-ISSA achieved high performance than the standard SSA, existing ISSA models and other prominent optimization-based feature selection algorithms with 64.72% accuracy, 67.3194% precision, 62.1528% recall, 62.3026% f-measure, 82.7226% specificity and consumed less time of 140.1366 seconds. PGS-ISSA also achieved comparatively better results for the other benchmark datasets with 0.3% to 6% improvement on statistical metrics and 10% to 25% reduction in execution time.

Keywords: Big data analytics, High dimensional data, Feature selection, Improved Squirrel Search Algorithm, Chaos theory, Acceleration coefficients, Inertia weights.

1. Introduction

Modern big data era has created a situation where massive volumes of high dimensional data are persistent in almost all major domains including the health care, commerce, social media, transportation, education and informatics. The online repositories which provide data for research purposes are very common sources to learn the gigantic growth of data sample size and features counts over the time [1]. Application of the traditional algorithms for data mining and the algorithms for machine learning on the big data has provided significantly efficient results at the early stages but has gradually reduced towards the recent years due to the spell of dimensionality issue. The spell of dimensionality problem is that the data develops in high dimensional space strewn and lowers the performance of the classifier algorithms that have been designed for working in low-dimensional space. Additionally, the handling of high dimensional features also requires

significantly increased resources for computations and memory storage that are additional burdens to any processing system [2]. The classification algorithms depend on the features utilized and hence the use of selected features of significant importance is essential to achieve higher performance of the classifier with less complexity. Feature selection is a kind of dimension reduction strategy to process the high dimensional data.It selects the best set of appropriate and important features and excludes the redundant and features that have no relationship to enhance the classifier efficiency.

Feature selection algorithms are often modelled as combinatorial optimization and the selection process depends on different properties or factors of those features. The methods of feature selection are categorized into three main categories: filter, wrapper, and embedded methods [3]. The filter-based methods are performed by selecting the feature subsets based on the data itself such as information gain, distance, feature dependency, etc. The wrapper-based methods utilize some learning algorithms (classifiers) to evaluate the properties of the features and select the best feature subset using a search process. Wrapper methods are efficient by the means of determining the classifier, feature assessment measures and the searching process. In comparison, the filter methods perform faster and are computationally cheaper. Moreover, the wrapper-based methods are most utilized methods due to their greater classifier accuracy predetermined by the learning algorithms. Some studies have utilized embedded based methods, which combine the benefits of both the filter and wrapper methods. Most recent studies mainly focused on employing the optimization based algorithms for feature selection because of their superior search capabilities and their influence in improving the classifier accuracy without increasing the computation complexity [4]. The optimization based feature selection algorithms belong mostly to the wrapper method and some are categorized under the embedded method.

This paper aims to develop an optimal feature selection method by analysing the limitations of the existing feature selection algorithms. It has been accepted by the researchers that the “no free lunch (NFL)” theorem [5] is true in the sense that no single optimization based algorithm can address all the optimization issues. Similarly, a single optimization algorithm cannot overcome the feature selection problem in all types of datasets and hence specific and advanced optimization algorithms are required for each type of data problems. Through extensive analysis, it has been found that the searching process in any optimization algorithm is the main factor that influences its entire performance. Based on this analysis results, the Squirrel search algorithm (SSA) has been a robust, effective and high performance through their gliding based search process [6]. Hence this algorithm has been selected and the possibility to enhance its performance was studied. As a result, the chaos theory based population initialization and acceleration coefficients and inertia weights are used to modify the search equations. The main features of this work for the proposed PGS-ISSA are:

- Introducing Chaos theory to improve population initialization for expanding the solution search space.
- The addition of acceleration coefficients in the position-update equations to improve the convergence rate in local search process.
- Usage of inertia weight in position-update process to increase the balance between the exploration and exploitation.
- The fitness function is calculated based on the minimum error rate for ensuring the selection of best features that improve the classification accuracy.
- The PGS-ISSA was evaluated and compared against existing algorithms using six different benchmark datasets from the UCI repository.

This PGS-ISSA has outperformed the standard SSA and other optimization algorithms in terms of feature selection. Rest of the paper is structured as following: Related works in 2nd section. 3th Section includes the proposed PGS-ISSA approach overview and 4th section contains evaluation results. 5th Section ends with insights into future study.

2. Related work

A Many existing optimization based algorithms have been successfully utilized in the feature section methods. However, in search of suitable algorithms for feature selection in different types of datasets, the utility of each optimization algorithm has been tested continuously. and new algorithms are developed for feature selection. Genetic Algorithm (GA) [7], Particle Swarm Optimization (PSO) [8], Artificial Bee Colony (ABC) [9], Ant Colony Optimization (ACO) [10] and Firefly Algorithm (FA) [11] are some of the common algorithms used in feature selection methods. However, these traditional algorithms do have drawbacks in algorithm convergence rate in view of their limited searching abilities. While hybrid techniques using these optimization models and other statistical algorithms provide better performance for certain problems, the

higher complexity of those models yield critical questions.

Recent years have seen the tremendous growth in number of advanced optimization algorithms and their subsequent utilization in feature selection methods. Gunavathi and Premalatha [12] proposed a feature selection approach using Cuckoo search algorithm (CSA) for cancer datasets. This approach provided 100% classification accuracy with the K-nearest neighbor classifier. However, this approach has limitations in the random walk-based search process of the CSO. Hafez et al. [13] proposed a feature selection approach using the Chicken swarm optimization (CSO) and provided less error rate of 11.52%. However, this CSO based feature selection method has less convergence which degrades the classification of larger datasets. Emary et al. [14] used Gray-wolf optimization (GWO) for feature subset selection and achieved higher accuracy of 97.88%. However, the GWO method has limited global search capacity which is a serious issue when handling larger data.

Zawbaa et al. [15] developed feature selection approach using moth-flame optimization (MFO) and produced reduced classification error of 2.26%. However, this method has higher computation time for larger datasets. Ibrahim et al. [16] employed the Salp swarm algorithm for feature selection (SSA-FS) which produced accuracy of 99% at less time. But this approach has poor performance for big datasets due to the high computation complexity. Mafarja and Mirjalili [17] introduced Whale optimization algorithm (WOA) based wrapper feature selection approach to classify data with 98.2% accuracy. But this approach provided high accuracy only through the utilization of simple crossover and mutations, which makes the WOA achieving low performance individually. Manikandan and Kalpana [18] designed artificial fish swarm optimization (AFSO) based feature selection with Classification and regression tree (CART) to achieve 7.9% improved accuracy. Still this algorithm has limited convergence rate for larger datasets.

The SSA has proven to be a high-performance solution for complex optimization problems and engineering based applications. [6]. It has been found that the gliding property in SSA has provided better solution selection. Concerning this performance, this paper has selected SSA for the feature selection problem. However, the random summer search method of SSA decreases the convergence speed, and the convergence precision is also reduced due to less effective global and local search methods. Hence an improved SSA has been planned to be introduced in this paper for efficient feature selection. Some studies have already developed such improved SSA models. Wang and Du [19] presented an improved SSA (ISSA) using progressive search and jumping search methods. The jumping search uses the escape operation and death operation to intensify the search space and exploration utility. The progressive search uses mutation process to intensify the exploration. This ISSA achieves high accuracy, better convergence speed and robustness. But this ISSA uses high resources for the optimization process. In another work, Wang and Du [20] presented Multi-objective Evolutionary Algorithm using Decomposition with External Population and Adaptive Weight Vectors Adjustment ISSA (MOEA/D-EWA-ISSA). This improved SSA has employed the evolutionary method with externally developed population and adaptive weights for the jumping search and progressive search methods. These advancements increased the precision of SSA but this method needs complex optimization for multi-objective problems. Sanaj and Prathap, [21] developed chaotic squirrel search algorithm (CSSA) and applied it for cloud-based task scheduling. This approach utilizes the ISSA of Wang and Du [19] with an additional chaotic process. This CSSA model achieved 26% increased convergence speed, 92%resource utilization and 98.8% accuracy. However, this CSSA has high exploration time that leads to high time consumption for larger tasks.

Zheng and Luo [22] developed an ISSA using four modifications to the SSA. An adaptive method of predator presence probability, normal cloud model for randomness and fuzziness, a selection strategy between successive positions, and a dimensional search enhancement strategy are applied to SSA to form the suggested ISSA. This ISSA achieved 22% higher optimum solutions and less error of 0.06. However, this model has high complexity when applied for multi-objective problems. El-Ashmawi and Abd Elminaam [23] developed a modified squirrel search algorithm (MSSA) using improved best-fit heuristic for producing a feasible initial solution while the operator method embedded in the updating stage to enhance the exploration capabilities of SSA. This MSSA achieved faster convergence with 2% to 9% reduction in execution time. However, it returned less performance against PSO and ABC for the bin-packing problem which might become disadvantage for similar optimization problems. Wang et al. [24] designed an ISSA using spatial variation and diffuse inspired by the invasive weed optimization (IWO) algorithm. This algorithm achieved RMSE values of 1.39 and also increased the accuracy with minimized time consumption of 86 seconds. However, the complexity of this ISSA

negatively impacts the performance for high resolution applications. Similar to this algorithm, Zhang et al. [25] proposed an improved SSA called Reproductive SSA (RSSA) in which the reproductive behavior of the invasive weed algorithm (IWO) and an adaptive step strategy are used to generate new population and enhance global search process to balance the exploration and exploitation. This RSSA achieved high convergence with 33.3% high optimum solutions, high accuracy and less error of 0.08. Although the global searching of RSSA is efficient, the local search performance is still limited.

The literature studies showed that the above discussed optimization based feature selection algorithm have improved the classification performance significantly than their preceding techniques. However, based on NFL theorem, these methods also suffer from some prominent drawbacks. The most important of those drawbacks are the computation complexity, slow convergence and stacked local optimum due to limited global search process. This research paper has considered these drawbacks and analysed various new optimization algorithms. Considering these drawbacks, the planned ISSA model for feature selection is designed to tackle most of the issues in existing models. In this paper, some modifications are done to improve its performance to adapt PGS-ISSA for the feature selection in big data classification problems. The proposed PGS-ISSA is developed and compared against to the existing SSA, ISSA and other optimization based algorithms to evaluate its performance.

3. Methodology

The proposed methodology consists of the general data classification process. First, the datasets are pre-processed and then the feature selection is performed using the PGS-ISSA. Finally, the classification is performed and the performance of the classifiers are analysed. The proposed PGS-ISSA is the improved version of the standard SSA and hence the SSA is first studied. Table 1. illustrates the list of notations used in this paper.

3.1 Squirrel search algorithm

SSA is based on the Inflight squirrels foraging for their food during the hot and cold seasons. During the

Table 1. Notations and definitions

Notation	Definition
NP	Initial population of squirrels
t_{max}	maximum iterations
n	decision variables
P_{dp}	predator presence probability
sf	scaling factor
G_c	gliding constant
FS_U	upper bound for decision variable
FS_L	lower bounds for decision variable
$rand()$	evenly distributed random number
$f = (f_1, f_2, \dots, f_{NP})$	fitness rate
$i = 1,2, \dots, NP$	individual squirrels
FS_{ht}	location of Inflight squirrel on hickory nut trees
FS_{at}	location of Inflight squirrel on acorn nuts trees
FS_{nt}	location of Inflight squirrel on normal trees
d_g	random gliding distance
R_1, R_2, R_3	random numbers between $[0, 1]$
L	Lift force
D	Drag force
R	resultant force
Mg	magnitude of equilibrium glide
V	Velocity
ϕ	angle of gliding
ρ	density of air
C_L	lift coefficient
S	surface area of body
C_D	frictional drag coefficient
h_g	loss in height
S_c	seasonal constant
S_{cmin}	Minimum seasonal constant
S_c^t	smallest initial seasonal value
$Levy$ ()	Levy function
r_a, r_b	uniform distribution functions on the interval [0, 1]
β	Constant
σ	Variance
x_i	i-th parameter of the chaotic squirrel
μ	bifurcation factor
$\alpha_1, \alpha_2, \alpha_3$	acceleration coefficients
ω	inertia weight

ω_{max} *and* ω_{min}	maximum and minimum values of the inertia weight

hot seasons, the squirrels are active in food searching and by gliding between the trees, they move to new locations to explore the environment. Due to the hot climate, their energy will be drained quickly and hence they consume the first seen abundance available acorns immediately without finding the optimal food source. After achieving their energy needs, the squirrels search for optimal hickory nuts source and store them for usage in winter. As foraging for food in winter is highly dangerous and costly for the squirrels, they use the stored nuts to maintain their energy. They stay less active but without hibernation during winters and become active when the cold season ends. This iterative process is performed by the squirrels during their entire lifetime. The mathematical modelling of these behaviours forms the SSA optimization and it has the following assumptions [6].

In a forest, the number of flight squirrels is considered as an NP (…) and each tree is searched by only one squirrel.

Three types of trees exist in the forest: regular trees, hickory trees (source of hickory nut) and oak nut trees (source of acorn nut).

Individual nutrition searching is assumed and optimal food sources are explored dynamically. The SSA process starts with the initializing the population positions in the D dimensional searching space.

Parameters initialization: The main parameters of SSA are maximum iterations t_{max}, population NP, decision variables n, attacker presence possibility P_{dp}, the scaling factor sf, gliding constant G_c, the upper and the lower bounds of decision variables FS_U and FS_L. All the above-mentioned parameters should be initialized in the initial stage of SSA process.

Initialization of Inflight Squirrels' Positions and Fitness evaluation: the positions of inflight Squirrel are arbitrarily initialized in the searching space as shown below:

$$FS_{i,j} = FS_L + rand() \times (FS_U - FS_L); \quad i = 1,2,3,\dots,NP, j = 1,2,3,\dots n \tag{1}$$

Where, $rand()$ is a number which is distributed equally and randomly within [0, 1].

The fitness rate $f = (f_1, f_2, \dots, f_{NP})$ of a specific inflight squirrel position is computed by substituting the decision variable value for the fitness function:

$$f_i = f_i\left(FS_{i,1}, FS_{i,2}, \dots., FS_{i,n}\right) i = 1,2,\dots,N \tag{2}$$

Then the superiority of nutrition sources distinct by the value of fitness for inflight squirrels' positions is arranged by ascending order:

$$\left[sorted_f, sorte_{index}\right] sort(f) \tag{3}$$

After the process of sorting the nutrition (food) sources of each inflight squirrel's position, three kind of trees are marked: hickory tree (hickory nuts source), oak tree (acorn nuts source), and regular tree. position of the best nutrition source (i.e., minimal fitness value) is considered as the hickory nut tree (FS_{ht}), positions of the next three nutrition sources are marked to be the oak nuts trees (FS_{at}), and the remaining are deliberated as regular tree (FS_{nt}):

$$FS_{ht} = FS(sorte\_index(1)) \tag{4}$$

$$FS_{at}(1:3) = FS(sorte\_index(2:4)) \tag{5}$$

$$FS_{nt}(1:NP-4) = FS\left(sorte_{index(5:NP)}\right) \tag{6}$$

Create new locations based on Gliding: After the Inflight squirrels' effective gliding course, one of three cases can happen It is presumed in each case that Inflight squirrel searches effectively during glid and all over the forest for its desired food in the absence of the predator, while predator Compel it to move walk randomly to hunt for a neighbouring hiding spot. these cases are:

Case 1: Inflight squirrels that move to the hickory nut trees from oak nut trees ('FS' at). The new location is given as:

$$FS_{at}^{t+1} = \begin{cases} FS_{at}^{t} + d_g \times G_c \times (FS_{ht}^{t} - FS_{at}^{t}) \\ if\ R_1 \geq P_{dp} \\ Random\ location\ otherwise \end{cases} \tag{7}$$

Where, d_g a is random gliding distance, R_1 is a random number, $R_1 \in [0, 1]$, FS_{ht} is the location of Inflight squirrel which gets tree of hickory nut and the current iteration represents by t. Balancing the exploration and exploitation achieved by gliding constant G_c in the model of mathematical.

Its value greatly impacts the efficiency of the proposed approach which is fixed as 1.9, obtained from the analysis.bw

Case 2: The Inflight squirrels glid from regular trees (FS_{nt}) to oak nut trees to fulfil their everyday energy requirements. Squirrels are given this new position as:

$$FS_{nt}^{t+1} = \begin{cases} FS_{nt}^{t} + d_g \times G_c \times (FS_{at}^{t} - FS_{nt}^{t}) \; if \; R_2 \geq P_{dp} \\ Random \; location \; otherwise \end{cases} \quad (8)$$

Where, R_2 represent a random number between [0,1].

Case 3: Certain squirrels that are on regular trees and oak nuts that have already been consumed travel to the trees of the hickory nut to stockpile hickory nuts in order to prevent winter food shortages. Squirrels new position is given as:

$$FS_{nt}^{t+1} = \begin{cases} FS_{nt}^{t} + d_g \times G_c \times (FS_{ht}^{t} - FS_{nt}^{t}) \; if \; R_3 \geq P_{dp} \\ Random \; location \; otherwise \end{cases} \quad (9)$$

Predator presence probability P_{dp} is considered to be 0.1 in all these three cases.

Where R_3 represent a random number between 0 and 1.

In all these three instances, Attacker existence probability P_{dp} is found to be 0.1.

The Gliding mechanism of Inflight squirrels' is represented by balance glide in which from (L) the sum of the force of lift and (D) drag (R) produces as a resulting force which has magnitude is equal and opposite the direction of weight of the of the inflight squirrel (Mg). R hence gives a linear gliding path at constant velocity to the Inflight Squirrel (V).

Always, gliding of Inflight squirrel at fixed speed decreased at angle ϕ to horizontal and glide ratio or lift-to-drag ratio, described as

$$\frac{L}{D} = 1/\tan\phi \quad (10)$$

The length of their glide path increases if Inflight squirrels making the glide angles narrower (ϕ) and hence the ratio of the lift-to-drag be higher. Here, the product of a downward deflection of the air moving through the wings is the lift. we can define it as:

$$L = 1/2\rho C_L V^2 S \quad (11)$$

Where the air density ρ (=1,204 kgm^{-3}), CL is considered as the coefficient of lift, V (=5.25 ms^{-1}) is the speed, and S (=154 cm^2) is the body surface area. The friction drag is described as:

$$D = 1/2\rho C_D V^2 S \quad (12)$$

Where, C_D s the coefficient of frictional drag, at high speed, the drag part is small and, at low speed, it gets higher. Therefore, the angle of glide at a constant state is set as:

$$\phi = \arctan\left(\frac{D}{L}\right) \quad (13)$$

From these parameters, the approximated gliding distance (d_g) is calculated based on these parameters:

$$d_g = \left(\frac{h_g}{\tan\phi}\right) \quad (14)$$

Where, h_g (=8 m) is the biggest loss that happened after gliding. It is observed that sf = 18 offers an adequate disruption spectrum of d_g in the range [0.5, 1.11].

Seasonal monitoring condition: The food searching process of inflight squirrels is dependent on the climate changes in different seasons. Hence, a seasonal monitoring constraint is presented to avoid the system being stuck in the optimal local solutions. A constant of seasonal S_C and its smallest value is estimated initially as:

$$S_c^t = \sqrt{\sum_{k=1}^{n}\left(FS_{at,k}^{t} - FS_{ht,k}\right)^2}, t = 1,2,3 \quad (15)$$

$$S_{cmin} = \frac{10E^{-6}}{365^{t/\left(\frac{t_{max}}{2.5}\right)}} \quad (16)$$

Where, t_{max} and t respectively are the max iterations and the current values. The value S_{cmin} disturbs the exploration and exploitation processes of the proposed approach. The Larger value of S_{cmin} encourages exploration and the smaller values of S_{cmin} improve algorithm exploitation capability.

Then the seasonal monitoring situation is tested. When $S_c^t < S_{cmin}$, the winter season has ended, and the Inflight squirrels that previously lost their proficiency to travel the jungle in a random way reposition their searching locations for nutrition source again.

The Random repositioning at the end of winter season: When surviving squirrels are at the end of the winter season, they become active and search for food in new directions. The reposition of such

Inflight squirrels is demonstrated through the next equation:

$$FS_{nt}^{new} = FS_L + Levy\,(n) \times (FS_U - FS_L) \quad (17)$$

where Levy function enhances the global exploration proficiency, which is given by

$$Levy\,(x) = 0.01 \times \frac{r_a \times \sigma}{|r_b|^{1/\beta}} \quad (18)$$

The two uniform functions r_a and r_b are present at the interval [0, 1], the constant β is selected as β = 1.5 and the estimate of the following σ is:

$$\sigma = \left(\frac{\Gamma(1+\beta) \times \sin(\pi\beta/2)}{\Gamma((1+\beta)/2) \times \beta \times 2^{((\beta-1)/2)}}\right)^{1/\beta} \quad (19)$$

where $\Gamma(x) = (x-1)!$

Stopping Criteria: The algorithm stops when iterations are maximum. Until then, the seasons are monitored and the new location updating is repeated.

Algorithm 1: Pseudo code for SSA:

Begin

Set parameters for input, t=0

Randomly generate positions for the n number of Inflight squirrels by using Eq. (1)

For all Inflight squirrel position get the fitness using Eq. (2)

based on fitness rating positions of Inflight squirrels, Sort them in ascending order.

Declare Inflight squirrels which on tree of hickory nut, oak nut trees, or regular trees.

In random way pick some of the Inflight squirrels that are on regular trees to move towards the hickory nut tree, and the remaining squirrels one moves towards the oak nut trees after them.

While ($t < t_{max}$) For t = 1 to n1 (where n1 is all Inflight squirrels on oak trees and they glid towards the hickory tree)

Update location using Eq. (7)

End for

For t = 1 to n2 (where n2 is all Inflight squirrels on regular trees and travel

towards oak trees)

Update location using Eq. (8)

End for

For t = 1 to n3 (where n3 is all Inflight squirrels which are on regular trees and travel towards a tree of hickory nuts)

Update location using Eq. (9)

End for

compute the seasonal constant (S_c

If (Seasonal monitoring condition is satisfied)

Randomly move inflight squirrels using Eq. (17)

End if

Update the min value of the seasonal constant (S_{cmin}) by using Eq. (16)

$t = t + 1$

End

The position of the squirrel on the hickory nut tree is the ultimate ideal solution.

End

3.2 Improved Squirrel Search Algorithm for Feature Selection

In the proposed PGS-ISSA, two important modifications are suggested to the standard SSA algorithm to overcome the pre-mature convergence and also to improve the exploration to avoid the local optima problem. The first modification is improving the initialization process using chaotic theory. The second modification is the improvement of the new location generation using acceleration coefficients and inertia weight. First, the algorithm is initialized with the parameters as in SSA. Then the locations are initialized by the chaos theory.

Initialize Inflight Squirrels' Locations: The initialization phase is modified by using the chaos theory to initialize the population location instead of the random initialization. In standard SSA, the initial population are selected in a random manner. This may fail to select most promising individuals which influence best solution space. The randomness in selecting populations cannot guarantee the uniform distribution. And it also results in earlier convergence and suffers from local optima. To overcome this problem, this work introduced chaos mapping which handles the complex nature of SSA with its unpredictable behaviour and mapping strategy. It is given as

$$x_{i+1} = \begin{cases} 4\mu x_i(0.5 - x_i), & 0 \le x_i < 0.5 \\ 1 - 4\mu x_i(0.5 - x_i)(1 - x_i), & 0.5 \le x_i \le 1 \end{cases} \quad (20)$$

Where x_i is the i-th parameter of the chaotic squirrel from the population at k iterations and μ is the bifurcation factor whose value is $3.5699 \leq \mu \leq 4$. in most cases, it is chosen as $\mu = 4$ for simple and effective initialization. By using this chaotic system, it provides possibility of improving uniform distribution. It also avoids earlier convergence by improving the diversity of squirrel's population and optimal search space by reaching global optimization which avoids earlier convergence of individuals.

The Inflight Squirrels are initialized in the search area based on this feature as follows:

$$FS_{i,j} = FS_L + x_{i+1} \times (FS_U - FS_L); \quad i = 1,2,\ldots.,NP, j = 1,2,\ldots n \tag{21}$$

In the early stage of search, the distance to the acorn tree is typically too big to contribute to an efficient gliding, which results in random partial and blindfold searching certainty of the movements of squirrels. The squirrels are chosen to be actualized using the acceleration coefficient α in order to solve the random search of individuals. Also, if the location of i-th squirrel is excellent, the quest is less random. hence with increasing the iteration number, the effect of randomness should be decreased. To regulate this effect, an inertia weight ω is added.

By using the modified equations, we can mathematically model the dynamic foraging behaviour:

Case 1:

$$FS_{at}^{t+1} = \begin{cases} FS_{at}^{t} + d_g \times G_c \times \alpha_1 . \omega (FS_{ht}^{t} - FS_{at}^{t}) \\ \quad if\ R_1 \geq P_{dp} \\ Random\ location\ otherwise \end{cases} \tag{22}$$

Case 2:

$$FS_{nt}^{t+1} = \begin{cases} FS_{nt}^{t} + d_g \times G_c \times \alpha_2 . \omega (FS_{at}^{t} - FS_{nt}^{t}) \\ \quad if\ R_2 \geq P_{dp} \\ Random\ location\ otherwise \end{cases} \tag{23}$$

Case 3:

$$FS_{nt}^{t+1} = \begin{cases} FS_{nt}^{t} + d_g \times G_c \times \alpha_3 . \omega (FS_{ht}^{t} - FS_{nt}^{t}) \\ \quad if\ R_3 \geq P_{dp} \\ Random\ location\ otherwise \end{cases} \tag{24}$$

where $(\alpha_1, \alpha_2, \alpha_3) \in [0,1]$ are acceleration coefficients such that their optimal values lie between 0.2 and 0.8. In most optimal cases, their value is around 0.5. Likewise ω is the inertia weight calculated by

$$\omega = \omega_{max} - \frac{t}{t_{max}} \times (\omega_{max} - \omega_{min}) \tag{25}$$

where, ω_{max} and $\omega_{min}$are respectively maximum and minimum of the inertia weight. The current and maximum numbers of the iteration are t and t_{max} .

The value of ω is determined between $\omega \in [0,1]$ but most probably the suitable values are between [0.2, 0.8]. This means ω_{max} and ω_{min} values are 0.8 and 0.2 respectively. This enhances the local search process and enhances the convergence rate and speed.

Seasonal monitoring and stopping criteria: the season monitoring and the subsequent relocations are performed as in the SSA. Finally, the algorithm terminates at maximum iterations.

Algorithm 2: Pseudo code for Proposed PGS-ISSA:

Begin

Set input parameters

Set iteration of t=0

Generate chaotic positions for a variety of Inflight squirrels by using Eq. (21) (a)

For all Inflight squirrel, position evaluate the fitness

based on fitness rating positions of Inflight squirrels, Sort them in ascending order.

Declare Inflight squirrels which on tree of hickory nut, oak nut trees, or regular trees.

In random way pick some of the Inflight squirrels that are on regular trees to move towards the hickory nut tree, and the remaining squirrels one moves towards the oak nut trees after them.

While ($t < t_{max}$)
For t = 1 to n1
Update location using Eq. (22)
End for
For t = 1 to n2
Update location using Eq. (23)
End for
For t = 1 to n3
Update location using Eq. (24)
End for
Compute the seasonal constant (S_c)
If (Seasonal monitoring condition is satisfied)
Randomly reposition Inflight squirrels by using Eq. (17)
End if
Update S_{cmin} using Eq. (16)
$t = t + 1$;
End
Return optimal solution
End

Table 2. List of datasets used in experiments

Datasets	Number of attributes	Number of instances
Higgs	28	1100000
Glass	9	214
Diabetes	20	768
Cleveland	14	303
Vehicle	18	846
Wine	13	178

4. Experiments and results

4.1 Datasets

The implementation of the proposed PGS-ISSA is performed using MATLAB. To access the performance of the approach for feature selection, the experiments are conducted on 6 benchmark datasets available at the UCI repository. Among these datasets, the larger Higgs dataset is also used to achieve big data classification. The details of the datasets which we used for evaluation shows in Table 2.

4.2 Performance evaluation of PGS-ISSA based feature selection

The experiments are conducted by employing the proposed PGS-ISSA and the standard SSA for feature selection using Support vector machine (SVM) classifier. Similarly, the improved and modified SSA models from the literature are also implemented to compare their efficiency with the proposed PGS-ISSA. ISSA [19], ISSA [22], and RSSA [25] are the improved SSA algorithms chosen for implementation. MOEA/D-EWA-ISSA [20] and CSSA [21] are omitted since it is highly similar to ISSA [19] but has high complexity. MSSA [23] is provides low performance than PSO while ISSA [24] is much similar to the RSSA [25] but has high error rate and hence they are not considered for implementation.

From the below table and charts, it can be seen that the proposed PGS-ISSA has improved the feature selection process better than the SSA and other improved SSA based approaches. Comparing the SSA, ISSA [19], ISSA [22], RSSA [25] and the proposed PGS-ISSA applied to SVM classifier, it can be seen that, in most cases, PGS-ISSA based feature selection approach has increased accuracy, precision, recall, f-measure, specificity and reduce execution time. In particular, for the bigger Higgs dataset, PGS-ISSA has high accuracy of 64.72% which is 5.39%, 3.49%, 3.2%, and 1.78%, higher than the SSA, ISSA [19], ISSA [22], and RSSA [25], respectively. PGS-ISSA achieved precision of 67.3194% which is 1.65%, 2%, 0.37%, and 0.74%, higher than the SSA, ISSA [19], ISSA [22], and RSSA [25], respectively. PGS-ISSA achieved recall of 62.1528% which is 4.07%, 2.26%, 1.08%, and 0.3%, higher than the SSA, ISSA [19], ISSA [22], and RSSA [25], respectively. It also achieved higher f-measure and specificity values than the compared algorithms. PGS-ISSA has consumed less time of 140.1366 seconds which is 24.9%, 20%, 16.4%, and 11% lesser time spent by SSA, ISSA [19], ISSA [22], and RSSA [25], respectively.

Likewise, PGS-ISSA has achieved better performance for the other datasets too. In some cases, other methods have slightly gained advantages in performance results. RSSA [25] has achieved high accuracy and f-measure for Wine data than the proposed PGS-ISSA. It has also achieved equal values of precision for wine data, high specificity for Cleveland data, less time for vehicle data compared to PGS-ISSA. RSSA [25] and ISSA [22] also achieved precision, recall, f-measure and specificity values equal to PGS-ISSA for glass dataset. Apart from these instances, PGS-ISSA has better values for all other instances. This indicates that the PGS-ISSA

Table 3. Performance comparison of PGS-ISSA and others ISSA in feature selection

Parameter	Dataset	SSA [6]	ISSA [19]	ISSA [22]	RSSA [25]	Proposed PGS-ISSA
Accuracy (%)	Higgs	59.3305	61.2322	61.5294	62.94	**64.72**
	Glass	98.1008	93.7711	95.435	96.5067	**98.4375**
	Diabetes	90.2322	89.0543	90.0418	91.1014	**91.7391**
	Cleveland	70.8767	71.00	70.8242	71.1100	**72.2088**
	Vehicle	85.7684	86.2020	86.5512	87.1667	**88.1890**
	Wine	93.9871	92.1991	94.0778	**95.4717**	94.3396
Precision (%)	Higgs	65.6667	65.3192	66.9457	66.5775	**67.3194**
	Glass	99.9950	99.8900	**100**	**100**	**100**
	Diabetes	88.7656	88.8780	89.0380	89.05	**90.2299**
	Cleveland	70.7656	72.1145	71.4815	72.857	**74.6032**
	Vehicle	92.9871	93.5632	93.0588	94.233	**95.3125**
	Wine	86.75	87.9999	87.05	**88**	**88**
Recall (%)	Higgs	58.087	59.8965	61.0751	61.8588	**62.1528**
	Glass	99.2234	99.8762	**100**	**100**	**100**
	Diabetes	94.6519	94.8214	95.1840	95.40	**96.3190**
	Cleveland	93.7650	93.2382	93.7959	94.5567	**95.9184**
	Vehicle	90.8780	90.7778	89.5075	90.505	**91.0448**
	Wine	98.755	95.0033	98.9545	99.0459	**100**
F-measure (%)	Higgs	58.2132	60.0050	60.4774	61.7889	**62.3026**
	Glass	99.9934	99.7550	**100**	**100**	**100**
	Diabetes	91.9872	92.2134	92.5888	93.005	**93.1751**
	Cleveland	80.2301	81.2379	81.6369	82.143	**83.9286**
	Vehicle	91.1991	91.8554	91.7778	92.5	**93.1298**
	Wine	92.1989	93.2601	93.5000	**93.85**	93.6170
Specificity (%)	Higgs	81.2017	80.6667	81.4775	82.1567	**82.7226**
	Glass	**100**	98.9803	**100**	**100**	**100**
	Diabetes	71.7171	71.8765	72.5821	73.2661	**74.6269**
	Cleveland	58.8723	59.8137	61.1905	**62.1033**	61.9048
	Vehicle	96.9876	97.7778	**98.9305**	97.9305	98.3957
	Wine	88.9765	89	89.667	90.012	**90.3226**
Time (seconds)	Higgs	186.67	175.265	167.645	157.47	**140.1366**
	Glass	3.8764	3.8192	3.3376	3.1191	**3.0295**
	Diabetes	15.1287	15.0976	14.794	14.4057	**13.4296**
	Cleveland	7.6565	7.3811	6.99	6.4119	**6.3030**
	Vehicle	12.0987	12.4567	11.667	**10.1786**	10.9835
	Wine	3.2269	3.3490	2.9987	2.7362	**2.7247**

has much better efficiency in selection of features in these datasets.

4.3 Comparative analysis of PGS-ISSA with other optimization-based feature selection algorithms

The performance of the PGS-ISSA feature selection method is also evaluated and compared with the other optimization-based methods described in literature. All evaluations were made over the Higgs data using the SVM classifier. The accuracy and execution time results of some methods in feature selection are listed in Table 4.

From Table 4 and the above chart, it is found that the PGS-ISSA based feature selection approach has better performance over other methods for the classification of the Higgs data. It has achieved accuracy of 64.72% and consumed 140.1366 seconds for execution, which are much better than the other compared methods including the SSA and improved SSA models. The proposed PGS-ISSA has achieved accuracy improvement of 1% to 15% than the other compared algorithms. It has also consumed less time which is reduced about 17 seconds to 150 seconds compared with the other optimization based feature selection algorithms. Finally, it can be concluded that the suggested PGS-ISSA has performed well in terms of efficiency for feature selection in classification of big data.

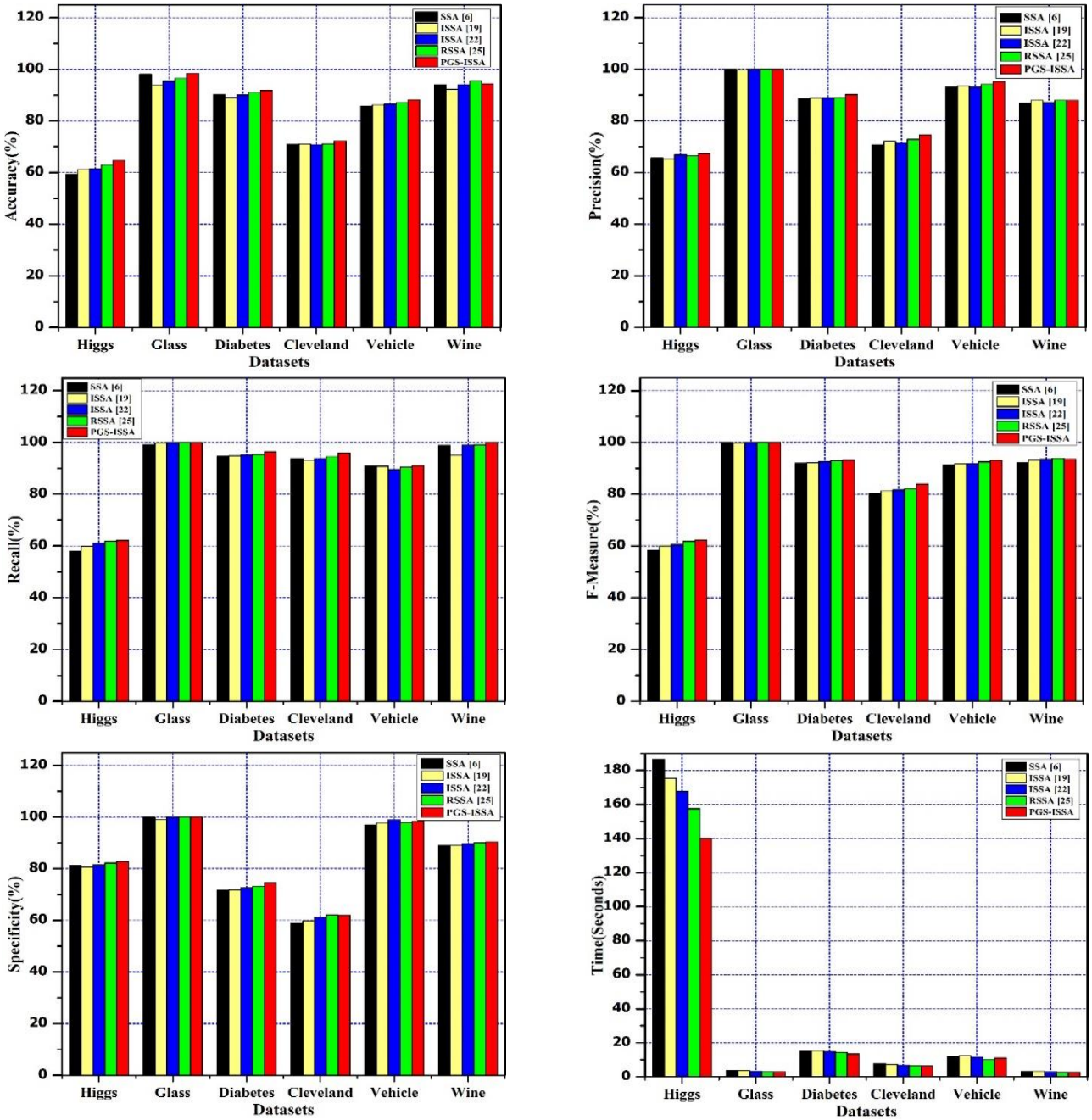

Figure. 1 Performance comparison of PGS-ISSA and others ISSA in feature selection

Table 4 PGS-ISSA Vs. other optimization-based feature selection methods

Approach	**Accuracy (%)**	**Time (seconds)**
GA [7]	49.808	298.45
PSO [8]	51.11	286.23
ABC [9]	50.85	293.99
ACO [10]	51.67	290.33
FA [11]	52.34	286.68
GA-PSO [12]	56.10	262.10
CSA [13]	57.88	253.69
CSO [14]	58.17	231.71
GWO [15]	57.89	245.87
MFO [16]	54.60	225.25
WOA [17]	56.67	189.56
AFSO [18]	55.45	199.42
SSA [6]	59.33	186.67
ISSA [19]	61.2322	175.265
ISSA [22]	61.5294	167.645
RSSA [25]	62.94	157.47
PGS-ISSA	**64.72**	**140.1366**

5. Conclusion

Feature selection problem in big data classification is one of the most vital issues considering the unique characteristics of big data analytics. In this article, an efficient feature selection method is suggested using population and global search improved SSA algorithm. This PGS-ISSA feature selection algorithm has provided improved classification performance better than the SSA. Experimental were conducted based on datasets from UCI repository to evaluate the performance of PGS-ISSA using three classifiers. Results indicated that the proposed PGS-ISSA achieved high performance for the bigger Higgs dataset than the standard SSA, existing ISSA models and other prominent optimization-based feature selection algorithms with 64.72% accuracy, 67.3194% precision, 62.1528% recall, 62.3026% f-measure, 82.7226% specificity and consumed less time of 140.1366 seconds. It achieved improvement of 1% to 6% on accuracy, 0.7% to 2% on precision, 0.3% to 4% on recall, 0.5% to 4% on f-measure, 0.5% to 2% on specificity while 11% to 25% reduction in execution time when compared with existing SSA, ISSA [19], ISSA [22], and RSSA [25] based future selection algorithms. Similarly, the proposed PGS-ISSA achieved better performance for the Glass, Diabetes, Cleveland, Vehicle and Wine datasets. In future, the possibility of increasing the accuracy of classification using advanced classifiers and parallel computing algorithms will be examined. Also, the likelihood of using the PGS-ISSA for other larger dataset-based applications will be also be investigated.

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

This work is a contribution of both the authors: Conceptualization, Issa Mohammed Saeed Ali, and Mukunthan Balakrishnan; methodology, Issa Mohammed Saeed Ali; software, Issa Mohammed Saeed Ali; validation, Issa Mohammed Saeed Ali, and Mukunthan Balakrishnan; formal analysis, Issa Mohammed Saeed Ali; writing original draft preparation, Issa Mohammed Saeed Ali; writing review and editing, Issa Mohammed Saeed Ali, and Mukunthan Balakrishnan

References

[1] A. Katal, M. Wazid, and R. H. Goudar, "Big data: Issues, challenges, tools and Good practices", In: *Proc. of 2013 Sixth International Conference on Contemporary Computing (IC3)*, pp. 404-409, 2013.

[2] V. B. Canedo, N. S. Maroño, and A. A. Betanzos, "Recent advances and emerging challenges of feature selection in the context of big data", *Knowledge-Based Syst.*, vol. 86, pp. 33-45, 2015.

[3] M. Rong, D. Gong, and X. Gao, "Feature Selection and Its Use in Big Data: Challenges, Methods, and Trends", *IEEE Access*, Vol. 7, pp. 19709-19725, 2019.

[4] L. Brezočnik, I. Fister, and V. Podgorelec, "Swarm Intelligence Algorithms for Feature Selection: A Review", *Appl. Sci.*, vol. 8, no. 9, p. 1521, 2018.

[5] X. S. Yang, "Swarm-Based Metaheuristic Algorithms and No-Free-Lunch Theorems", *Theory and New Applications of Swarm Intelligence*, No. May, InTech, 2012.

[6] M. Jain, V. Singh, and A. Rani, "A novel nature-inspired algorithm for optimization: Squirrel search algorithm", *Swarm Evol. Comput.*, Vol. 44, No. June 2017, pp. 148-175, 2019.

[7] S. S. Hong, W. Lee, and M. M. Han, "The feature selection method based on genetic algorithm for efficient of text clustering and text classification", *Int. J. Adv. Soft Comput. its Appl.*, Vol. 7, No. 1, pp. 22-40, 2015.

[8] L. M. Abualigah, A. T.Khader, and E. S. Hanandeh, "A new feature selection method to improve the document clustering using particle swarm optimization algorithm", *J. Comput. Sci.*, Vol. 25, pp. 456-466, 2018.

[9] M. Schiezaro and H. Pedrini, "Data feature selection based on Artificial Bee Colony algorithm", *EURASIP J. Image Video Process.*, Vol. 2013, No. 1, pp. 1-8, 2013.

[10] S. Tabakhi, P. Moradi, and F. Akhlaghian, "An unsupervised feature selection algorithm based on ant colony optimization", *Eng. Appl. Artif. Intell.*, Vol. 32, pp. 112-123, 2014.

[11] L. Zhang, K. Mistry, C. P. Lim, and S. C. Neoh, "Feature selection using firefly optimization for classification and regression models", *Decis. Support Syst.*, Vol. 106, pp. 64-85, 2018.

[12] C. Gunavathi and K. Premalatha, "Cuckoo search optimisation for feature selection in cancer classification: A new approach", *Int. J. Data Min. Bioinform.*, Vol. 13, No. 3, pp. 248-265, 2015.

[13] A. I. Hafez, H. M. Zawbaa, E. Emary, H. A. Mahmoud, and A. E. Hassanien, "An innovative approach for feature selection based on chicken swarm optimization", In: *Proc. of 2015 7th Int. Conf. Soft Comput. Pattern Recognition, SoCPaR 2015*, pp. 19-24, 2016.

[14] A. Abraham, P. Krömer, and V. Snášel, "Afro-European Conference for Industrial Advancement: Proceedings of the First International Afro-European Conference for Industrial Advancement AECIA 2014", *Adv. Intell. Syst. Comput.*, Vol. 334, pp. 1-13, 2015.

[15] H. M. Zawbaa, E. Emary, B. Parv, and M. Sharawi, "Feature selection approach based on moth-flame optimization algorithm", In: *Proc. of 2016 IEEE Congr. Evol. Comput.* CEC 2016, pp. 4612-4617, 2016.

[16] H. T. Ibrahim, W. J. Mazher, O. N. Ucan, and O. Bayat, "Feature Selection using Salp Swarm Algorithm for Real Biomedical Datasets", *IJCSNS Int. J. Comput. Sci. Netw. Secur.*, Vol. 17, No. 12, pp. 12-20, 2017.

[17] M. Mafarja and S. Mirjalili, "Whale optimization approaches for wrapper feature selection", *Appl. Soft Comput.*, Vol. 62, No. November, pp. 441-453, 2018.

[18] R. P. S. Manikandan and A. M. Kalpana, "Feature selection using fish swarm optimization in big data", *Cluster Comput.*, Vol. 22, No. 5, pp. 10825-10837, 2019.

[19] Y. Xiaobing, Y. Xianrui, and C. Hong, "An improved gravitational search algorithm for global optimization", *J. Intell. Fuzzy Syst.*, Vol. 37, No. 4, pp. 5039-5047, 2019.

[20] Y. Wang and T. Du, "A Multi-objective Improved Squirrel Search Algorithm based on Decomposition with External Population and Adaptive Weight Vectors Adjustment", *Phys. A Stat. Mech. its Appl.*, Vol. 542, No. 61501107, p. 123526, 2020.

[21] M. S. Sanaj and P. M. J. Prathap, "Nature inspired chaotic squirrel search algorithm (CSSA) for multi objective task scheduling in an IAAS cloud computing atmosphere", *Eng. Sci. Technol. an Int. J.*, Vol. 23, No. 4, pp. 891-902, 2020.

[22] Y. Wang and T. Du, "An Improved Squirrel Search Algorithm for Global Function Optimization", *Algorithms*, Vol. 12, No. 4, p. 80, 2019.

[23] W. H. El-Ashmawi and D. S. A. Elminaam, "A modified squirrel search algorithm based on improved best fit heuristic and operator strategy for bin packing problem", *Appl. Soft Comput. J.*, Vol. 82, p. 105565, 2019.

[24] P. Wang, Y. Kong, X. He, M. Zhang, and X. Tan, "An Improved Squirrel Search Algorithm for Maximum Likelihood DOA Estimation and Application for MEMS Vector Hydrophone Array", *IEEE Access*, Vol. 7, No. Ml, pp. 118343-118358, 2019.

[25] X. Zhang, K. Zhao, L. Wang, Y. Wang, and Y. Niu, "An Improved Squirrel Search Algorithm with Reproductive Behavior", *IEEE Access*, Vol. 8, pp. 101118-101132, 2020.