Power System Technology
ISSN:1000-3673

Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

Yellow Ground Squirrel Algorithm (YGSA): A Novel Metaheuristic Algorithm for Global Optimization

Hojjat Farrahi Farimani[1], *Davoud Bahrepour[2], Seyed Reza Kamel Tabbakh[2], Reza Ghaemi[3]

[1.] Department of Computer Engineering, Neyshabur Branch, Islamic Azad University, Neyshabur, Iran.

[2.] Department of Computer Engineering, Mashhad Branch, Islamic Azad University, Mashhad, Iran.

[3.] Department of Computer Engineering, Quchan Branch, Islamic Azad University, Quchan, Iran.

Corresponding author: bahrepour@mshdiau.ac.ir

Abstract: In recent years, numerous methods have emerged for optimizing problems inspired by nature. Each of these methods, according to their nature, has applications for solving specific problems with unique characteristics. Given the wide array of optimization problems across various fields, introducing novel algorithms with distinct behaviors can be both appealing and practical. In this article, a nature-inspired meta-heuristic optimization algorithm called Yellow Ground Squirrel Algorithm (YGSA) is introduced. The main idea behind YGSA is to simulate the natural behavior of a yellow ground squirrel in terms of pursuit and evasion from a farmer (the predator) and reaching the nest. The objective of the proposed algorithm is to enhance and attain a more equitable equilibrium between the fundamental stages of exploration and exploitation in contrast to preceding approaches. This balance arises from the fact that the yellow ground squirrel, during its escape, simultaneously tries to increase the distance from the farmer (the predator) and decrease the distance to its nest (burrow). To validate the proposed algorithm, 56 benchmark evaluation functions from various unimodal and multimodal types have been evaluated. The evaluation results are compared with the HBO, EBO-CMAR, FO, SCA, PSO, CS, GSA, MVO, and WSO algorithms. The optimization results for unimodal test functions demonstrate the high exploitation capability of YGSA in approaching the optimal solution, while the optimization results for multimodal test functions indicate the strong exploration ability of YGSA in finding the main optimal region in the search space.

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

Keywords: Optimization, Meta-Heuristic Algorithm, Nature-Inspired Strategy, Optimization Algorithm, Yellow Ground Squirrel Algorithm (YGSA)

1. Introduction

The use of advanced technologies in various scientific fields increases the complexity of problems. The proposed solutions to these problems are rapidly advancing with the motto of "maximum profit with minimum cost"[1]. Therefore, the use of optimization methods and algorithms in analysis and decision-making has gained significant attention. In general, optimization methods and algorithms can be classified into the following two main categories [1, 11]:

1. **Exact algorithms:** These algorithms are capable of finding the exact optimal solution, but they lack sufficient efficiency for hard optimization problems, and their execution time increases exponentially with the dimensions of the problems. Examples of these algorithms include Linear Programming (LP), Nonlinear Programming (NLP), and Dynamic Programming (DP).
2. **Approximation algorithms:** These algorithms can find good (near-optimal) solutions for hard optimization problems in a short time. These algorithms are further divided into three categories: heuristic algorithms, meta-heuristic algorithms, and hyper-heuristic algorithms. Heuristic algorithms face two main problems, which are getting stuck in local optima and premature convergence to these points. To overcome these issues, meta-heuristic algorithms have been developed, which provide solutions to escape from local optima and have the capability to be applied to a wide range of problems.

Since some optimization problems formulated in different sciences can be solved using optimization algorithms, the use and development of optimization models have received attention from engineers over the decades. The complexities and natures of problems in various sciences have led to the development and presentation of new algorithms to solve these types of problems. In general, approximate algorithms can be categorized into four overarching groups [1]:

A. **Solution-based and population-based algorithms:** Solution-based algorithms modify a solution during the search process (e.g., Tabu Search Algorithm (TSA) [2] and Simulated Annealing (SA) algorithm [3]), while population-based algorithms consider a population of solutions during the search process (e.g., evolutionary algorithms [4] such as Genetic

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

Algorithms (GAs) and Genetic Programming (GP), Ant Colony Optimization (ACO) [5], Artificial Bee Colony (ABC) optimization [6], Particle Swarm Optimization (PSO) [8], Firefly Algorithm (FA) [20], and Battle Royal Optimization (RBO) algorithm [9]).

B. **Nature-inspired and non-nature-inspired:** Many meta-heuristic algorithms are inspired by nature, while some others are not. The well-known nature-inspired algorithms include Grey Wolf Optimizer (GWO) [12], Firefly Algorithm (FA) [13, 20], Flower Pollination Algorithm (PFA) [14], Whale Optimization Algorithm (WOA) [15], Butterfly Optimization Algorithm (BOA) [16], and Emperor Penguin Colony (EPC) algorithm [17, 18].

C. **With memory and without memory:** Some meta-heuristic algorithms do not have memory, meaning that they do not utilize the information obtained during the search process (e.g., SA algorithm [3]). However, some meta-heuristic algorithms, such as TSA [2], utilize memory and store the obtained information during the search.

D. **Deterministic and probabilistic:** Deterministic meta-heuristic algorithms, like TSA [2], solve the problem using deterministic decisions. On the other hand, probabilistic meta-heuristic algorithms, such as SA algorithm [3], employ a set of probabilistic rules during the search.

All optimization algorithms possess specific operators and features that facilitate exploration in both local and global search spaces [11]. Once the operators within the problem domain have been identified, these algorithms initiate exploration and exploitation at various stages during their execution within the search space of the problem. The exploration phase entails a global search, while the exploitation phase involves a local search within the promising regions of the search space that were discovered during the exploration phase [19, 20]. Exploration is often referred to as random search as it involves the random selection of a solution or a set of solutions from the entire solution space. This randomness allows the algorithm to evade local optima and explore the global search space, although this is not always guaranteed. On the other hand, exploitation is known as neighborhood search since, during each iteration, it seeks the best solution within the vicinity of the current optimal solution that exhibits potential for improvement. Consequently, achieving a balance between these two factors is paramount in designing optimal optimization algorithms. Therefore, the proper control and adjustment of the algorithm's tuning parameters are of crucial importance in establishing such a balance, although they are not sufficient on their own [21]. An efficient optimizer must create a suitable equilibrium between exploration and exploitation; otherwise, there is a risk of becoming trapped in local optima and encountering issues related to premature convergence.

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

This paper introduces a novel meta-heuristic optimization algorithm named the YGSA, inspired by the escape behavior of yellow ground squirrels in both natural environments and agricultural fields when evading farmers. The behavior of the squirrel during escape serves as the basis for this algorithm, which aims to reduce the error rate and achieve better balance by controlling the exploration and exploitation phases jointly. During its escape, the yellow ground squirrel dynamically seeks to increase the distance from the farmer while simultaneously decreasing the distance from its nest. By modeling this escape behavior, the algorithm ensures a proper equilibrium between exploration, exploitation, and the ability to escape local optima. The strengths of the proposed algorithm can be summarized as follows:

- ✓ Optimal, dynamic, and simultaneous control of the exploration and exploitation parameters during the optimization stages as the primary advantage.
- ✓ Higher reliability in searching the entire problem space, minimizing the likelihood of falling into local optima, compared to other similar meta-heuristic algorithms.

The proposed algorithm was implemented and evaluated on 56 benchmark test functions, and its performance was compared with nine other algorithms, including HBO [41], GSA [33], PSO [8], SCA [40], MFO [30], MVO [36], CS [32], EBO-CMAR [38], and WSO [11].

The subsequent sections of the paper are structured as follows: Section 2 has provided an overview of the research background. Section 3 furnishes a comprehensive depiction of the proposed YGSA, encompassing a general description and the operational mechanism. Section 4 presents the simulation, initial parameterization, and analysis of the YGSA across various test functions. Finally, the concluding section provides a summary, draws conclusions, and offers suggestions for future research.

2. *Research Gaps and Related works*

The proliferation of optimization algorithms has prompted the No Free Lunch (NFL) [22] theorem to address the fundamental question of whether there is still a necessity for the development of novel algorithms. The NFL theorem elucidates the fact that while an optimization algorithm may demonstrate high performance in solving specific optimization problems, it may falter when applied to a distinct set of optimization problems. This discrepancy arises from the inherent diversity and distinct mathematical models of real-world problems. Consequently, there is no guarantee that any particular optimization algorithm will exhibit high efficacy in solving all

Power System Technology
ISSN:1000-3673

Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

optimization problems [23]. Therefore, the existence of a unified optimization algorithm that can consistently provide satisfactory results for all optimization problems is unfounded. Consequently, the allure of research in this domain persists, driving researchers to propose new optimization algorithms. Figure (1) depicts the classification of the presented some algorithms.

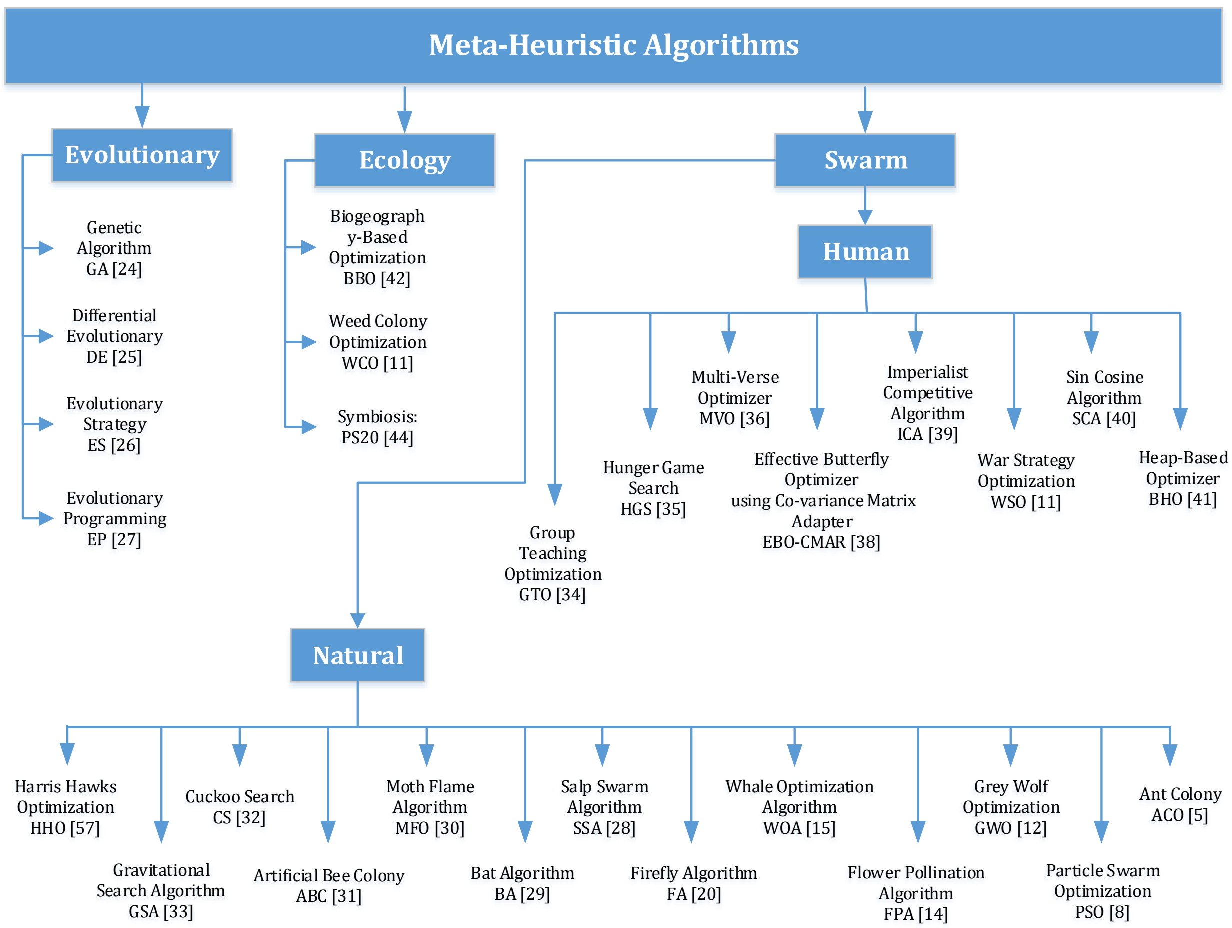


Figure 1. Category of meta-heuristic algorithms

Considering the distinctive features of each novel algorithm employed for achieving optimal solutions across diverse problem domains, there are numerous challenges and issues to be addressed [11, 37]. These challenges encompass the following aspects:

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

1. Some algorithms exhibit slow convergence towards the global optimal solution and impose a substantial computational burden.
2. Certain algorithms lack a suitable balance between the exploration and exploitation phases, addressing them disparately at different stages.
3. Several algorithms suffer from premature convergence to local optima, rendering them unsuitable for solving real-world problems.
4. Certain algorithms encompass a multitude of adjustable parameters, with their effectiveness heavily dependent on these parameters. Optimal selection of parameter values necessitates a significant computational effort.

3. *The proposed method*

This section presents the introduction of the proposed YGSA and provides an explanation of the modeling process for its various components.

3.1. Motion and Distance Modeling

Motion and distance modeling play a crucial role in the proposed algorithm due to the pursuit and evasion behaviors exhibited by the farmer (predator) and the squirrel. Understanding the environment and its impact on all three agents, namely the farmer, squirrel, and nest, is pivotal as it alters the problem conditions. Thus, it becomes necessary to introduce parameters that formulate the motion of the squirrel and farmer.

The key optimization aspect of the YGSA lies in calculating distances between the three elements: farmer, squirrel, and nest, within a two-dimensional environment. Consequently, several relationships and functions can be formulated based on the environment and the motion model of the farmer and squirrel to represent their movements. Before selecting the appropriate distance calculation method, it is crucial to investigate the pursuit and evasion environment and choose the most suitable model for simulation. Previous research has examined spaces and distance calculation methods within this context [45].

Parameters such as the previous and next locations of the farmer and the previous location of the squirrel do not significantly affect the pursuit and evasion process in the proposed algorithm. Furthermore, the motion modeling of the squirrel and farmer employs interrupted motion behavior inspired by the natural observation of squirrels escaping from farmers. These models assume the absence of external agents influencing the internal environment of the pursuit and evasion space,

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

where the movement of the squirrel and farmer is limited to a two-dimensional setting. Based on the results obtained from Table (1), a homogeneous Euclidean space is considered the most suitable for simulating the motion environment of the squirrel and farmer. This choice is motivated by its simplicity in implementation and applicability in two-dimensional pursuit and evasion scenarios. After selecting the space type for simulating the pursuit and evasion environment, an appropriate method is required to calculate the distances between the squirrel and farmer, as well as the distance between the squirrel and nest (line calculations).

Table 1. Comparison of motion spaces [45]

No.	Distance Functions	Strengths	Weaknesses
1	Limited Euclidean space	Simple, easy to deploy, suitable for situational data tracking	Risk of not considering initial effects (object relationship with environment)
2	Aquarium place-time	Homogeneous environment with difference in restriction of object movement	Risk of not considering initial effects (object relationship with environment)
3	Heterogeneous field space	Suitable for visualization, allowing changes in display	Limited to small samples
4	Mosaic space	Simple, easy to use, interaction with environment	Unsuitable for movement patterns on a smaller surface than cell surface
5	Network space	Allows comparison, integration, and categorization of high-level trajectories	Differences in cell size and shape limit comparison between surfaces of different sizes
6	Euclidean homogeneous space	Ability to generalize space, accurate modeling of heterogeneous motion constraints	Limited significant range of motion patterns

Various distance functions exist for calculating the distance between two points in a homogeneous Euclidean space, including Euclidean distance, Manhattan distance, dynamic time warping, Fréchet distance, and Hausdorff distance [46, 47, 48, 49]. Considering the nature of the

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

homogeneous Euclidean space, which is the optimal choice for the proposed algorithm, Euclidean distance is the most suitable for calculating the distances between the squirrel, farmer, and nest. This selection is justified by the pursuit and evasion occurring within a two-dimensional plane, and the ease of implementing Euclidean distance in a homogeneous Euclidean space without incurring a heavy computational burden [49, 50].

With the adoption of the homogeneous Euclidean environment and Euclidean distance in the proposed algorithm, equations (1) and (2) can be employed to calculate the distance between two points on a two-dimensional plane.

$$D_{euc} = L_p = (T_1, T_2) = \sqrt[p]{\sum_{i=1}^{n} (T_1^i - T_2^i)^p} \quad p = 2 \tag{1}$$

$$|T_1| = |T_2| = n \tag{2}$$

Here, T_1 and T_2 represent the desired points on the n-dimensional plane [50, 51].

3.2. Formulation of Squirrel and Farmer in YGSA

This section aims to delve into the modeling of the yellow ground squirrel and the farmer's behavior within the YGSA. Furthermore, the performance of the proposed algorithm will be showcased through the utilization of a flowchart.

3.2.1 Yellow Ground Squirrel Behavior

Yellow ground squirrels, predominantly found in East Asian countries, inhabit agricultural fields and sustain themselves by feeding on roots, fruits, plants, and crops cultivated by farmers, consequently causing damage to agricultural products [52, 53]. Thus, the presence of these squirrel species in agricultural lands poses a threat to horticultural and agricultural yields. To safeguard their crops, farmers endeavor to ensure the removal or capture of these squirrels from their farmland. However, locating and capturing these squirrels in agricultural fields is a challenging task. Upon spotting a squirrel, the farmer initiates the pursuit, while the squirrel strives to maintain a constant and safe distance from the farmer, even attempting to increase this distance to enhance its own safety. Similar to the farmer, the squirrel exhibits intelligent behavior by employing crafty tactics to escape and approach its nest. Throughout the pursuit, the squirrel strives to keep a

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

constant distance from the farmer. Once it reaches what it perceives as a safe distance from the farmer, it briefly pauses to evaluate its next movement paths. Subsequently, as the farmer resumes movement towards the squirrel, the squirrel repeats this process. Simultaneously, at each stage of pursuit and evasion, the squirrel endeavors to locate its nest and decrease the distance to the nest while increasing the distance from the farmer. This behavior during the squirrel's evasion from the farmer and its pauses at each stage can serve as a strategy to determine the next suitable position that maintains a safe distance from the farmer and facilitates nest localization. By meticulously adhering to this movement pattern, the squirrel achieves successful evasion and outwits the farmer.

It is important to note that squirrels, like any other living organism, have inherent limitations. One such limitation is their incomplete perception of the surrounding environment. Squirrels may not accurately recognize the environment they inhabit. Additionally, the farmer may persistently pursue the squirrel, leaving the squirrel with no opportunity to maintain a safe distance and preventing it from selecting a suitable next position.

Hence, the following assumptions are made for this problem:

1. The squirrel and the farmer are considered as two intelligent agents.
2. There exists only one squirrel, one farmer, and one nest in the search space.
3. The nest's location remains constant in each iteration.
4. The farmer cannot pursue and capture the squirrel directly in a straight line.
5. In the initial step, the location of the nest is predefined, while the locations of the squirrel and the farmer are randomly assigned on the plane.
6. In each iteration (pursuit and evasion), the farmer and the squirrel have visibility of each other and are aware of each other's positions.
7. The farmer takes the first step towards the squirrel, followed by the squirrel's movement (escaping from the farmer while maintaining a safe distance).
8. The squirrel's movement speed is greater than that of the farmer.

Considering the squirrel's escape nature and its methodology at each stage, the squirrel strives to maintain or increase its safe distance from the farmer. Consequently, the farmer is unable to pursue the squirrel in a straight line.

Power System Technology
ISSN:1000-3673

Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

3.2.2. Farmer Behavior Modeling

The farmer, acting as an intelligent agent, seeks to capture the squirrel. The farmer must act strategically to decrease the distance between themselves and the squirrel in order to catch it. However, due to the squirrel's evasive nature, the farmer cannot directly pursue the squirrel in a straight line. Therefore, to prevent the squirrel from reaching its nest, the farmer needs to create various positions between the nest and the squirrel for effective movement. As squirrels in nature possess their own intelligence for evading farmers, this pursuit and evasion process continues. In the proposed algorithm, to account for the farmer's intelligence as an agent capable of measuring distances and the inability to capture the squirrel directly in a straight line, the farmer attempts to update their position using curvature (θ) in the path. This updating of the farmer's position aims not only to reduce the distance to the squirrel but also to obstruct the squirrel's entry into the nest. To achieve this, the farmer first calculates the distance between themselves and the squirrel (d_{fs}) and the distance between themselves and the nest (d_{fn}). By considering these distances as two vectors, vector d_{f+1} can be determined to estimate the farmer's next position. Figure (2) illustrates an example of the positions and distances of the farmer, squirrel, and nest, where (x_s, y_s), (x_f, y_f), and (x_n, y_n) represent the positions of the squirrel, farmer, and nest, respectively.

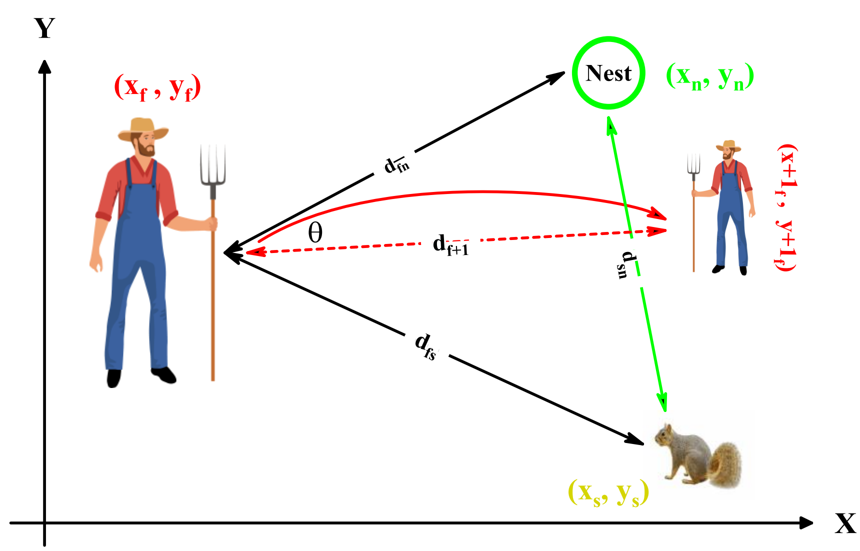


Figure 2. Updating the farmer's position using curvature (θ)

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

3.3 YGSA (Yellow Ground Squirrel Algorithm)

Optimization algorithms are widely employed in tackling complex problems and scenarios where an exact and precise solution is unattainable or inconceivable. While these algorithms exhibit high efficacy in addressing such problems, each algorithm possesses its own limitations and deficiencies, including susceptibility to local optima, extended convergence time, applicability to specific rather than general contexts, accurate determination of solution quality and optimality, and lack of population diversity [54, 55]. Conversely, certain algorithms necessitate an initial population, such as GA [24, 56] and PSO [8], which may not be available for all problem conditions. Furthermore, algorithms like PSO [8], BA [29], GWO [12], Salp Swarm Algorithm (SSA) [28], Heap-Based Optimizer (HBO) [41], and War Strategy Optimizer (WSO) [11] leverage swarm intelligence, thereby demanding both an initial population and interconnectivity within the population. In contrast, the yellow ground squirrel solely interacts with the farmer and evades their pursuit to avoid being hunted. The squirrel does not benefit from swarm intelligence and instead relies solely on its positional information. The flowchart of the proposed YGSA is depicted in Figure (3).

As previously mentioned, the initial positions (initial points) of the squirrel and the farmer are randomly determined. Additionally, the nest's position remains constant throughout the algorithm. Assuming that the squirrel and the farmer are aware of each other's positions during all pursuit and evasion iterations, the first stage involves the farmer moving towards the squirrel. However, the squirrel strives to reduce its distance to the nest while maintaining or increasing its distance from the farmer. In other words, the squirrel's position is updated based on minimizing the distance to the nest and maximizing the distance to the farmer. Subsequently, in the following stage, the farmer employs curvature in its movement path to update its position in the search space, simultaneously reducing the distance to the squirrel and increasing the distance between the squirrel and the nest. The termination condition of this algorithm can be met through three scenarios: reaching a specified number of iterations, the squirrel entering the nest, or the farmer catching the squirrel.

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

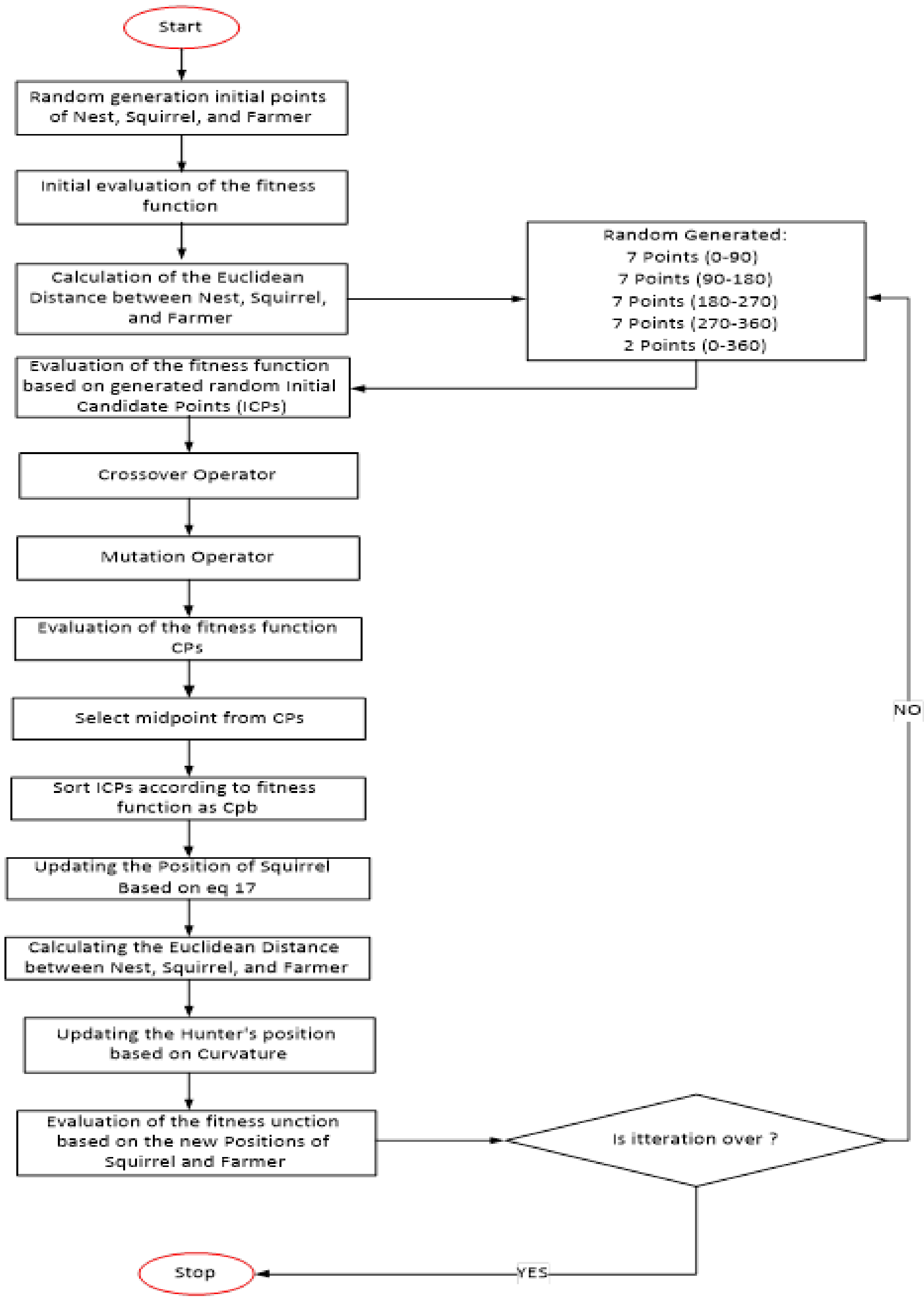


Figure 3. YGSA flowchart

Power System Technology
ISSN:1000-3673

Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

To formalize the proposed YGSA, several steps must be taken into account, which are as follows:

Step 1: Head Rotation

During each iteration of the pursuit and evasion process, the yellow ground squirrel assesses its surrounding environment by rotating its head 90 degrees in four directions, encompassing a complete 360-degree rotation, in order to determine its subsequent position. This behavior is inspired by observations of real-life animal behavior in nature. The full 360-degree rotation is considered necessary due to the farmer's constantly changing position relative to the squirrel. Figure (4) depicts the positions of the yellow ground squirrel, farmer, and nest within a Cartesian coordinate system, specifically in a two-dimensional space.

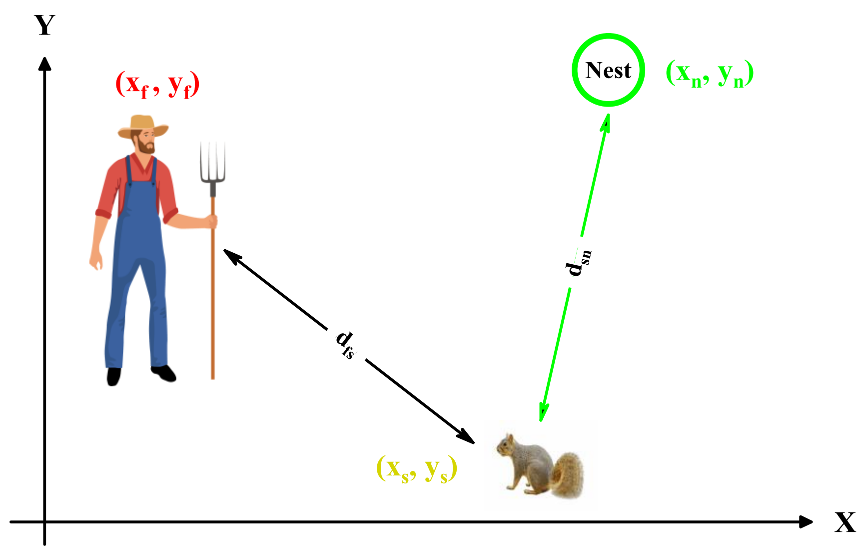


Figure 4. Positioning the hunter, squirrel and nest in the cartesian coordinate system

In light of simulating the rotational behavior of the squirrel's head to encompass the entire 360-degree range, it becomes essential to introduce a conceptual term known as the unit circle. The unit circle represents a circle with a radius of 1 within a Cartesian coordinate system on the Euclidean plane [58, 59]. Figure (5) provides an illustration of a sample unit circle.

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

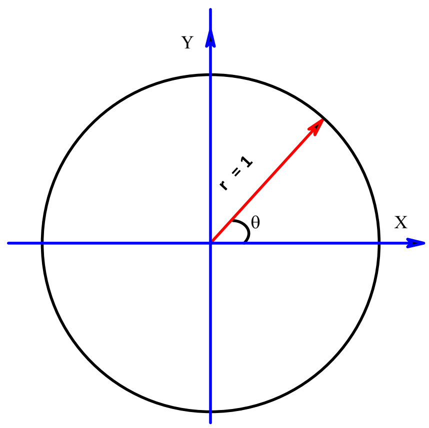


Figure 5. The unit circle

By utilizing the unit circle, it becomes possible to calculate trigonometric ratios and ascertain angles and points within a two-dimensional plane. Equation (3) can be employed to convert angles into corresponding points $(x\,,y)$ within the two-dimensional coordinate system.

$$(x\,,y) = \begin{cases} x = d * cos\theta + x \\ y = d * sin\theta + y \end{cases} \tag{3}$$

As depicted in Figure (6), the yellow ground squirrel initiates its pursuit and evasion strategy by selecting multiple random angles within Quadrant 1 (C_1) through head rotation. Each selected angle corresponds to a hypothetical squirrel occupying the same position, resulting in the random distribution of multiple squirrels within this quadrant. This process of squirrel distribution is then repeated for Quadrants 2, 3, and 4. In the general scenario, considering the random distribution of 7 squirrels in each quadrant and 2 squirrels at two random angles encompassing the full 360-degree range, a total of 30 candidate positions are generated for the subsequent movement. This behavior exemplifies the exploratory nature of the algorithm in determining the next positions for the squirrel.

Power System Technology
ISSN:1000-3673

Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

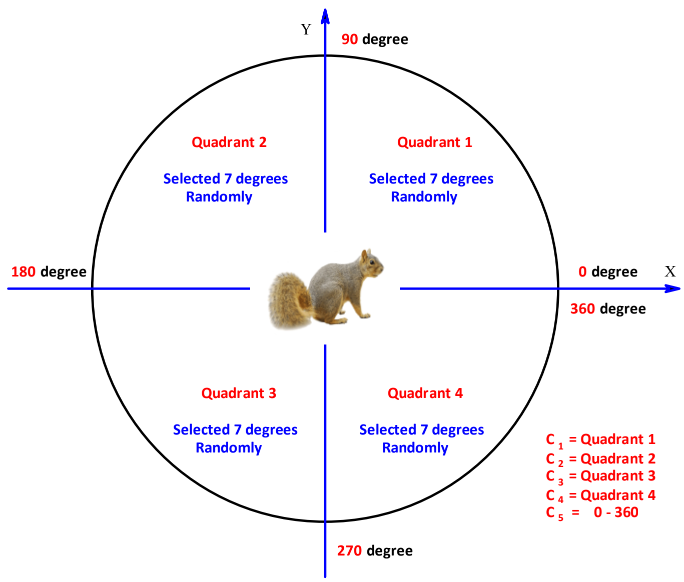


Figure 6. Distribute squirrels in each quadrant of the circle

Equations (4) to (12) demonstrate the calculation process of candidate points for the subsequent movement of the squirrel in each of the four trigonometric quadrants (C_1 to C_4), as well as C_5, using randomly generated angles. These equations provide a method for determining the candidate points encompassing the entire 360-degree range for the squirrel's future motion.

$$C_1 = Deg\ X_i = Random\ Deg[0\ to\ 90], \quad i = 1,2,3, \dots ,7 \tag{4}$$

$$(x.y) = \begin{cases} y = sin\theta \\ x = cos\theta \end{cases} \tag{5}$$

$$C_2 = Deg\ X_i = Random\ Deg[90\ to\ 180], \quad i = 1,2,3, \dots ,7 \tag{6}$$

$$(x, y) = \begin{cases} y = \sin\theta \\ x = -\cos\theta \end{cases} \tag{7}$$

$$C_3 = Deg\ X_i = Random\ Deg[180\ to\ 270], \quad i = 1,2,3, \dots ,7 \tag{8}$$

$$(x, y) = \begin{cases} y = -\sin\theta \\ x = -\cos\theta \end{cases} \tag{9}$$

$$C_4 = Deg\ X_i = Random\ Deg[270\ to\ 360], \quad i = 1,2,3, \dots ,7 \tag{10}$$

$$(x, y) = \begin{cases} y = sin\theta \\ x = -cos\theta \end{cases} \tag{11}$$

$$C_5 = Deg\ X_i = Random\ Deg[0\ to\ 360], i = 1,2 \tag{12}$$

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

Figure (6) illustrates the random distribution of squirrels in the trigonometric quadrants C_1 to C_4, as shown by Equations (4) to (12), aiming to enhance the exploration of the entire 360-degree state space and reduce the likelihood of converging to local optimum C_5. Equation (12) is utilized to select two random angles within the complete 360-degree range. To ensure uniqueness among candidate points, any duplicate selection is substituted with another randomly chosen point.

The rationale behind the random selection in each trigonometric quadrant and the full 360-degree range is to mitigate the risk of being trapped in local optima. This methodology bears resemblance to Levy flights [60, 61] and random walks [62, 63]. The decision to partition the 360-degree range into four trigonometric quadrants draws inspiration from observations of squirrel behavior in the natural world. Furthermore, the choice of seven angles in each quadrant is driven by the trade-off between computational complexity and the extent of solution exploration. Higher numbers would impose greater computational overhead, while lower numbers would restrict the exploration of potential solutions.

Step 2: Crossover and Mutation Operators

The YGSA incorporates crossover and mutation operators from the GA [64, 65] to generate new candidate points based on the initial squirrel movements obtained using Equation (13). This step involves expanding the Initial Candidate Points (ICPs) from 30 to 60, with the additional 30 points generated through crossover and mutation operations using the best candidate points for squirrel movement. In the YGSA, a chromosome encoding scheme is proposed where points are represented as (x, y) coordinates in a two-dimensional plane, and a single-point crossover operator is utilized.

The mutation operator plays a crucial role in preventing premature convergence to local optima by enabling exploration of the entire search space. In the implementation, 80% of the initial candidate points are selected for crossover operations, while the remaining 20% are allocated for mutation operations. Moreover, Equation (13) is employed to calculate the probabilities of crossover and mutation for the newly created Candidate Points (CPs). This approach yields a total of 60 CPs, including those generated through mutation and crossover operations.

$$CPs = \sum ICPs + CPs_{mutation} + CPs_{crossover} = 60 \tag{13}$$

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

Step 3: The Fitness Function

In the previous stage, an investigation was conducted into the process of selecting initial candidate points, taking into account the angles of the squirrel, as well as the generation of new candidate points via mutation and crossover operators. Figure (4) provides a visual representation of the squirrel's distance to the farmer (d_{fs}), its distance to the nest (d_{sn}), and their respective positions in a two-dimensional plane. As mentioned earlier, the proposed algorithm relies on the distances between the squirrel and the farmer, the squirrel and the nest, as well as their Euclidean ratio to drive its operation. Moreover, the squirrel endeavors to maintain a constant or increased distance from the farmer while simultaneously decreasing its distance to the nest. Reaching the nest ensures the squirrel's safety, making the distance from the squirrel to the nest (d_{sn}), a crucial metric that the squirrel aims to minimize. Consequently, the new candidate points derived from the preceding step undergo sorting based on the shortest squirrel-to-nest distance (d_{sn}), with Equation (14) employed as the initial fitness function.

$$Initial\ Fitness = \ sort\left(\min\left(d_{sn}\right)\ from\ \ CPs\right) \tag{14}$$

The fitness calculation involves utilizing a sorted set of 60 members (candidate points) and selecting the 30 members with the shortest distances to the nest. Following the principle of data distribution, the value around which data points are distributed is referred to as the central value, and any numerical measure representing the center of a dataset is known as a measure of centrality [66, 67, 68]. This principle is employed to model the algorithm and prevent premature convergence to local optima. While the best CPs (c_{bp}) set is sorted based on the shortest squirrel-to-nest distance, an effort is made to increase the likelihood of selecting other members by designating the median member of this set, thereby avoiding local optima. Subsequently, the calculation of m, which represents the median member of the c_{bp} set, is performed using Equation (15). The median member is considered a candidate point, and two superior members (with shorter squirrel-to-nest distances) and two inferior members (with greater squirrel-to-nest distances) relative to m are also selected. Figure (8) and Equation (16) illustrate the process of selecting the two preceding and two subsequent members of m within the {s} set.

$$m = \frac{c_{bp}}{2} \tag{15}$$

$$s = \{m-2, m-1, m+1, m+2\}\ ) \tag{16}$$

Power System Technology
ISSN:1000-3673

Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

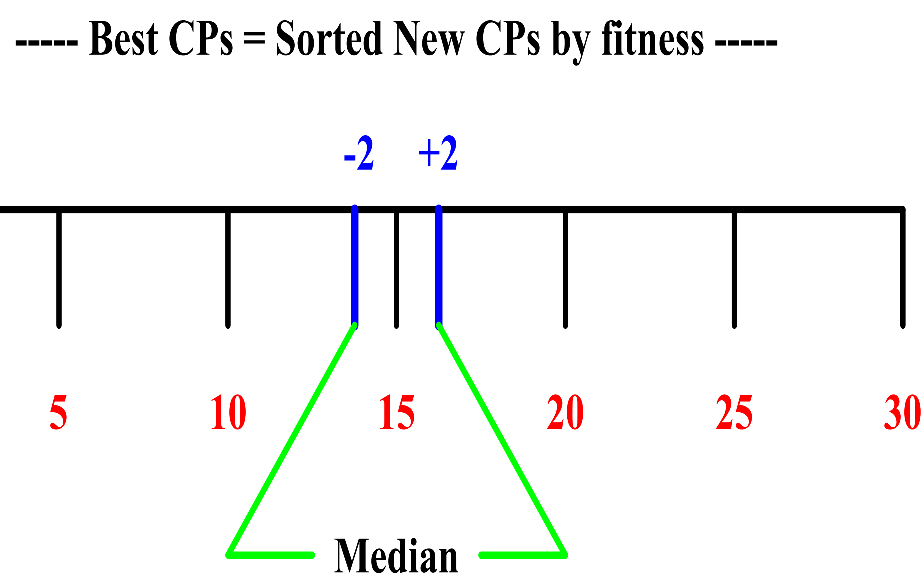


Figure 8. Calculating best candidate points using median methods

Step 4: Selecting the Next Position

In the fourth step of the algorithm execution, the subsequent position of the squirrel is updated by selecting a candidate point from the {s} set that has the maximum distance to the farmer. The next position of the squirrel is determined according to Equation (17).

$$d_{s+1} = Max\ (d_{fs})\ from\ \{s\} \tag{17}$$

The selection of the next position involves utilizing the set obtained from step 1 and sorting the New CPs set based on the shortest distance from the squirrel to the nest ((d_{sn}) in step 2. Additionally, in step 3, the {s} set is identified, and in step 4, a member with the maximum distance from the squirrel to the farmer (d_{fs}) is chosen. By following this intelligent approach, the algorithm enables a balance between exploration and exploitation in each iteration, thus preventing both elitism and getting trapped in local optima.

Power System Technology
ISSN:1000-3673

Received: 14-10-2023 *Revised: 12-11-2023* *Accepted: 20-12-2023*

1- Initialize a squirrel position and a farmer position randomly

2- Initialize the movement rate for the squirrel and the farmer

3 - Calculate the Euclidean distance d_{sn}*(a cost function).*

4- Calculate the Euclidean distance d_{fs}.

5- Create a potential squirrel Candidate Points (CPs) to determine the new position of the squirrel as follows:

6- ***While*** *(t < Max iteration or (*d_{fs} *<0.0001 or* d_{sn} *<0.0001))*

1. *Divide the coordinate of the problem space into* C_1*-*C_5
2. $\text{Initial Canddiate Points (ICPs)} = \sum_{i=1}^{4} c_i = (4 * 7) + 2$
3. $\text{New CPs} = \sum \text{ICPs} + \text{CPs}_{\text{mutation}} + \text{CPs}_{\text{crossover}} = 60$
4. $\text{Fitness} = \text{sort}\left(\min\left(d_{sn}\right) \text{ from New CPs} = 30\right)$
5. $m = \frac{C_{\text{Best CPs}}}{2}$
6. $s = \{m-2, m-1, m+1, m+2\}$

7- $d_{s+1} = \text{Max}\{s\}$*; That is the squirrel's distance must be bigger than the farmer's distance and less than the nest position as well.*

8- Update the squirrel position to the new position according to the considered movement rate

9- Calculate the consequent d_{fs} *and* d_{fn}

10- Use the θ *in the direction of the farmer between the squirrel and nest*

11- Update the farmer position according to step 10

12- *Check the terminate condition; if it is not satisfied, move to step 1; otherwise, stop the algorithm.*

End while

Figure 9. Steps of the proposed YGSA

To determine the coordinates of the points and normalize their values, the Min-Max normalization method [69, 70] is employed. This method involves scaling the data within the range of [0, 1]. Min-Max normalization is commonly used when assessing the similarity between points due to its simplicity in implementation and effectiveness [70, 72]. Taking into account the aforementioned considerations in the design stages of the proposed algorithm, Figure (9) illustrates the execution steps of the YGSA.

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

4. Results and Discussion

This section presents the evaluation of the performance of the proposed YGSA. The evaluation encompasses the introduction of the experimental environment, test functions, initial experiment parameters, and the compared algorithms. Subsequently, the obtained results are thoroughly analyzed and discussed.

4.1 Experimental Environment

There exists a wide array of meta-heuristic algorithms designed to solve diverse problems, each with its own set of advantages, contingent upon the application and problem characteristics. In this study, our objective is to introduce the proposed YGSA as a straightforward yet effective approach for tackling complex problems. The implementation of the proposed algorithm is depicted in Figures (3) and (9), illustrating its flowchart and step-wise execution, respectively. MATLAB software version 2020a was utilized for the implementation, using a computer with a Windows 7 Professional 64-bit operating system and 8 gigabytes of RAM. Various test functions were employed to evaluate the algorithm's performance across different scenarios.

4.2 Test Functions and Comparable Algorithms

To assess the performance of the proposed algorithm and compare it with other existing algorithms, a set of 56 test functions, encompassing both unimodal and multimodal complex functions, was employed [71,73,78]. Table (2) presents the details of these test functions and their respective parameters for evaluating the proposed algorithm, as well as nine other algorithms. Additionally, Tables (3) to (7) depict relevant characteristics for each category of functions [11, 74, 75]. In each table, "Vars" denotes the number of dimensions of the functions, "Range" indicates the upper and lower bounds of the variables, and "f_{min}" represents the minimum global value of the functions.

Furthermore, to conduct a comprehensive evaluation of the proposed YGSA, a comparative analysis was performed against nine other algorithms, including HBO [41], GSA [33], PSO [8], SCA [40], MFO [30], MVO [36], CS [32], EBO-CMAR [38], and WSO [11]. This comparative analysis aims to enhance the validation of the obtained results from the proposed algorithm.

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

Table 2. Test functions

No.	F. No	Name	Table No
1	F1 – F9	Unimodal fixed-dimension	Table (3)
2	F10 – F23	Unimodal variable-dimension	Table (4)
3	F24 – F33	Multimodal fixed-dimension	Table (5)
4	F34 – F46	Multimodal variable-dimension	Table (6)
5	F47 – F56	CEC-06 2019	Table (7)

Table 3. Unimodal fixed-dimension test functions

F. No.	Name	Vars	Range	$f$min
F1	Beal	2	[-4.5, 4.5]	0
F2	Booth	2	[-10, 10]	0
F3	Brent	2	[-10, 10]	0
F4	Matyas	2	[-10, 10]	0
F5	Schaffer N.4	2	[-100, 100]	0.292579
F6	Wayburn Seader 3	2	[-500, 500]	19.10588
F7	Leon	2	[-1.2, 1.2]	0
F8	Cube	2	[-10, 10]	0
F9	Zettl	2	[-5, 10]	-0.00379

Table 4. Unimodal variable-dimension test functions.

F. No.	Name	Vars	Range	$f$min
F10	Sphere	30	[-100, 100]	0
F11	Powell sum	30	[-1, 1]	0
F12	Schwefel’s 2.20	30	[-100, 100]	0
F13	Schwefel’s 2.21	30	[-100, 100]	0
F14	Step	30	[-100, 100]	0
F15	Stepint	30	[-5.12, 5.12]	-155
F16	Schwefel’s 2.22	30	[-100, 100]	0

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

F17	Schwefel's 2.23	30	[-100, 100]	0
F18	Rosenbrock	30	[-30, 30]	0
F19	Brown	30	[-1, 4]	0
F20	Dixon and price	30	[-10, 10]	0
F21	Powell Singular	30	[-4, 5]	0
F22	Xin-Shi Yang	30	[-20, 20]	0
F23	Perm 0, D, Beta	5	[-Var, Var]	0

Table 5. Multimodal fixed-dimension test functions.

F. No.	Name	Vars	Range	***f*** min
F24	Egg Crate	2	[-5, 5]	0
F25	Ackley N.3	2	[-32, 32]	-195.629
F26	Adjiman	2	[-1, 2]	-2.02181
F27	Bird	2	[-2pi, 2pi]	-106.765
F28	Camel 6 Hump	2	[-5, 5]	-1.0316
F29	Branin RCOS	2	[-5, 5]	0.397887
F30	Hartman 3	3	[0, 1]	-3.86278
F31	Hartman 6	6	[0, 1]	-3.32236
F32	Cross-in-tray	2	[-10, 10]	-2.06261
F33	Bartels conn	2	[-500, 500]	1

Table 6. Multimodal variable-dimension test functions.

F. No.	Name	Vars	Range	***f*** min
F34	Schwefel's 2.26	30	[-500, 500]	-418.983
F35	Rastrigin	30	[-5.12, 5.12]	0
F36	Periodic	30	[-10, 10]	0.9
F37	Qing	30	[-500, 500]	0
F38	Alpine N.1	30	[-10, 10]	0
F39	Xin-She Yang	30	[-5, 5]	0
F40	Ackley	30	[-5, 5]	0
F41	Trignometric 2	30	[-500, 500]	1
F42	Salomon	30	[-100, 100]	0

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

F43	Griewank	30	[-100, 100]	0
F44	Xin-She Yang N.4	30	[-10, 10]	-1
F45	Xin-She Yang N.2	30	[-2pi, 2pi]	0
F46	Gen.Penalized	30	[-50, 50]	0

Table 7. CEC-06-2019 test functions.

f. No.	Name	Vars	Range	*f* min
F47	CEC01	9	[-8192, 8192]	1
F48	CEC02	16	[-16384,16384]	1
F49	CEC03	18	[-4, 4]	1
F50	CEC04	10	[-100, 100]	1
F51	CEC05	10	[-100, 100]	1
F52	CEC06	10	[-100, 100]	1
F53	CEC07	10	[-100, 100]	1
F54	CEC08	10	[-100, 100]	1
F55	CEC09	10	[-100, 100]	1
F56	CEC10	10	[-100, 100]	1

4.3 Initial Parameters of the Proposed YGSA

The proposed YGSA incorporates several parameters, each with its respective initial values. Specifically, the initial number of candidate points is set to 30, the crossover probability is set to 0.8, and the mutation probability is set to 0.2. Furthermore, the algorithm takes into account the movement speed parameters for both the farmer and the squirrel. Accordingly, the farmer's movement speed is assigned a value of 0.2, while the squirrel's movement speed is set to 0.3, considering that squirrels typically exhibit higher movement speeds compared to farmers in natural settings.

In addition to a predetermined number of iterations, the simulation process incorporates two termination criteria. One criterion is based on the distance between the farmer and the squirrel. When the distance between the farmer and the squirrel reaches the default threshold of 0.0001 (d_{fs} < 0.0001), the algorithm terminates, signifying the successful capture of the squirrel by the farmer. Moreover, if the value of d_{sn} becomes less than 0.0001, indicating that the squirrel has reached

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

the nest, the algorithm also terminates. Table (8) provides an overview of the settings and initial parameters employed in the proposed YGSA.

Table 8. YGSA parameters setting

No.	Initial Parameter	Value
1	Initial Candidate points size	30
2	P_c (Crossover)	0.8
3	P_m (Mutation)	0.2
4	V (Speed of Squirrel)	0.3 m/s
5	V (Speed of Farmer)	0.2 m/s
6	Threshold	0.001 m
7	Times of execution algorithm	30 times
8	Iteration in each execute	500 iterations

4.4 Analysis of Experiment Results

The YGSA has been executed 30 times for each test function, with 500 iterations per run. The evaluation of the proposed algorithm's overall performance involves comparing the results obtained from 30 independent runs with those of other algorithms using the med (minimum), mean, and std criteria. The med criterion, representing the minimum mean, serves as the primary measure for assessing and comparing the efficiency of each algorithm. The performance of the YGSA, in comparison to the other investigated algorithms, is presented in Tables (9) to (15), which correspond to the test functions.

a. Evaluation of unimodal test functions

The initial set of test functions aims to evaluate the overall performance of the YGSA on unimodal test functions. This category encompasses two subsets: unimodal fixed-dimension test functions and unimodal variable-dimension test functions. According to Tables (3) and (4), functions F1 to F23 are classified under this category. For each function, the med, mean, and std parameters are calculated to facilitate the evaluation process. The evaluation results are reported in Tables (9) and (10), where the bold formatting highlights the optimal results obtained from the experiments for each algorithm.

Power System Technology
ISSN:1000-3673

Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

Table 9. Comparison results for 9 unimodal fixed-dimension test functions

F. No	State	YGSA	HBO	GSA	PSO	SCA	MFO	MVO	CS	EBO-CMAR	WSO
F1	med	**0.000E+00**	**0.000E+00**	6.290E-21	**0.000E+00**	7.23E-05	**0.00E+00**	1.60E-08	**0.00E+00**	**0.00E+00**	**0.00E+00**
	mean	**0.000E+00**	**0.000E+00**	9.480E-21	**0.000E+00**	7.88E-05	2.35E-32	1.98E-08	**0.00E+00**	**0.00E+00**	**0.00E+00**
	std	**0.000E+00**	**0.000E+00**	7.410E-21	**0.000E+00**	5.60E-05	1.00E-31	1.44E-08	**0.00E+00**	**0.00E+00**	**0.00E+00**
F2	med	1.314E-04	**0.000E+00**	8.290E-21	**0.000E+00**	1.48E-04	**0.00E+00**	1.36E-07	**0.00E+00**	**0.00E+00**	**0.00E+00**
	mean	4.517E-04	**0.000E+00**	1.200E-20	**0.000E+00**	3.13E-04	**0.00E+00**	1.77E-07	**0.00E+00**	**0.00E+00**	**0.00E+00**
	std	3.568E-03	**0.000E+00**	1.300E-20	**0.000E+00**	3.31E-04	**0.00E+00**	1.61E-07	**0.00E+00**	**0.00E+00**	**0.00E+00**
F3	med	**1.380E-87**	**1.380E-87**	1.510E-05	**1.380E-87**	**1.380E-87**	**1.380E-87**	**1.380E-87**	**1.380E-87**	**1.380E-87**	**1.380E-87**
	mean	**1.380E-87**	**1.380E-87**	4.090E-05	**1.380E-87**	**1.380E-87**	**1.380E-87**	**1.380E-87**	**1.380E-87**	**1.380E-87**	**1.380E-87**
	std	1.360E-101	6.800E-103	6.780E-05	4.600E-103	4.60E-103	4.60E-103	4.60E-103	4.60E-103	2.40E-103	2.35E-103
F4	med	3.610E-69	5.100E-107	4.680E-05	1.300E-95	1.70E-132	2.60E-142	3.55E-09	1.12E-55	**0.00E+00**	**0.00E+00**
	mean	1.345E-08	9.200E-78	6.880E-22	7.360E-93	2.10E-121	2.75E-66	5.03E-09	1.26E-52	**0.00E+00**	**0.00E+00**
	std	3.009E-07	5.050E-77	7.430E-22	3.080E-92	9.800E-121	1.380E-65	5.210E-09	4.330E-52	**0.000E+00**	**0.000E+00**
F5	med	**2.926E-01**	**2.926E-01**	**3.014E-01**	**2.926E-01**	**2.926E-01**	**2.926E-01**	**2.926E-01**	**2.926E-01**	**2.926E-01**	**2.926E-01**
	mean	**2.926E-01**	**2.926E-01**	**3.064E-01**	**2.926E-01**	**2.926E-01**	**2.927E-01**	**2.926E-01**	**2.926E-01**	**2.926E-01**	**2.926E-01**
	std	2.340E-05	7.140E-17	7.930E-17	7.930E-17	1.760E-07	2.880E-04	6.190E-07	6.730E-11	9.200E-09	**4.530E-17**

Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

F6	med	**1.911E+01**	**1.911E+01**	**1.911E+01**	**1.911E+01**	**1.911E+01**	**1.911E+01**	**1.911E+01**	**1.911E+01**	**1.911E+01**	**1.911E+01**
	mean	**1.943E+01**	**1.911E+01**	**1.911E+01**	**1.911E+01**	**1.912E+01**	**1.911E+01**	**1.911E+01**	**1.911E+01**	**1.911E+01**	**1.911E+01**
	std	6.588E-01	1.450E-14	3.400E-15	9.170E-15	1.465E-02	6.760E-15	1.452E-03	7.470E-15	2.378E-15	**4.530E-17**
F7	med	1.426E-04	8.530E-17	2.570E-03	1.500E-05	6.500E-05	1.156E-03	1.250E-08	**0.000E+00**	7.360E-21	**0.000E+00**
	mean	2.053E-03	3.590E-13	2.669E+03	1.450E-20	1.330E-04	4.254E-03	4.800E-07	**0.000E+00**	3.900E-12	**0.000E+00**
	std	2.219E-03	1.290E-12	2.082E-03	3.860E-20	1.960E-04	5.276E-03	3.400E-08	**0.000E+00**	1.210E-08	**0.000E+00**
F8	med	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**
	mean	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**
	std	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**
F9	med	**-3.790E-03**	**-3.790E-03**	**-3.790E-03**	**-3.790E-03**	**-3.790E-03**	**-3.790E-03**	**-3.790E-03**	**-3.790E-03**	**-3.790E-03**	**-3.790E-03**
	mean	**-3.790E-03**	**-3.790E-03**	**-3.790E-03**	**-3.790E-03**	**-3.790E-03**	**-3.790E-03**	**-3.790E-03**	**-3.790E-03**	**-3.790E-03**	**-3.790E-03**
	std	1.851E-07	1.700E-18	9.900E-19	1.300E-18	2.890E-10	1.330E-18	2.900E-08	1.330E-18	**0.000E+00**	1.800E-10

Table 10. Comparison results for 14 unimodal variable-dimension test functions

F. No	State	YGSA	HBO	GSA	PSO	SCA	MFO	MVO	CS	EBO-CMAR	WSO
F10	med	4.455E-141	4.600E-28	2.400E-17	2.100E-12	2.900E-04	3.100E-10	1.903E-01	7.800E-05	8.300E-16	**0.000E+00**
	mean	8.510E-08	8.500E-27	2.400E-17	4.800E-11	7.370E-03	8.000E+02	2.184E-01	1.100E-04	8.990E-16	**0.000E+00**

Power System Technology
ISSN:1000-3673

Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

	std	1.903E-06	3.600E-06	8.500E-18	2.100E-10	2.329E-02	2.769E+03	6.965E-02	1.100E-04	2.710E-16	**0.000E+00**
F11	med	**0.000E+00**	5.300E-70	5.000E-18	4.900E-25	2.200E-15	3.200E-31	8.600E-08	4.400E-17	5.730E-33	**0.000E+00**
	mean	**0.000E+00**	1.300E-64	2.700E-17	6.600E-22	2.000E-10	1.200E-27	8.800E-08	1.100E-15	6.530E-32	**0.000E+00**
	std	1.631E-13	7.300E-64	6.200E-17	2.800E-21	8.000E-10	5.000E-27	5.900E-08	2.900E-15	1.720E-31	**0.000E+00**
F12	med	**1.419E-296**	2.000E-18	2.300E-08	3.600E-06	1.500E-05	1.000E+02	2.621E+00	1.978E-02	4.560E-06	2.530E-263
	mean	5.649E-05	2.800E-18	2.300E-08	6.800E-06	5.900E-05	7.600E+01	2.803E+00	2.831E-02	5.270E-06	**6.310E-259**
	std	1.263E-03	3.800E-18	3.200E-09	8.900E-06	1.300E-04	8.307E+01	7.980E-01	3.062E-02	2.230E-06	**0.000E+00**
F13	med	7.339E-84	1.138E+00	3.400E-09	4.102E-01	1.172E+01	6.761E+01	6.221E-01	1.059E+01	4.770E-04	**3.530E-262**
	mean	1.533E-07	1.216E+00	1.499E-02	4.067E-01	1.374E+01	6.647E+01	6.552E-01	1.158E+01	8.020E-04	**9.270E-260**
	std	3.425E-06	8.007E-01	7.246E-02	1.181E-01	8.472E+00	2.099E-01	2.099E-01	3.537E+00	7.090E-04	**0.000E+00**
F14	med	5.740E+00	**3.100E-28**	1.900E-17	2.100E-12	4.112E+00	3.300E-10	1.949E-01	1.200E-04	8.250E-16	8.206E-10
	mean	5.740E+00	**1.300E-26**	2.000E-17	1.500E-10	4.203E+00	3.588E+03	1.903E-01	1.700E-04	8.070E-16	1.505E-08
	std	2.312E-14	**5.400E-26**	6.200E-18	5.300E-10	4.928E-01	6.365E+03	2.881E-02	1.800E-04	2.040E-16	4.259E-08
F15	med	**-1.550E+02**	**-1.550E+02**	-1.190E+02	-1.330E+02	-1.070E+02	**-1.550E+02**	-1.470E+02	**-1.550E+02**	-1.220E+02	**-1.550E+02**
	mean	-1.531E+02	**-1.550E+02**	-1.190E+02	-1.322E+02	-1.062E+02	**-1.550E+02**	-1.478E+02	-1.546E+02	-1.061E+02	**-1.550E+02**
	std	5.857E+00	**0.000E+00**	2.937E+00	1.046E+01	4.616E+00	**0.000E+00**	2.398E+00	1.003E+00	3.123E+00	**0.000E+00**

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

F16	med	**1.643E-305**	6.600E-19	5.851E+01	9.700E-06	2.500E-06	4.000E+02	1.100E-19	1.000E+10	1.820E-05	8.720E-261
	mean	6.493E-06	1.500E-18	6.360E+01	2.000E-04	7.500E-05	4.480E+02	1.100E-21	1.000E+10	3.220E-05	**6.470E-258**
	std	1.452E-04	1.700E-18	4.805E+01	6.700E-04	3.000E-04	2.044E+02	3.600E-21	0.000E+00	2.600E-05	**0.000E+00**
F17	med	**5.009E-155**	2.700E-59	3.100E-88	2.100E-19	1.581E-02	8.600E-19	2.000E-15	7.600E-06	8.120E-37	1.18E-122
	mean	1.637E-40	2.800E-50	4.300E-88	2.800E-26	1.712E+03	3.200E-11	1.600E-14	1.510E-03	3.940E-34	**7.10E-47**
	std	3.660E-39	1.500E-49	4.600E-88	1.200E-15	8.249E+03	1.600E-10	4.900E-14	6.310E-03	1.320E-34	**1.59E-45**
F18	med	2.778E+01	7.611E+01	2.608E+01	2.592E+01	3.118E+01	5.552E+02	3.183E+01	2.952E+01	9.934E+00	**1.950E-09**
	mean	2.789E+01	6.384E+01	2.895E+01	3.644E+03	7.335E+01	3.209E+06	2.659E+02	4.890E+01	**9.770E+00**	1.500E+06
	std	3.080E-01	3.350E+01	1.392E+01	1.799E+04	1.032E+02	1.600E+07	5.315E+02	2.919E+01	1.137E+00	**2.610E-07**
F19	med	1.539E-79	5.300E-31	4.200E-17	3.100E+01	2.000E-08	1.200E+01	5.300E-04	6.900E-07	3.430E-15	**0.000E+00**
	mean	4.570E-11	2.500E-30	4.300E-17	3.900E+01	7.800E-07	1.448E+01	5.500E-04	1.300E-06	3.870E-15	**0.000E+00**
	std	1.022E-09	5.300E-30	1.300E-17	2.410E+01	2.800E-06	9.364E+00	1.800E-04	1.300E-06	1.910E-15	**0.000E+00**
F20	med	**6.667E-01**	**6.667E-01**	**6.667E-01**	6.705E-01	7.208E-01	9.572E+01	8.505E-01	6.987E-01	**6.667E-01**	**6.667E-01**
	mean	**6.684E-01**	**6.667E-01**	**6.727E-01**	9.616E+01	2.613E+00	4.029E+04	2.075E+00	1.019E+00	**6.667E-01**	**6.667E-01**
	std	2.094E-02	2.800E-04	3.027E-02	1.340E+02	4.858E+00	8.389E+04	2.713E+00	6.391E-01	**3.010E-12**	2.870E-04
F21	med	7.339E-69	1.660E-03	1.148E-02	3.903E+02	1.223E-02	2.112E+02	3.201E-01	7.100E-03	5.120E-09	**0.000E+00**
	mean	3.841E-08	1.860E-03	2.005E-02	6.910E+02	2.521E+00	8.923E+02	3.204E-01	1.020E-02	8.810E-09	**0.000E+00**

Power System Technology
ISSN:1000-3673

Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

	std	8.588E-07	7.400E-04	2.257E-02	8.128E+02	1.216E+01	1.296E+03	1.522E-01	6.391E-01	1.340E-08	**0.000E+00**
F22	med	**-1.000E+00**	4.000E-232	3.600E-41	4.000E-232	6.000E-199	4.000E-232	1.000E-177	4.000E-232	4.300E-232	4.300E-232
	mean	-9.980E-01	**4.000E-232**	1.600E-35	**4.000E-232**	5.000E-189	**4.000E-232**	2.000E-159	**4.000E-232**	**4.300E-232**	**4.300E-232**
	std	4.472E-02	**0.000E+00**	5.000E-35	**0.000E+00**	**0.000E+00**	**0.000E+00**	8.000E-159	**0.000E+00**	**0.000E+00**	**0.000E+00**
F23	med	8.104E+00	1.131E+00	3.483E+02	**7.123E-02**	1.995E+01	5.285E-01	2.209E-01	1.264E+00	2.016E+00	1.140E-01
	mean	2.883E+01	1.474E+00	5.124E+02	**1.843E-01**	2.154E+01	7.276E+00	3.981E-01	1.487E+00	2.164E+00	4.390E-01
	std	5.195E+01	1.450E+00	4.699E+02	**2.279E-01**	1.324E+01	1.576E+01	4.890E-01	1.323E+00	2.010E+00	9.230E-01

Table (11) provides a comprehensive summary of the YGSA's performance, compared to the other algorithms discussed, for the 23 test functions based on the med criterion. In this table, the symbol "≤" indicates equal performance among the algorithms in terms of finding the optimal solution (minimum) for each function. Conversely, the symbol "<" signifies the superiority of one algorithm over another in finding the optimal solution.

Table 11. Overall performance in unimodal test functions F1-F23

Test functions		**Comparison of algorithms based on median (med)**
Unimodal fix-dimension test functions	F1	YGSA ≤ WSO ≤ HBO ≤ EBO-CMAR ≤ CS ≤ PSO < MFO < GSA < MVO < SCA
	F2	WSO ≤ HBO ≤ EBO-CMAR ≤ CS ≤ PSO ≤ MFO < GSA < MVO < YGSA < SCA
	F3	YGSA ≤ WSO ≤ HBO ≤ EBO-CMAR ≤ CS ≤ PSO ≤ MFO ≤ MVO ≤ SCA < GSA
	F4	WSO < MFO < SCA < HBO < PSO < YGSA < CS < MVO < GSA

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

	F5	YGSA ≤ WSO ≤ HBO ≤ EBO-CMAR ≤ CS ≤ PSO ≤ MFO ≤ GSA ≤ MVO ≤ SCA
	F6	YGSA ≤ WSO ≤ HBO ≤ EBO-CMAR ≤ CS ≤ PSO ≤ MFO ≤ GSA ≤ MVO ≤ SCA
	F7	WSO ≤ CS < EBO-CMAR < HBO < MVO < PSO < SCA < YGSA < MFO < GSA
	F8	YGSA ≤ WSO ≤ HBO ≤ EBO-CMAR ≤ CS ≤ PSO ≤ MFO ≤ GSA ≤ MVO ≤ SCA
	F9	YGSA ≤ WSO ≤ HBO ≤ EBO-CMAR ≤ CS ≤ PSO ≤ MFO ≤ GSA ≤ MVO ≤ SCA
Unimodal variable-dimension test functions	F10	WSO < YGSA < HBO < EBO-CMAR < GSA < PSO < MFO < SCA < MVO
	F11	YGSA ≤ WSO < HBO < EBO-CMAR < MFO < PSO < GSA < CS < SCA <MVO
	F12	YGSA < WSO < HBO < GSA < EBO-CMAR < PSO < SCA < CS < MVO < MFO
	F13	WSO < YGSA < GSA < EBO-CMAR < PSO < MVO < HBO < CS < SCA < MFO
	F14	HBO < GSA < EBO-CMAR < PSO < MFO < WSO < CS < MVO < SCA < YGSA
	F15	YGSA ≤ WSO ≤ HBO ≤ CS ≤ MFO < MVO < PSO < EBO-CMAR < GSA < SCA
	F16	YGSA < WSO < MVO < HBO < SCA < PSO < EBO-CMAR < GSA < MFO < CS
	F17	YGSA < WSO < GSA < HBO < EBO-CMAR < PSO < MFO <MVO < CS < SCA
	F18	WSO < PSO < GSA < YGSA < SCA < CS MVO < MFO < HBO < EBO-CMAR
	F19	WSO < YGSA < HBO < GSA < EBO-CMAR < SCA < CS < MVO < MFO <PSO

Power System Technology
ISSN:1000-3673

Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

	F20	YGSA ≤ WSO ≤ HBO ≤ EBO-CMAR ≤ GSA < PSO < CS < SCA < MVO < MFO
	F21	WSO < YGSA < EBO-CMAR < HBO < CS < GSA < SCA < MVO < MFO < PSO
	F22	YGSA < HBO ≤ PSO ≤ MFO ≤ CS ≤WSO ≤ EBO-CMAR < SCA < MVO < GSA
	F23	PSO < WSO < MVO < MFO < HBO < CS < EBO-CMAR < YGSA < SCA < GSA

Analyzing the outcomes presented in Table (11), it becomes evident that the proposed YGSA (highlighted in red) demonstrates the best performance in finding the optimal solution for functions F1, F3, F5, F6, F8, and F9 within the unimodal fixed-dimension test functions category. However, it exhibits subpar performance for functions F2, F4, and F7. For instance, in function F1, the YGSA achieves comparable performance to the WSO and HBO algorithms, while outperforming only the CS, MVO, and SCA algorithms in function F4.

Within the unimodal variable-dimension test functions category, the proposed YGSA displays superior performance compared to other algorithms for functions F11, F12, F15, F16, F17, F20, and F22. However, it demonstrates inferior performance for functions F10, F13, F14, F18, F19, F21, and F23. For instance, the YGSA outperforms other algorithms significantly in finding the optimal solution for functions F16 and F17 but exhibits weaker performance in functions F14 and F18. It is worth mentioning that the YGSA performs worse than the WSO algorithm only for functions F10, F13, and F21, while achieving better results than the other compared algorithms in the remaining cases.

b. Evaluation of multimodal test functions

This section focuses on evaluating the overall performance of the proposed YGSA and other compared algorithms on multimodal benchmark test functions. Similar to the previous category, this type of functions consists of two subsets: multimodal fixed-dimension test functions and multimodal variable-dimension test functions. Functions F24 to F46, as listed in Tables (5) and (6), belong to this category. For each function, the med, mean, and std parameters are computed for all compared algorithms to assess their performance. The results obtained from this evaluation

Power System Technology
ISSN:1000-3673

Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

are presented in Tables (12) and (13), with the optimal results for each algorithm being highlighted in bold.

Table 12. Comparison results for 10 multimodal fixed-dimension test functions

F. No	State	YGSA	HBO	GSA	PSO	SCA	MFO	MVO	CS	EBO-CMAR	WSO
F24	med	**0.000E+00**	**0.000E+00**	8.120E-20	1.000E-129	4.400E-158	**0.000E+00**	2.260E-07	7.730E-52	**0.000E+00**	**0.000E+00**
	mean	1.285E-17	**0.000E+00**	1.090E-19	7.800E+00	3.500E-150	**0.000E+00**	2.940E-07	1.090E-49	**0.000E+00**	**0.000E+00**
	std	2.874E-16	**0.000E+00**	1.160E-19	1.700E-127	1.700E-149	**0.000E+00**	2.600E-07	1.910E-49	**0.000E+00**	**0.000E+00**
F25	med	**-1.956E+02**	**-1.956E+02**	**-1.956E+02**	**-1.956E+02**	**-1.956E+02**	**-1.956E+02**	**-1.956E+02**	**-1.956E+02**	**-1.956E+02**	**-1.956E+02**
	mean	**-1.956E+02**	**-1.956E+02**	**-1.956E+02**	**-1.956E+02**	**-1.956E+02**	**-1.956E+02**	**-1.956E+02**	**-1.956E+02**	**-1.956E+02**	**-1.956E+02**
	std	7.192E-05	5.780E-14	5.800E-14	5.800E-14	6.380E-05	5.800E-14	2.900E-06	5.800E-14	**3.000E-14**	**3.000E-14**
F26	med	**-2.022E+00**	**-2.022E+00**	**-2.022E+00**	**-2.022E+00**	**-2.022E+00**	**-2.022E+00**	**-2.022E+00**	**-2.022E+00**	**-2.022E+00**	**-2.022E+00**
	mean	**-2.022E+00**	**-2.022E+00**	**-2.022E+00**	**-2.022E+00**	**-2.022E+00**	**-2.022E+00**	**-2.022E+00**	**-2.022E+00**	**-2.022E+00**	**-2.022E+00**
	std	1.606E-05	1.360E-15	6.540E-04	1.360E-15	7.600E-10	1.360E-15	3.490E-11	1.360E-15	**4.680E-16**	**4.680E-16**
F27	med	**-1.068E+02**	**-1.068E+02**	**-1.068E+02**	**-1.068E+02**	**-1.068E+02**	**-1.068E+02**	**-1.068E+02**	**-1.068E+02**	**-1.068E+02**	**-1.068E+02**
	mean	**-1.068E+02**	**-1.068E+02**	**-1.066E+02**	**-1.068E+02**	**-1.067E+02**	**-1.068E+02**	**-1.068E+02**	**-1.068E+02**	**-1.068E+02**	**-1.068E+02**

Power System Technology
ISSN:1000-3673

Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

	std	1.079E-02	2.950E-14	5.564E-01	8.700E-15	1.946E-02	2.800E-14	2.350E-06	**4.100E-15**	2.720E-14	9.470E-15
F28	med	**-1.032E+00**	**-1.032E+00**	**-1.032E+00**	**-1.032E+00**	**-1.032E+00**	**-1.032E+00**	**-1.032E+00**	**-1.032E+00**	**-1.032E+00**	**-1.032E+00**
	mean	**-1.032E+00**	**-1.032E+00**	**-1.032E+00**	**-1.032E+00**	**-1.032E+00**	**-1.032E+00**	**-1.032E+00**	**-1.032E+00**	**-1.032E+00**	**-1.032E+00**
	std	**0.000E+00**	6.710E-16	5.550E-16	6.720E-16	1.290E-05	6.800E-16	5.750E-08	6.800E-16	7.400E-17	**0.000E+00**
F29	med	**3.979E-01**	**3.979E-01**	**3.979E-01**	**3.979E-01**	**3.979E-01**	**3.979E-01**	**3.979E-01**	**3.979E-01**	**3.979E-01**	**3.979E-01**
	mean	**3.979E-01**	**3.979E-01**	**3.979E-01**	**3.979E-01**	**3.979E-01**	**3.979E-01**	**3.979E-01**	**3.979E-01**	**3.979E-01**	**3.979E-01**
	std	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**	3.390E-04	**0.000E+00**	2.580E-08	**0.000E+00**	**0.000E+00**	**0.000E+00**
F30	med	**-3.863E+00**	**-3.863E+00**	**-3.863E+00**	**-3.863E+00**	**-3.855E+00**	**-3.863E+00**	**-3.863E+00**	**-3.863E+00**	**-3.863E+00**	**-3.863E+00**
	mean	-3.855E+00	-3.863E+00	-3.863E+00	**-3.862E+00**	-3.856E+00	-3.863E+00	-3.863E+00	-3.863E+00	-3.863E+00	-3.863E+00
	std	1.370E-02	2.710E-15	1.990E-15	1.576E-03	2.786E-03	2.270E-15	3.020E-07	2.270E-15	**9.360E-16**	1.320E-15
F31	med	-3.313E+00	-3.322E+00	-3.322E+00	-3.200E+00	**-3.075E+00**	-3.203E+00	-3.322E+00	-3.322E+00	-3.322E+00	-3.322E+00
	mean	-3.259E+00	-3.322E+00	-3.322E+00	-3.221E+00	**-3.032E+00**	-3.238E+00	-3.265E+00	-3.322E+00	-3.298E+00	-3.32E+00
	std	5.668E-02	1.360E-15	**4.530E-16**	1.130E-01	1.455E-01	6.013E-02	6.075E-02	4.530E-16	5.013E-02	9.00E-16
F32	med	**-2.063E+00**	**-2.063E+00**	**-2.063E+00**	**-2.063E+00**	**-2.063E+00**	**-2.063E+00**	**-2.063E+00**	**-2.063E+00**	**-2.063E+00**	**-2.063E+00**

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

	mean	**-2.063E+00**	**-2.063E+00**	**-2.063E+00**	**-2.063E+00**	**-2.063E+00**	**-2.063E+00**	**-2.063E+00**	**-2.063E+00**	**-2.063E+00**	**-2.063E+00**
	std	2.700E-05	9.030E-16	9.060E-16	9.060E-16	5.310E-06	9.030E-16	6.140E-09	9.060E-16	**4.680E-16**	**4.680E-16**
F33	med	**1.000E+00**	**1.000E+00**	1.000004	**1.000E+00**	**1.000E+00**	**1.000E+00**	1.003E+00	**1.000E+00**	**1.000E+00**	**1.000E+00**
	mean	**1.000E+00**	**1.000E+00**	1.000010	**1.000E+00**	**1.000E+00**	**1.000E+00**	1.003E+00	**1.000E+00**	**1.000E+00**	**1.000E+00**
	std	**1.817E-07**	0.000E+00	1.310E-05	0.000E+00	0.000E+00	0.000E+00	1.642E-03	0.000E+00	0.000E+00	4.900E-03

Table 13. Comparison results for 13 multimodal variable-dimension test functions

F. No	State	YGSA	HBO	GSA	PSO	SCA	MFO	MVO	CS	EBO-CMAR	WSO
F34	med	2.491E+02	**1.270E-05**	3.278E+02	1.994E+02	2.808E+02	1.344E+02	1.529E+02	1.249E+02	1.738E+02	**1.270E-05**
	mean	2.576E+02	2.369E+00	3.257E+02	1.971E+02	2.803E+02	1.330E+02	1.537E+02	1.213E+02	1.978E+02	**1.430E-05**
	std	2.068E+01	3.680E+00	1.433E+01	3.456E+01	9.926E+00	2.269E+01	1.960E+01	1.569E+01	2.493E+01	**4.040E-06**
F35	med	**0.000E+00**	1.492E+00	1.642E+01	8.059E+01	3.603E+00	1.572E+02	9.259E+01	6.633E+01	1.194E+01	**0.000E+00**
	mean	5.819E-10	1.758E+00	1.741E+01	7.983E+01	1.019E+01	1.576E+02	9.653E+01	6.689E+01	2.326E+01	**0.000E+00**
	std	1.301E-08	1.244E+00	4.115E+00	2.180E+01	1.453E+01	3.324E+01	2.712E+01	1.104E+01	5.032E+00	**0.000E+00**
F36	med	**9.000E-01**	1.243E+00	1.000E+00	2.776E+00	4.207E+00	4.256E+00	1.002E+00	1.170E+00	1.000E+00	**9.000E-01**
	mean	**9.000E-01**	1.225E+00	1.000E+00	2.776E+00	4.532E+00	4.113E+00	1.002E+00	1.163E+00	1.000E+00	**9.000E-01**
	std	1.196E-14	1.096E-01	**1.020E-16**	9.552E-01	1.744E+00	1.098E+00	4.560E-04	2.505E-02	1.240E-13	1.170E-16

Power System Technology
ISSN:1000-3673

Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

F37	med	3.558E+03	7.980E-11	1.592E+06	**6.290E-11**	5.112E+03	3.030E-06	3.466E+02	1.000E+10	4.850E-07	3.660E+03
	mean	3.947E+03	4.960E-07	2.194E+06	**1.330E-09**	4.494E+04	1.290E-05	3.489E+02	1.000E+10	4.140E-04	3.830E+03
	std	8.873E+02	1.350E-07	2.193E+06	5.360E-09	1.651E+05	2.730E-05	1.076E+02	**0.000E+00**	1.591E-03	6.020E+02
F38	med	**1.610E-261**	1.870E-12	2.330E-09	5.970E-06	5.094E-03	4.440E+00	3.689E+00	4.801E+00	3.520E-06	2.220E-261
	mean	3.990E-08	7.030E-09	2.380E-09	3.554E-01	1.910E-02	7.647E+00	3.758E+00	5.487E+00	9.630E-06	**2.300E-259**
	std	8.919E-07	2.040E-08	4.630E-10	1.229E+00	4.792E-02	7.697E+00	1.352E+00	1.712E+00	1.440E-05	**0.000E+00**
F39	med	1.054E-63	5.280E-22	2.860E-06	5.074E+04	1.280E-05	3.671E+05	3.315E-02	7.755E-01	3.100E-11	**3.150E-70**
	mean	8.626E-11	1.670E-19	1.128E-03	1.928E+06	2.280E-04	4.481E+06	2.602E+00	8.208E+00	5.630E-11	**7.770E-39**
	std	1.929E-09	4.770E-19	3.176E-03	6.519E+06	4.560E-04	1.085E+07	1.011E+01	1.912E+01	8.560E-11	**2.460E-38**
F40	med	**-8.880E-16**	2.220E-14	3.480E-09	1.400E-06	1.860E+01	1.899E+01	6.509E-01	1.936E+00	1.080E-05	**-8.880E-16**
	mean	1.655E-06	2.730E-14	3.430E-09	3.950E-06	1.393E+01	1.412E+01	7.353E-01	1.701E+00	1.100E-05	**-8.880E-16**
	std	3.698E-05	1.470E-14	6.890E-10	6.030E-06	7.875E+00	8.439E+00	6.198E-01	1.005E+00	4.050E-08	**0.000E+00**
F41	med	8.753E+01	**1.000E+00**	5.152E+03	5.933E+00	1.039E+02	1.213E+02	1.820E+02	1.293E+02	5.485E+00	**1.00E+00**
	mean	8.759E+01	**1.000E+00**	6.566E+03	8.117E+00	1.074E+02	6.013E+04	1.822E+02	1.318E+02	8.300E+00	**1.00E+00**
	std	1.091E+00	2.140E-16	4.248E+03	4.973E+00	2.074E+01	1.307E+05	2.756E+01	2.777E+01	4.818E+00	**4.90E-03**
F42	med	**0.000E+00**	1.999E-01	1.309E+00	2.999E-01	2.000E-01	4.100E+00	4.999E-01	1.400E+00	1.999E-01	4.090E-106

Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

	mean	2.871E-09	2.299E-01	1.249E+00	3.479E-01	2.561E-01	6.072E+00	5.479E-01	1.379E+00	1.799E-01	**2.730E-42**
	std	6.418E-08	4.661E-02	2.330E-01	6.532E-02	9.621E-02	3.702E+00	9.626E-02	2.680E-01	4.140E-02	**8.650E-42**
F43	med	**0.000E+00**	**0.000E+00**	**0.000000**	1.232E-02	1.590E-03	3.920E-02	3.467E-02	1.853E-03	1.960E-13	**0.000E+00**
	mean	1.830E-16	7.400E-18	4.930E-04	1.113E-02	1.154E-01	7.583E-01	3.440E-02	5.808E-03	2.090E-13	**0.000E+00**
	std	4.091E-15	2.820E-19	2.206E-03	6.440E-03	2.116E-01	1.383E+00	9.681E-03	9.444E-03	1.190E-13	**0.000E+00**
F44	med	**-1.000E+00**	8.450E-22	6.670E-30	9.390E-25	1.590E-10	1.200E-13	6.450E-16	6.710E-14	4.800E-13	**-1.000E+00**
	mean	**-1.000E+00**	7.230E-20	6.900E-30	1.880E-14	1.690E-10	1.070E-13	6.520E-16	7.010E-14	4.260E-13	**-1.000E+00**
	std	**0.000E+00**	2.050E-19	1.900E-30	6.500E-14	8.180E-11	1.130E-13	1.680E-16	3.240E-14	1.820E-13	**0.000E+00**
F45	med	1.544E-10	5.290E-12	3.510E-12	3.060E-11	3.470E-10	2.710E-11	1.790E-11	1.590E-11	1.900E-11	**1.210E-20**
	mean	2.121E-09	5.310E-12	3.530E-12	3.060E-11	3.820E-10	2.610E-11	2.330E-11	1.540E-11	9.870E-11	**1.220E-20**
	std	6.058E-09	7.410E-13	5.330E-14	1.340E-12	1.950E-10	3.030E-12	1.510E-11	3.680E-12	3.960E-12	**2.520E-22**
F46	med	2.337E+00	**1.250E-28**	2.350E-18	4.920E-12	2.378E+00	1.099E-02	3.432E-02	3.920E+00	5.660E-15	9.050E-08
	mean	2.499E+00	**1.170E-27**	5.490E-04	4.395E-03	3.028E+00	1.640E+07	3.924E-02	5.786E+00	2.180E-14	2.320E-07
	std	1.430E-01	**3.630E-27**	2.457E-03	5.494E-03	1.743E+00	8.201E+07	1.937E-02	5.488E+00	3.400E-14	2.720E-07

A summary of the performance of the proposed YGSA compared to the other algorithms under consideration is provided in Table (14), based on the med criterion, for the 23 test functions. In this table, the symbol "≤" indicates comparable performance among the algorithms in finding the

Power System Technology
ISSN:1000-3673

Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

optimal solution for each function, while the symbol "<" denotes the inferior performance of one algorithm relative to another.

Table 14. Overall performance in multimodal test functions F24-F46

Test functions		Comparison of algorithms based on median (med)
Multimodal fix-dimension test functions	F24	YGSA ≤ WSO ≤ HBO ≤ EBO-CMAR ≤ MFO < SCA < PSO < CS < GSA < MVO
	F25	YGSA ≤ WSO ≤ HBO ≤ EBO-CMAR ≤ CS ≤ PSO ≤ MFO ≤ GSA ≤ MVO ≤ SCA
	F26	YGSA ≤ WSO ≤ HBO ≤ EBO-CMAR ≤ CS ≤ PSO ≤ MFO ≤ GSA ≤ MVO ≤ SCA
	F27	YGSA ≤ WSO ≤ HBO ≤ EBO-CMAR ≤ CS ≤ PSO ≤ MFO ≤ GSA ≤ MVO ≤ SCA
	F28	YGSA ≤ WSO ≤ HBO ≤ EBO-CMAR ≤ CS ≤ PSO ≤ MFO ≤ GSA ≤ MVO ≤ SCA
	F29	YGSA ≤ WSO ≤ HBO ≤ EBO-CMAR ≤ CS ≤ PSO ≤ MFO ≤ GSA ≤ MVO ≤ SCA
	F30	YGSA ≤ WSO ≤ HBO ≤ EBO-CMAR ≤ CS ≤ PSO ≤ MFO ≤ GSA ≤ MVO ≤ SCA
	F31	SCA < PSO < MFO < YGSA ≤ WSO ≤ HBO ≤ CS ≤ EBO-CMAR ≤ GSA
	F32	YGSA ≤ WSO ≤ HBO ≤ EBO-CMAR ≤ CS ≤ PSO ≤ MFO ≤ GSA ≤ MVO ≤ SCA
	F33	YGSA ≤ WSO ≤ HBO ≤ EBO-CMAR ≤ CS ≤ PSO ≤ MFO ≤ GSA ≤ MVO ≤ SCA
	F34	WSO ≤ HBO < CS < MFO < MVO < EBO-CMAR < PSO < YGSA < SCA < GSA
Mul timod	F35	YGSA ≤ WSO < HBO < SCA < EBO-CMAR < GSA < CS < PSO < MVO < MFO

Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

	F36	YGSA ≤ WSO < GSA < EBO-CMAR < MVO < CS < HBO < PSO < SCA < MFO
	F37	PSO < HBO < EBO-CMAR < MFO < MVO < YGSA < WSO < SCA < GSA < CS
	F38	YGSA < WSO < HBO < GSA < EBO-CMAR < PSO < SCA < MVO< MFO < CS
	F39	WSO < YGSA < HBO < EBO-CMAR < GSA < SCA < MVO < CS < PSO < MFO
	F40	YGSA < WSO < HBO < GSA < PSO < EBO-CMAR < MVO < CS < CSA < MFO
	F41	WSO ≤ HBO < EBO-CMAR < PSO < YGSA < SCA < MFO < CS < MVO < GSA
	F42	YGSA < WSO < HBO < EBO-CMAR < SCA < PSO < MVO < GSA < CS < MFO
	F43	YGSA ≤ WSO ≤ HBO < GSA < EBO-CMAR < SCA < CS < PSO < MVO < MFO
	F44	YGSA ≤ WSO < GSA < PSO < HBO < MVO < CS < MFO < EBO-CMAR < SCA
	F45	WSO < GSA < HBO < CS < MVO < EBO-CMAR < MFO < PSO < YGSA < SCA
	F46	HBO < SGA < EBO-CMAR < PSO < WSO < MFO < MVO < YGSA < SCA < CS

Upon analyzing the results presented in Table (14), it becomes apparent that the proposed YGSA demonstrates favorable performance in finding the optimal solution for multimodal fixed-dimension test functions F24, F25, F26, F27, F28, F29, F30, F32, and F33. It performs on par with other algorithms, except for function F31, where it exhibits weaker performance compared to the SCA, PSO, and MFO algorithms.

Within the multimodal variable-dimension test functions category, the YGSA shows weaker performance compared to the WSO algorithm for functions F35, F36, F43, and F44. However, it achieves the best performance among all the compared algorithms for test functions F34, F40, and

Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

F42. Nevertheless, it fails to deliver satisfactory performance for functions F37, F41, F45, and F46.

c. Evaluation of CEC-06-2019 test functions

In this section, the evaluation of YGSA has been conducted using cec-06-2019 test functions, which represent the latest collection of pertinent test functions. Widely recognized for their usage in testing and assessing the performance of diverse algorithms, the CEC-06 collection comprises 10 test functions denoted as CEC01 to CEC10 [76, 77, 78]. These functions possess a high degree of scalability, with rotational and dynamic properties being inherent in all but CEC01 to CEC03. The CEC-06-2019 functions are designed to introduce varied shapes within different regions of the search space. Specifically, the functions CEC04 to CEC10 are ten-dimensional in nature, with problem variables defined in the range of [-100, 100]. Table (7) provides detailed specifications of these functions at the outset of this section. The evaluation results obtained for the compared algorithms are presented in Table (15).

Table 15. Comparison results for 10 CEC-06-2019 test functions

F. No	State	YGSA	HBO	GSA	PSO	SCA	MFO	MVO	CS	EBO-CMAR	WSO
CEC01	med	**2.799E+04**	3.860E+04	1.880E+09	1.350E+07	1.470E+10	3.990E+04	2.520E+08	3.310E+10	7.120E+05	3.360E+04
	mean	**3.855E+04**	5.610E+05	8.740E+09	1.010E+09	9.050E+10	1.740E+08	1.155E+10	2.330E+11	6.230E+10	1.450E+05
	std	7.737E+03	9.900E+00	7.650E+08	1.730E+18	3.200E+21	3.330E+17	1.820E+20	1.570E+21	4.190E+21	**3.100E+00**
CEC02	med	**1.797E+00**	1.830E+00	1.830E+00	1.830E+00	1.830E+00	1.830E+00	1.830E+00	1.830E+00	1.830E+00	1.830E+00
	mean	**1.797E+00**	1.830E+00	1.830E+00	1.830E+00	2.020E+00	1.830E+00	1.840E+00	1.830E+00	1.830E+00	1.830E+00
	std	1.732E-02	3.540E-18	1.360E-17	1.790E-18	8.200E-02	1.740E-19	1.450E-04	1.170E-19	4.180E-06	**2.760E-20**

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

	med	**1.270E+01**	**1.270E+01**	**1.270E+01**	**1.270E+01**	**1.270E+01**	**1.270E+01**	**1.270E+01**	**1.270E+01**	**1.270E+01**	**1.270E+01**
CEC03	mean	**1.270E+01**	**1.270E+01**	**1.270E+01**	**1.270E+01**	**1.270E+01**	**1.270E+01**	**1.270E+01**	**1.270E+01**	**1.270E+01**	**1.270E+01**
	std	2.589E-06	7.750E-27	5.360E-12	3.180E-26	3.580E-10	3.410E-14	7.140E-07	4.150E-07	4.030E-11	**5.230E-27**
	med	9.420E+02	6.090E+00	8.560E+00	4.970E+00	3.255E+01	4.210E+00	1.303E+01	1.238E+03	1.224E+02	**3.005E+00**
CEC04	mean	1.323E+03	1.347E+01	3.322E+01	2.343E+01	6.006E+01	**1.557E+01**	4.030E+01	3.282E+03	7.692E+02	1.892E+01
	std	7.115E+02	3.124E+01	2.117E+02	1.182E+02	1.858E+02	8.721E+01	3.331E+02	**2.530E+00**	4.686E+02	5.949E+01
	med	**1.040E+00**	1.081E+00	1.200E+00	1.110E+00	1.340E+00	1.140E+00	1.260E+00	2.100E+00	1.970E+00	1.109E+00
CEC05	mean	1.866E+00	1.090E+00	1.230E+00	1.120E+00	1.330E+00	**1.060E+00**	1.290E+00	2.530E+00	2.005E+00	1.120E+00
	std	2.369E-01	5.000E-03	7.100E-02	4.000E-03	8.000E-03	3.000E-03	2.100E-02	5.000E-02	1.400E-01	**2.000E-03**
	med	8.450E+00	**2.820E+00**	7.310E+00	9.440E+00	8.500E+00	1.116E+01	6.880E+00	1.140E+01	1.140E+01	3.730E+00
CEC06	mean	8.740E+00	**2.960E+00**	7.380E+00	9.050E+00	8.430E+00	1.106E+01	6.910E+00	1.130E+01	9.930E+00	3.770E+00
	std	4.905E-01	1.060E+00	**2.900E-01**	2.650E+00	3.400E-01	6.900E-01	2.220E+00	4.800E-01	9.980E+00	2.001E+00
CEC07	med	2.846E+02	1.048E+02	6.596E+02	1.486E+02	2.787E+02	2.428E+02	3.403E+02	8.887E+02	6.735E+02	**6.560E+01**
	mean	4.633E+02	1.062E+02	6.715E+02	1.498E+02	2.801E+02	2.706E+02	3.590E+02	9.031E+02	6.930E+02	**5.735E+01**

Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

	std	**1.778E+02**	1.662E+04	1.054E+04	1.919E+04	8.155E+03	2.956E+04	6.152E+04	3.323E+04	2.956E+04	1.318E+04
	med	5.044E+00	4.450E+00	5.530E+00	5.300E+00	5.780E+00	**4.170E+00**	5.610E+00	6.660E+00	6.280E+00	4.42E+00
CEC08	mean	5.122E+00	4.440E+00	5.760E+00	5.190E+00	5.720E+00	4.050E+00	5.520E+00	6.630E+00	6.230E+00	**4.29E+00**
	std	2.459E-01	2.200E-01	4.300E-01	7.000E-01	1.300E-01	1.220E+00	3.800E-01	**1.000E-01**	2.900E-01	3.60E-01
	med	1.104E+02	2.850E+00	3.130E+00	**2.450E+00**	3.170E+00	**2.450E+00**	2.810E+00	8.257E+02	6.570E+00	2.570E+00
CEC09	mean	1.793E+02	2.960E+00	3.110E+00	**2.460E+00**	3.190E+00	2.490E+00	2.920E+00	9.516E+02	1.703E+01	2.620E+00
	std	9.516E+01	1.800E-01	4.100E-02	4.000E-03	6.000E-02	**1.000E-02**	1.200E-01	1.712E+02	9.176E+02	3.700E-02
	med	**1.934E+01**	2.000E+01	1.989E+01	2.018E+01	2.016E+01	2.048E+01	2.000E+01	2.057E+01	2.035E+01	2.000E+01
CEC010	mean	1.938E+01	1.834E+01	1.873E+01	2.002E+01	2.017E+01	2.028E+01	2.003E+01	2.054E+01	2.035E+01	**1.789E+01**
	std	2.124E-01	2.863E+01	3.544E+01	4.120E+00	**1.000E-04**	3.550E+00	4.000E-03	9.000E-03	1.600E-02	3.333E+01

In line with the previous test function collections, Table (16) offers a comprehensive overview of the performance of the proposed YGSA relative to other algorithms within the CEC-06-2019 test function collection. Analyzing the outcomes from Tables (15) and (16) reveals that YGSA demonstrates superior performance in CEC01, CEC02, CEC05, and CEC10. It also exhibits performance on par with other algorithms in CEC03, while comparatively weaker results are observed in CEC04 and CEC09. Notably, based on the findings in Table (16), the recently compared WSO algorithm attains the best performance according to the med criterion for optimizing the CEC-06-2019 collection, following YGSA. Specifically, WSO outperforms other algorithms in CEC01 and CEC02 after YGSA, and it also displays the most favorable performance in CEC07. These conclusions align with the observations made in Tables (11) and (14) for the

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

previous two test function collections. Consequently, it can be concluded that both YGSA and WSO exhibit superior performance compared to other algorithms.

Table 16. Overall performance in CEC-06-2019 test functions F46-F56

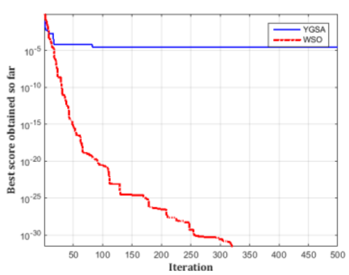
CEC-06-2019	**Comparison of algorithms based on median (med) value**
CEC01	YGSA < WSO < HBO < MFO < EBO-CMAR < PSO < MVO < GSA < SCA < CS
CEC02	YGSA < WSO ≤ HBO ≤ MFO ≤ EBO-CMAR ≤ PSO ≤ MVO ≤ GSA ≤ SCA ≤ CS
CEC03	YGSA ≤ WSO ≤ HBO ≤ EBO-CMAR ≤ CS ≤ PSO ≤ MFO ≤ MVO ≤ SCA < GSA
CEC04	WSO < MFO < PSO < HBO < GSA < MVO < SCA < EBO-CMAR < YGSA < CS
CEC05	YGSA < HBO < WSO < PSO < MFO < MVO < GSA < SCA < EBO-CMAR < CS
CEC06	HBO < WSO < MVO < GSA < GSA < YGSA < PSO < MFO < EBO-CMAR < CS
CEC07	WSO < CS < EBO-CMAR < HBO < MVO < PSO < SCA < YGSA < MFO < GSA
CEC08	MFO < WSO < HBO < YGSA < PSO < GSA < MVO < SCA < EBO-CMAR < CS
CEC09	MFO ≤ PSO < WSO < MVO < HBO < GSA < SCA < EBO-CMAR < CS < YGSA
CEC10	YGSA < GSA < WSO ≤ HBO ≤ MVO < SCA < PSO < EBO-CMAR < MFO < CS

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

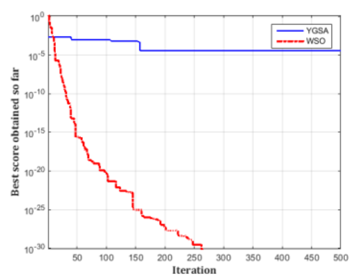
d. Convergence curve analysis

In order to evaluate the convergence rate of the YGSA, a series of experiments were conducted. The performance of YGSA was compared with that of nine other algorithms, utilizing 56 test functions. Each comparison involved an average of 30 runs, with 500 iterations per run. However, in the context of this section, our focus is solely on comparing the convergence rate between the YGSA and the Whale Swarm Optimization (WSO), as the WSO algorithm demonstrated superior performance when compared to the other eight algorithms. Figures (10) to (14) depict the convergence rates of YGSA and WSO for three test function sets, encompassing a total of 56 functions.

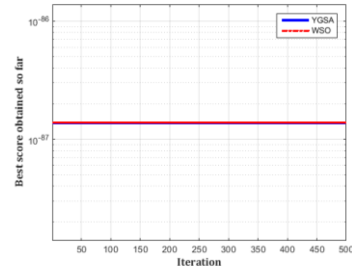
F1



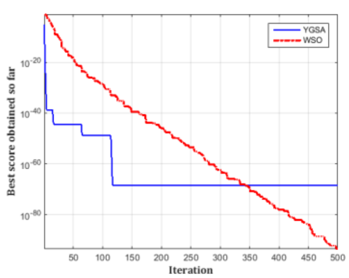
F2



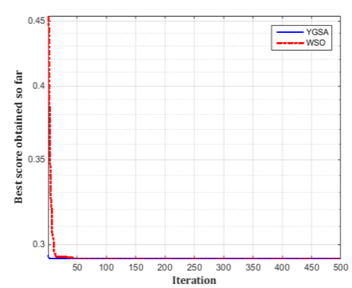
F3



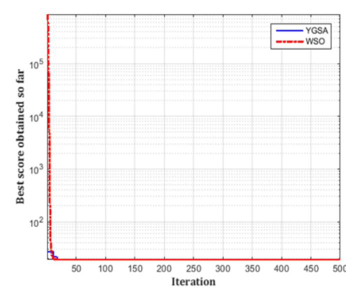
F4



F5



F6



Power System Technology
ISSN:1000-3673

Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

F7 F8 F9

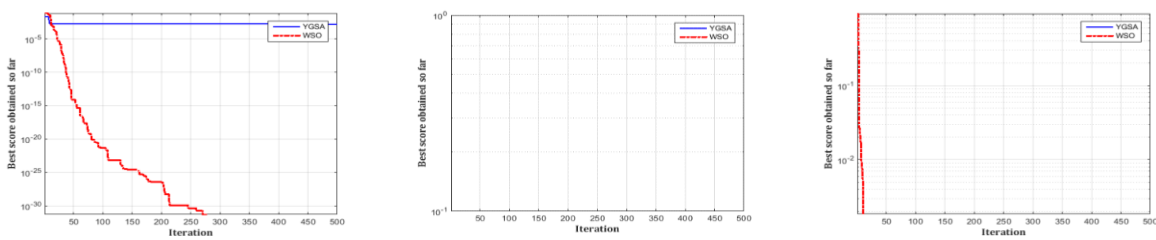
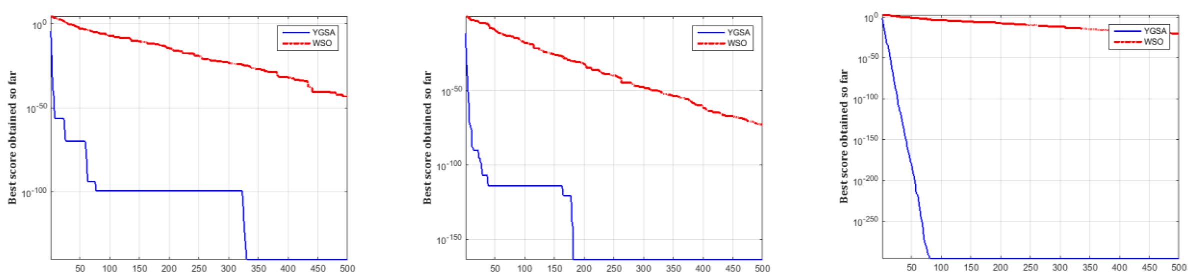


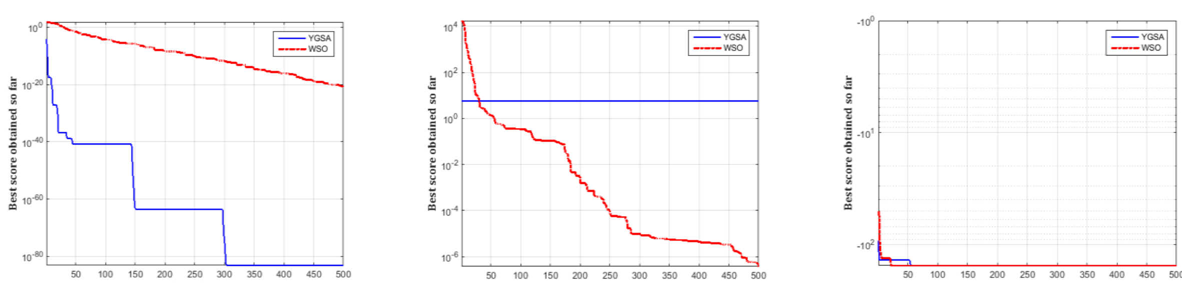
Figure 10. Unimodal fixed-dimension F1-F9 convergence rate plot

Figure (10) illustrates the convergence rate comparison after 500 iterations between the proposed YGSA and WSO for the unimodal fixed-dimension test function set. From Figure (10), it is evident that YGSA demonstrates faster convergence rates than WSO for functions F5 and F6. Additionally, both algorithms exhibit comparable convergence rates for functions F3, F8, and F9. However, YGSA shows slower convergence rates compared to WSO for functions F1, F2, F4, and F7. Based on the observation that the YGSA demonstrates faster convergence rates for two functions and comparable convergence rates for four functions, it can be inferred that YGSA exhibits superior convergence rates compared to the WSO within this specific set of test functions.

F10 F11 F12



F13 F14 F15



Power System Technology
ISSN:1000-3673

Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

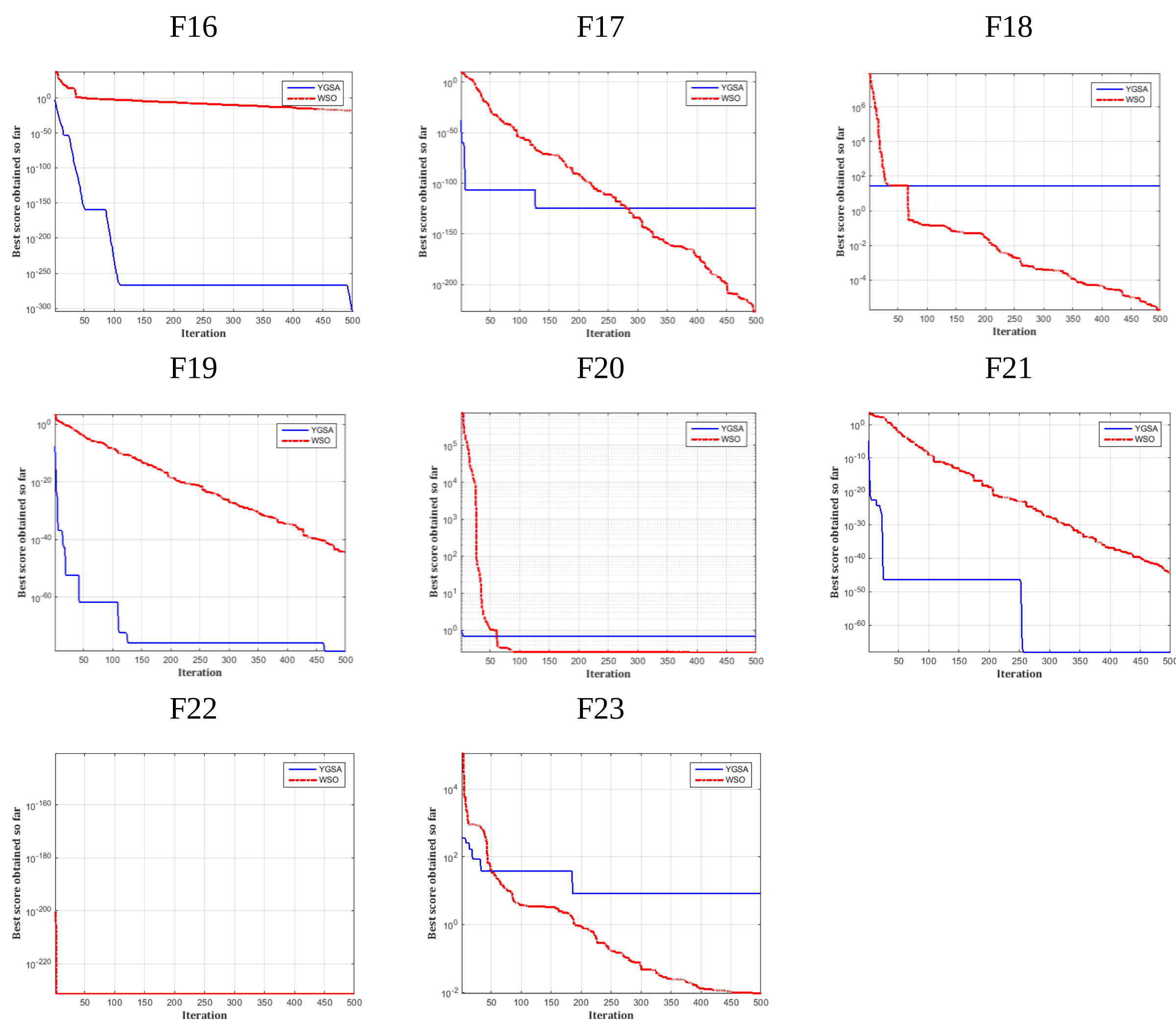


Figure 11. Unimodal variable-dimension F10-F23 convergence rate plot

Figure (11) presents the convergence rate comparison between YGSA and WSO for the unimodal variable-dimension test function set. As depicted in Figure (11), YGSA achieves faster convergence rates than WSO for functions F10, F11, F12, F13, F16, F19, and F21. Conversely, WSO demonstrates faster convergence rates for functions F14, F15, F17, F18, F20, F22, and F23.

Figure (12) illustrates the comparison of convergence rates between the YGSA and the WSO specifically for the multimodal fixed-dimension test function set. Based on the results, YGSA performs better than WSO for functions F24, F27, and F33, while WSO outperforms YGSA for

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

function F31. However, both algorithms display comparable convergence rates for functions F25, F26, F28, F30, and F32 in terms of finding the optimal solution.

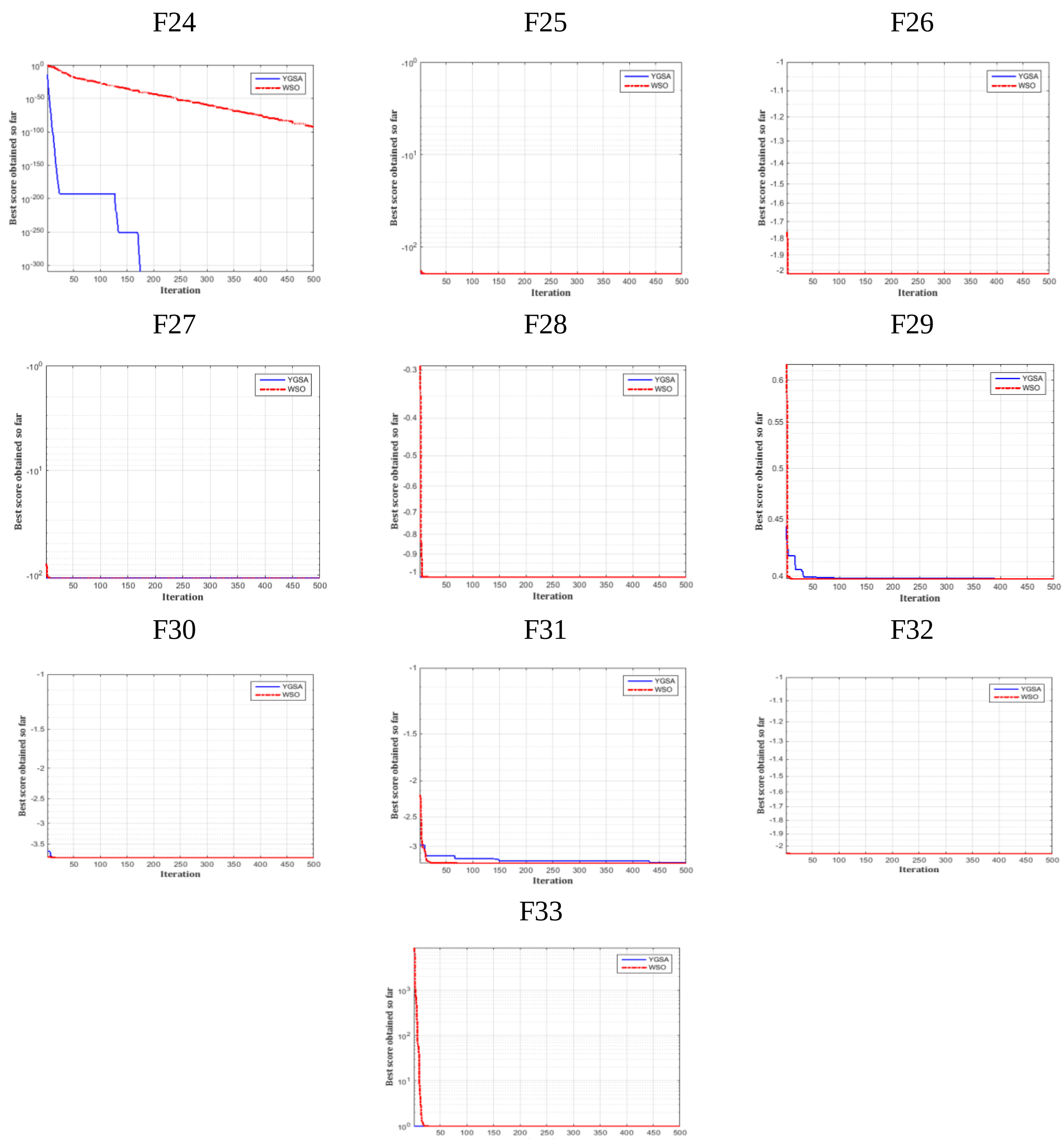


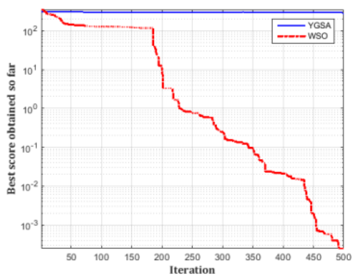
Figure 12. Multimodal fixed-dimension F24-F33 convergence rate plot

Power System Technology
ISSN:1000-3673

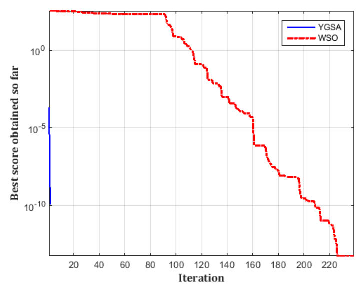
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

With respect to the set of multimodal variable-dimension test functions, the results depicted in Figure (13) illustrate that YGSA demonstrates superior convergence rates for functions F35, F36, F38, F39, F40, F42, F43, and F44. In contrast, WSO exhibits enhanced performance for functions F34, F37, F41, F45, and F46.

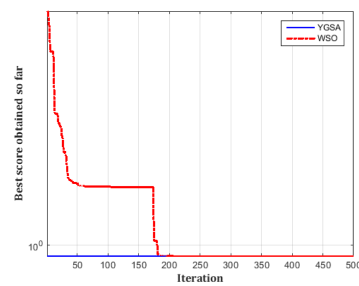
F34



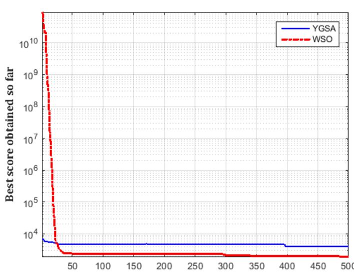
F35



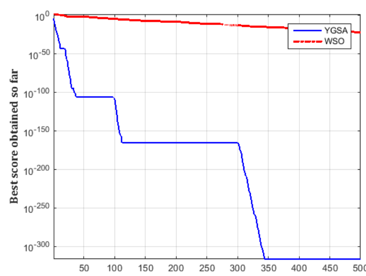
F36



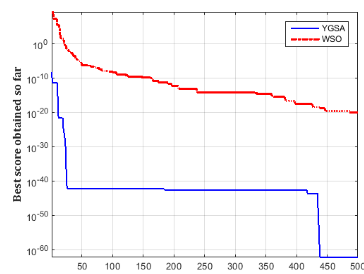
F37



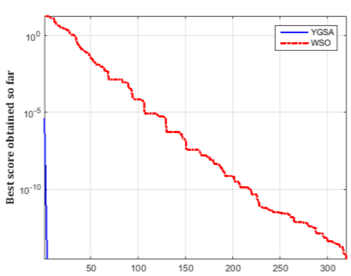
F38



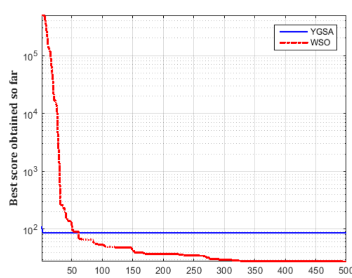
F39



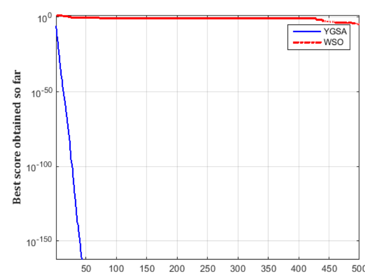
F40



F41

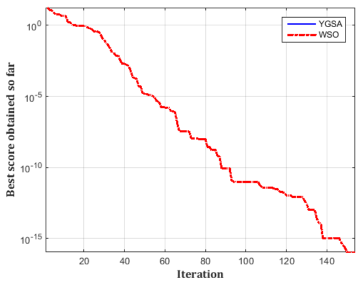


F42

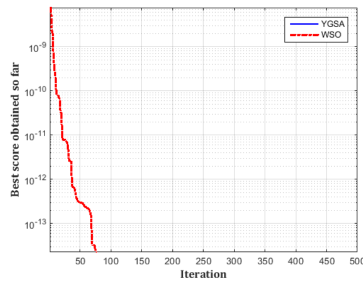


Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

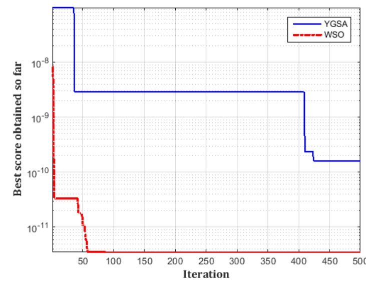
F43



F44



F45



F46

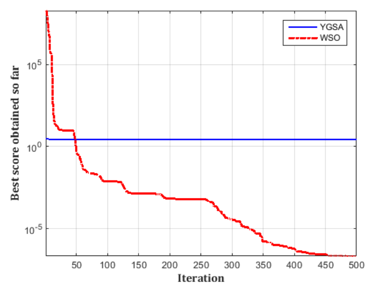


Figure 13. Multimodal variable-dimension F34-F46 convergence rate plot

It is notable that WSO performs exceptionally well for functions F43 and F44, where YGSA fails to exhibit the required efficiency within the defined range. Conversely, YGSA demonstrates significantly faster convergence rates for functions F35, F36, F40, and F42, converging towards the optimal solution in the early iterations.

Finally, Figure (14) shows the convergence trends for the CEC-06-2019 test functions. As mentioned previously, this set of test functions, known as new test functions, possesses highly complex characteristics. These functions aim to explore and evaluate the entire search space with varying modes, including rotations and translations. Evaluating the performance of the proposed algorithms with this test function set can provide insights into their effectiveness in solving diverse problems.

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

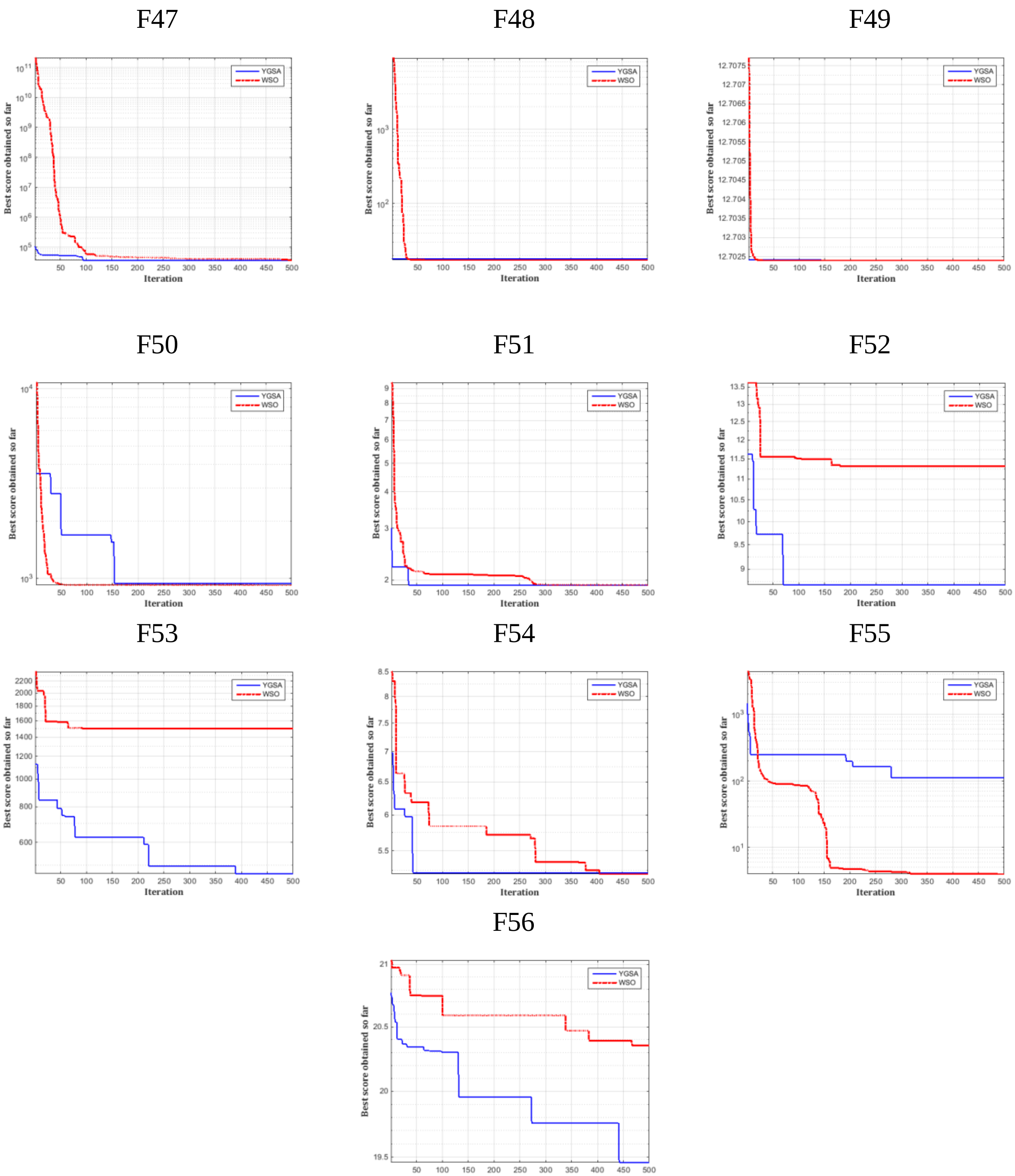


Figure 14. CEC-06-2019 F47-F56 convergence rate plot

Power System Technology
ISSN:1000-3673

Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

Analyzing Figure (14), it can be observed that YGSA exhibits superior convergence performance in discovering the optimal solution for functions F47 (CEC01), F48 (CEC02), F51 (CEC05), F52 (CEC06), F53 (CEC07), F54 (CEC08), and F56 (CEC10). Conversely, WSO surpasses YGSA in terms of performance for functions F49 (CEC03), F50 (CEC04), and F55 (CEC09). Overall, YGSA performs well in seven functions, while WSO performs well in three functions within this complex test function set. These results indicate a more desirable convergence behavior of the proposed YGSA compared to WSO.

e. The execution time

The duration between the initiation of algorithm execution by the simulator and the fulfillment of the termination condition is referred to as algorithm execution time. In this section, the evaluation of the execution time for the proposed YGSA algorithm and the WSO algorithm has been conducted on the set of 56 test functions. Tables (17) to (19) present the average execution time, measured in minutes, for both the proposed YGSA and the WSO algorithm.

Table 17. Average execution time of unimodal test functions

Test functions		WSO	YGSA
Unimodal fix-dimension test functions	F1	2 .2 m	6 m
	F2	2 m	5.3 m
	F3	2 m	2.4 m
	F4	2.3 m	4.8 m
	F5	2.1 m	5.7 m
	F6	2 m	3 m
	F7	2.8 m	6.1 m
	F8	1.7 m	2.3 m
	F9	2 m	3.2 m
Unimodal variable-dimension test functions	F10	2.3 m	4 m
	F11	2.1 m	3.4 m
	F12	3 m	4 m
	F13	2.1 m	3.1 m
	F14	2 m	2.8 m
	F15	2.1 m	3 m

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

	F16	2.7 m	3.1 m
	F17	2.1 m	3 m
	F18	2 m	3.2 m
	F19	2.2 m	2.6 m
	F20	3.1 m	4.2 m
	F21	2.6 m	3 m
	F22	2 m	3.3 m
	F23	2.5 m	3.2 m

Table 18. Average execution time of multimodal test functions

Test functions		YGSA	WSO
Multimodal fix-dimension test functions	F24	7 m	3 m
	F25	6 m	3.5 m
	F26	5.3 m	2.7 m
	F27	4.9 m	2.3 m
	F28	4.7 m	2.7 m
	F29	4.1 m	2.2 m
	F30	4.4 m	2.9 m
	F31	3.8 m	3 m
	F32	5 m	3.2 m
	F33	4.5 m	3 m
Multimodal variable-dimension test functions	F34	5.2 m	3.4 m
	F35	6.2 m	3.9 m
	F36	5.4 m	3.9 m
	F37	5 m	2.7 m
	F38	4.4 m	3.1 m
	F39	4 m	3 m

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

	F40	5.4 m	2.7 m
	F41	4.8 m	3 m
	F42	4.3 m	2.5 m
	F43	6.5 m	2.6 m
	F44	5.5 m	3.4 m
	F45	4 m	2.9 m
	F46	6 m	3.3 m

Table 19. Average execution time of CEC-06-2019 test functions

Test functions		YGSA	WSO
CEC-06-2019 test functions	F47 (CEC01)	10.2 m	6.5 m
	F48 (CEC02)	9 m	5.5 m
	F49 (CEC03)	11 m	5 m
	F50 (CEC04)	10.4 m	6 m
	F51 (CEC05)	9.3 m	5.8 m
	F52 (CEC06)	9.9 m	6.3 m
	F53 (CEC07)	10 m	5.2 m
	F54 (CEC08)	9.5 m	6.1 m
	F55 (CEC09)	11.3 m	6.6 m
	F56 (CEC10)	10.6 m	6 m

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

The simulation results indicate that the WSO algorithm consistently outperforms the proposed YGSA in terms of execution time for each test function. Table (17) provides the average execution time for unimodal test functions, Table (18) for multimodal test functions, and Table (19) for CEC-06-2019 test functions.

For instance, consider the case of function F22 from the unimodal test function set. While the proposed YGSA yields superior results in finding the optimal solution compared to WSO, it takes approximately 3.3 minutes to complete the test function. Conversely, WSO requires only 2 minutes to compute the same function. In the multimodal test function set, characterized by functions with multiple optima, the proposed YGSA necessitates more time to execute a test function and locate optimal points compared to WSO. For example, in the case of function F42, the proposed YGSA takes approximately 4.3 minutes to complete the algorithm execution, while WSO accomplishes it in just 2.5 minutes. Similarly, in the case of the CEC-06-2019 test function set, known for its highly complex nature, the WSO algorithm demonstrates faster execution time in finding optimal solutions for all test functions. As an illustration, executing function F51 (CEC05) in the proposed YGSA takes 9.3 minutes, while WSO achieves it in 6.3 minutes.

The higher execution time observed in the proposed YGSA for the test functions can be attributed to the foraging behavior of the squirrel. In the initial stages, the squirrel might not accurately select optimal points or choose positions far from the nest, depending on the farmer's location. Although this foraging strategy increases the overall execution time of the algorithm, it reduces the risk of being captured by the farmer. Another contributing factor to the higher execution time in the proposed YGSA is the number of algorithm steps involved. While the computational complexity of the proposed YGSA is not notably high, the execution time can be prolonged due to the larger number of algorithm steps. In contrast, the WSO algorithm features significantly fewer algorithm steps compared to the proposed YGSA, and the concurrent exploration and exploitation phases within each step of the proposed YGSA lead to increased execution time.

In summary, the algorithm execution time may vary across different test functions, but the capability of these algorithms to find optimal solutions remains independent of the execution time.

4.5 Analysis and Evaluation of Results

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

The remarkable exploration capability of the YGSA is attributed to the squirrel's mechanism of selecting the next position, as outlined in equations (4) to (16). These equations govern the squirrel's random selection of angular values within each of the four trigonometric quadrants by rotating its head 360 degrees, enabling it to update its next position. The stochastic nature of this selection process effectively mitigates the risk of converging to local optima. Subsequently, the chosen angles are transformed into coordinates on a two-dimensional plane using trigonometric equations. Ultimately, the squirrel updates its position by calculating the Euclidean distances to both the farmer and the nest, selecting coordinates that minimize the distance to the nest and maximize the distance to the farmer. This underscores the YGSA's remarkable capability during the exploration phase. The process iterates until the squirrel reaches the nest, gets captured by the farmer, or fulfills the termination condition, leading to algorithmic termination.

In essence, the squirrel's behavior in exploring various positions within the four trigonometric quadrants and selecting the next position hinges on the objective of minimizing the distance to the nest and maximizing the distance to the farmer. This approach facilitates the updating of the next position in each iteration of the extraction phase. Consequently, the YGSA executes both exploration and extraction phases concurrently within each iteration.

Comparatively, other algorithms such as WSO, HBO, and similar approaches employ analogous formulas to update their next step positions. These algorithms typically separate the exploration and extraction stages, thereby increasing the likelihood of falling into local optima. Table (17) provides a comprehensive overview based on each test set for the YGSA and WSO.

Table 17. Overall performance of YGSA and WSO

No	F. No	Name	YGSA	WSO	Equal
1	F1 – F23	Unimodal benchmark test functions	10	9	4
2	F24 – F46	Multimodal benchmark test functions	11	6	6
3	F47 – F56	CEC-06-2019	6	3	1

Table (17) reveals that the YGSA outperforms others, as indicated by the percentage values depicted in Figures (15) to (17). Across test set one (F1-F23), YGSA achieves an overall performance of 43.47%, while WSO attains 39.13%. Both algorithms exhibit an equal performance of 17.40%. Similarly, in test set two (F24-F46), YGSA surpasses WSO with a

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

performance of 47.82% compared to WSO's 26.08%. Again, both algorithms demonstrate an identical performance of 26.08%. In test set three (F47-F56), YGSA exhibits the highest performance at 60%, while WSO achieves 30%, and both algorithms perform equally at 10%. Consequently, the obtained results support the argument that YGSA outperforms not only other algorithms but also the best existing algorithm among the nine alternatives, namely WSO, in finding optimal solutions across the 56 test functions.

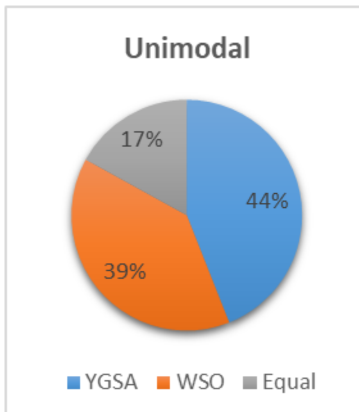


Figure 15. Overall performance of unimodal test functions

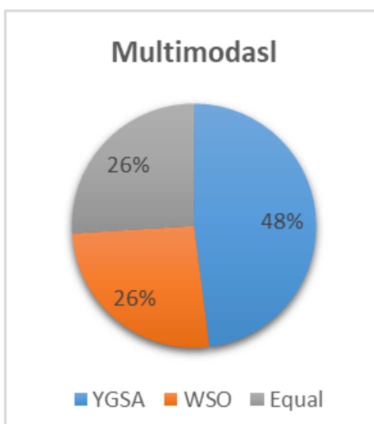


Figure 16. Overall performance of multimodal test functions

Power System Technology
ISSN:1000-3673

Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

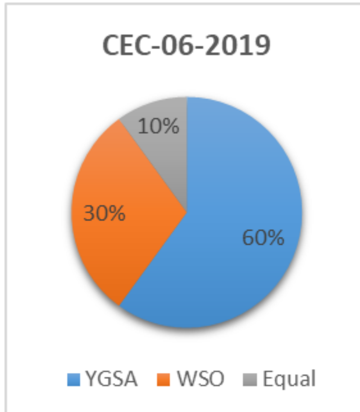


Figure 17. Overall performance of CEC-06-2019 test functions

5. *Conclusion and Future Works*

This paper introduced YGSA, a meta-heuristic algorithm inspired by the behavior of the yellow ground squirrel, for solving optimization problems. YGSA incorporates the squirrel's strategy of escaping from the farmer while maintaining a fixed distance and reaching the nest by utilizing candidate points. The primary objective of the YGSA is to enhance the efficiency of both the exploration and exploitation phases in each step, thereby improving the effectiveness of solving optimization functions and reducing the likelihood of falling into local optima. The performance of the proposed algorithm was evaluated by conducting tests on 56 benchmark test functions, including unimodal, multimodal with high dimensions, and multimodal with fixed dimensions. To further analyze the algorithm's capabilities, the optimization results obtained from YGSA were compared with those of nine well-known and widely used algorithms, including HBO, EBO-CMAR, MFO, SCA, PSO, CS, GSA, MVO, and WSO. The results for unimodal functions demonstrated the high exploitation ability of YGSA in converging towards global optimal solutions, while the simulation results for multimodal functions showcased its superior exploration capability in effectively exploring the search space and finding optimal regions. Notably, YGSA achieved the best results among the competing algorithms for complex CEC-06-2019 test functions, known for their high complexity in finding optimal points. Thus, the simulation results indicated that the proposed YGSA outperformed the nine competing algorithms in solving unimodal, multimodal, and complex optimization problems.

As a direction for future research, efforts can be focused on improving the execution time of the YGSA meta-heuristic algorithm and delving into its intricacies. Furthermore, enhancing the

Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

algorithm's performance in finding optimal points in benchmark test functions could be achieved by increasing the motion index through the consideration of multiple squirrels in the environment and incorporating a greater number of nests in the pursuit and evasion scenario. The development of the binary YGSA and the multi-objective YGSA is currently underway. Additionally, the proposed algorithm can be extended to various applications, such as disease detection, intrusion detection in computer networks, virtual machine consolidation, and scheduling in cloud data centers, to enhance resource allocation and reduce energy consumption.

References

1- Talbi, El-Ghazali. *Meta-heuristics: from design to implementation*. John Wiley & Sons, 2009.

2- Glover, Fred, and Manuel Laguna. *Tabu search*. Springer US, 1998.

3- Kirkpatrick, Scott, C. Daniel Gelatt Jr, and Mario P. Vecchi. "Optimization by simulated annealing." *science* 220, no. 4598 (1983): 671-680.

4- Eiben, Agoston E., and James E. Smith. *Introduction to evolutionary computing*. Springer-Verlag Berlin Heidelberg, 2015.

5- Dorigo, Marco, Vittorio Maniezzo, and Alberto Colorni. "Ant system: optimization by a colony of cooperating agents." *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 26, no. 1 (1996): 29-41.

6- Dorigo, Marco, and Thomas Stützle. *Ant colony optimization: overview and recent advances*. Springer International Publishing, 2019.

7- Yonezawa, Yasuo, and Takashi Kikuchi. "Ecological algorithm for optimal ordering used by collective honey bee behavior." In *MHS'96 Proceedings of the Seventh International Symposium on Micro Machine and Human Science*, pp. 249-256. IEEE, 1996.

8- Kennedy, James, and Russell Eberhart. "Particle swarm optimization." In *Proceedings of ICNN'95-international conference on neural networks*, vol. 4, pp. 1942-1948. IEEE, 1995.

9- Ghaemi, Manizheh, and Mohammad-Reza Feizi-Derakhshi. "Forest optimization algorithm." *Expert Systems with Applications* 41, no. 15 (2014): 6676-6687.

10- Rahkar Farshi, Taymaz. "Battle royale optimization algorithm." *Neural Computing and Applications* 33, no. 4 (2021): 1139-1157.

11- Ayyarao, Tummala SLV, N. S. S. Ramakrishna, Rajvikram Madurai Elavarasan, Nishanth Polumahanthi, M. Rambabu, Gaurav Saini, Baseem Khan, and Bilal Alatas. "War strategy optimization algorithm: a new effective meta-heuristic algorithm for global optimization." *IEEE Access* 10 (2022): 25073-25105.

Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

12- Mirjalili, Seyedali, Seyed Mohammad Mirjalili, and Andrew Lewis. "Grey wolf optimizer." *Advances in engineering software* 69 (2014): 46-61.
13- Mirjalili, Seyedali. "Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems." *Neural computing and applications* 27 (2016): 1053-1073.
14- Yang, Xin-She. "Flower pollination algorithm for global optimization." In *Unconventional Computation and Natural Computation: 11th International Conference, UCNC 2012, Orléan, France, September 3-7, 2012. Proceedings 11*, pp. 240-249. Springer Berlin Heidelberg, 2012.
15- Mirjalili, Seyedali, and Andrew Lewis. "The whale optimization algorithm." *Advances in engineering software* 95 (2016): 51-67.
16- Mirjalili, Seyedeh Zahra, Seyedali Mirjalili, Shahrzad Saremi, Hossam Faris, and Ibrahim Aljarah. "Grasshopper optimization algorithm for multi-objective optimization problems." *Applied Intelligence* 48 (2018): 805-820.
17- Harifi, Sasan, Madjid Khalilian, Javad Mohammadzadeh, and Sadoullah Ebrahimnejad. "Emperor Penguins Colony: a new meta-heuristic algorithm for optimization." *Evolutionary Intelligence* 12 (2019): 211-226.
18- Yoosefdoost, Icen, Milad Basirifard, and José Álvarez-García. "Reservoir Operation Management with New Multi-Objective (MOEPO) and Meta-heuristic (EPO) Algorithms." *Water* 14, no. 15 (2022): 2329.
19- Bandaru, Sunith, and Kalyanmoy Deb. "Meta-heuristic techniques." *Decision sciences* (2016): 693-750.
20- Yang, Xin-She. "Firefly algorithm, stochastic test functions and design optimisation." *International journal of bio-inspired computation* 2, no. 2 (2010): 78-84.
21- MiarNaeimi, Farid, Gholamreza Azizyan, and Mohsen Rashki. "Horse herd optimization algorithm: A nature-inspired algorithm for high-dimensional optimization problems." *Knowledge-Based Systems* 213 (2021): 106711.
22- Wolpert, David H., and William G. Macready. "No free lunch theorems for optimization." *IEEE transactions on evolutionary computation* 1, no. 1 (1997): 67-82.
23- Trojovský, Pavel, and Mohammad Dehghani. "Pelican optimization algorithm: A novel nature-inspired algorithm for engineering applications." *Sensors* 22, no. 3 (2022): 855.
24- Deb, Kalyanmoy. "Genetic algorithm in search and optimization: the technique and applications." In *Proceedings of International Workshop on Soft Computing and Intelligent Systems,(ISI, Calcutta, India)*, pp. 58-87. Proceedings of International Workshop on Soft Computing and Intelligent Systems,(ISI, Calcutta, India), 1998.

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

25- Storn, Rainer, and Kenneth Price. "Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces." *Journal of global optimization* 11, no. 4 (1997): 341.
26- Beyer, Hans-Georg, and Hans-Paul Schwefel. "Evolution strategies–a comprehensive introduction." *Natural computing* 1 (2002): 3-52.
27- Fogel, David B. *Artificial intelligence through simulated evolution*. Wiley-IEEE Press, 1998.
28- Mirjalili, Seyedali, Amir H. Gandomi, Seyedeh Zahra Mirjalili, Shahrzad Saremi, Hossam Faris, and Seyed Mohammad Mirjalili. "Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems." *Advances in engineering software* 114 (2017): 163-191.
29- Yang, Xin-She. "A new meta-heuristic bat-inspired algorithm." *Nature inspired cooperative strategies for optimization (NICSO 2010)* (2010): 65-74.
30- Mirjalili, Seyedali. "Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm." *Knowledge-based systems* 89 (2015): 228-249.
31- Karaboga, Dervis, and Bahriye Basturk. "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm." *Journal of global optimization* 39 (2007): 459-471.
32- Yang, Xin-She, and Suash Deb. "Cuckoo search via Lévy flights." In *2009 World congress on nature & biologically inspired computing (NaBIC)*, pp. 210-214. Ieee, 2009.
33- Rashedi, Esmat, Hossein Nezamabadi-Pour, and Saeid Saryazdi. "GSA: a gravitational search algorithm." *Information sciences* 179, no. 13 (2009): 2232-2248.
34- Zhang, Yiying, and Zhigang Jin. "Group teaching optimization algorithm: A novel meta-heuristic method for solving global optimization problems." *Expert Systems with Applications* 148 (2020): 113246.
35- Yang, Yutao, Huiling Chen, Ali Asghar Heidari, and Amir H. Gandomi. "Hunger games search: Visions, conception, implementation, deep analysis, perspectives, and towards performance shifts." *Expert Systems with Applications* 177 (2021): 114864.
36- Mirjalili, Seyedali, Seyed Mohammad Mirjalili, and Abdolreza Hatamlou. "Multi-verse optimizer: a nature-inspired algorithm for global optimization." *Neural Computing and Applications* 27 (2016): 495-513.
37- Askari, Qamar, Irfan Younas, and Mehreen Saeed. "Political Optimizer: A novel socio-inspired meta-heuristic for global optimization." *Knowledge-based systems* 195 (2020): 105709.
38- Kumar, Abhishek, Rakesh Kumar Misra, and Devender Singh. "Improving the local search capability of effective butterfly optimizer using covariance matrix adapted retreat phase." In *2017 IEEE congress on evolutionary computation (CEC)*, pp. 1835-1842. IEEE, 2017.

Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

39- Atashpaz-Gargari, Esmaeil, and Caro Lucas. "Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition." In *2007 IEEE congress on evolutionary computation*, pp. 4661-4667. Ieee, 2007.
40- Mirjalili, Seyedali. "SCA: a sine cosine algorithm for solving optimization problems." *Knowledge-based systems* 96 (2016): 120-133.
41- Askari, Qamar, Mehreen Saeed, and Irfan Younas. "Heap-based optimizer inspired by corporate rank hierarchy for global optimization." *Expert Systems with Applications* 161 (2020): 113702.
42- Simon, Dan. "Biogeography-based optimization." *IEEE transactions on evolutionary computation* 12, no. 6 (2008): 702-713.
43- Mehrabian, Ali Reza, and Caro Lucas. "A novel numerical optimization algorithm inspired from weed colonization." *Ecological informatics* 1, no. 4 (2006): 355-366.
44- Chen, Hanning, and Yunlong Zhu. "Optimization based on symbiotic multi-species coevolution." *Applied Mathematics and Computation* 205, no. 1 (2008): 47-60.
45- Goodrich, Michael T., Joseph SB Mitchell, and Mark W. Orletsky. "Approximate geometric pattern matching under rigid motions." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 21, no. 4 (1999): 371-379.
46- Soleymani, Ali, Jonathan Cachat, Kyle Robinson, Somayeh Dodge, Allan Kalueff, and Robert Weibel. "Integrating cross-scale analysis in the spatial and temporal domains for classification of behavioral movement." *Journal of Spatial Information Science* 8 (2014): 1-25.
47- Ranacher, Peter, and Katerina Tzavella. "How to compare movement? A review of physical movement similarity measures in geographic information science and beyond." *Cartography and geographic information science* 41, no. 3 (2014): 286-307.
48- Laube, Patrick. *Computational movement analysis*. Cham: Springer International Publishing, 2014.
49- Gottfried, Interpretation–BMI B., and H. Aghajan. "Progress in movement pattern analysis." *Behaviour monitoring and interpretation-BMI: Smart environments* 3 (2009): 43.
50- Chen, Lei, and Raymond Ng. "On the marriage of lp-norms and edit distance." In *Proceedings of the Thirtieth international conference on Very large data bases-Volume 30*, pp. 792-803. 2004.
51- Alt, Helmut, and Michael Godau. "Computing the Fréchet distance between two polygonal curves." *International Journal of Computational Geometry & Applications* 5, no. 01n02 (1995): 75-91.

Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

52- Matrosova, Vera A., Ilya A. Volodin, Elena V. Volodina, and Nina A. Vasilieva. "Stability of acoustic individuality in the alarm calls of wild yellow ground squirrels Spermophilus fulvus and contrasting calls from trapped and free-ranging callers." *Naturwissenschaften* 97 (2010): 707-715.
53- Vasilieva, N. A., and A. V. Tchabovsky. "Reproductive decisions in a “fast-living” sciurid: a case study of the yellow ground squirrel (Spermophilus fulvus)." *Biology Bulletin Reviews* 8 (2018): 12-22.
54- Ojha, Varun Kumar, Ajith Abraham, and Václav Snášel. "Meta-heuristic design of feedforward neural networks: A review of two decades of research." *Engineering Applications of Artificial Intelligence* 60 (2017): 97-116.
55- Hosseini, Soodeh, and Behnam Mohammad Hasani Zade. "New hybrid method for attack detection using combination of evolutionary algorithms, SVM, and ANN." *Computer Networks* 173 (2020): 107168.
56- Murata, Tadahiko, and Hisao Ishibuchi. "MOGA: multi-objective genetic algorithms." In *IEEE international conference on evolutionary computation*, vol. 1, pp. 289-294. Piscataway, NJ, USA: IEEE, 1995.
57- Heidari, Ali Asghar, Seyedali Mirjalili, Hossam Faris, Ibrahim Aljarah, Majdi Mafarja, and Huiling Chen. "Harris hawks optimization: Algorithm and applications." *Future generation computer systems* 97 (2019): 849-872.
58- Weisstein, Eric W. "Circle packing." *https://mathworld. wolfram. com/* (2002).
59- Newman, Lex, and Alan Nelson. "Circumventing Cartesian Circles." *Noûs* 33, no. 3 (1999): 370-404.
60- Brown, Clifford T., Larry S. Liebovitch, and Rachel Glendon. "Lévy flights in Dobe Ju/’hoansi foraging patterns." *Human Ecology* 35 (2007): 129-138.
61- Reynolds, Andy M., and Mark A. Frye. "Free-flight odor tracking in Drosophila is consistent with an optimal intermittent scale-free search." *PloS one* 2, no. 4 (2007): e354.
62- Gelman, Andrew, Walter R. Gilks, and Gareth O. Roberts. "Weak convergence and optimal scaling of random walk Metropolis algorithms." *The annals of applied probability* 7, no. 1 (1997): 110-120.
63- Lawler, Gregory F. "Weak convergence of a random walk in a random environment." *Communications in Mathematical Physics* 87 (1982): 81-87.
64- Hassanat, Ahmad, Khalid Almohammadi, Esra’A. Alkafaween, Eman Abunawas, Awni Hammouri, and VB Surya Prasath. "Choosing mutation and crossover ratios for genetic algorithms—a review with a new dynamic approach." *Information* 10, no. 12 (2019): 390.

Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

65- Ahvanooey, Milad Taleby, Qianmu Li, Ming Wu, and Shuo Wang. "A Survey of Genetic Programming and Its Applications." *KSII Trans. Internet Inf. Syst.* 13, no. 4 (2019): 1765-1794.
66- Godwin, H. J. "On the distribution of the estimate of mean deviation obtained from samples from a normal population." *Biometrika* 33, no. 3 (1945): 254-256.
67- Grechuk, Bogdan, Anton Molyboha, and Michael Zabarankin. "Mean- Deviation Analysis in the Theory of Choice." *Risk Analysis: An International Journal* 32, no. 8 (2012): 1277-1292.
68- Farrahi Farimani, Hojjat, Davoud Bahrepour, and Seyed Reza Kamel Tabbakh. "Reallocation of Virtual Machines to Cloud Data Centers to Reduce Service Level Agreement Violation and Energy Consumption Using the FMT Method." *Journal of Information Systems and Telecommunication (JIST)* 4, no. 28 (2020): 316.
69- Patro, S. G. O. P. A. L., and Kishore Kumar Sahu. "Normalization: A preprocessing stage." *arXiv preprint arXiv:1503.06462* (2015).
70- Eesa, Adel S., and Wahab Kh Arabo. "A normalization methods for backpropagation: a comparative study." *Science Journal of University of Zakho* 5, no. 4 (2017): 319-323.
71- Almufti, S., R. Marqas, and V. Ashqi. "Taxonomy of bio-inspired optimization algorithms." *Journal Of Advanced Computer Science & Technology* 8, no. 2 (2019): 23.
72- Roeva, Olympia, Tsonyo Slavov, and Stefka Fidanova. "Population-based vs. single point search meta-heuristics for a pid controller tuning." In *Handbook of research on novel soft computing intelligent algorithms: theory and practical applications*, pp. 200-233. IGI Global, 2014.
73- Yang, Xin-She. "Test problems in optimization." *arXiv preprint arXiv:1008.0549* (2010).
74- Molga, Marcin, and Czesław Smutnicki. "Test functions for optimization needs." *Test functions for optimization needs* 101 (2005): 48.
75- Hussain, Kashif, Mohd Najib Mohd Salleh, Shi Cheng, and Rashid Naseem. "Common benchmark functions for meta-heuristic evaluation: A review." *JOIV: International Journal on Informatics Visualization* 1, no. 4-2 (2017): 218-223.
76- Rai, Rebika, Krishna Gopal Dhal, Arunita Das, and Swarnajit Ray. "An Inclusive Survey on Marine Predators Algorithm: Variants and Applications." *Archives of Computational Methods in Engineering* (2023): 1-40.
77- Luo, Wenjian, Xin Lin, Changhe Li, Shengxiang Yang, and Yuhui Shi. "Benchmark functions for cec 2022 competition on seeking multiple optima in dynamic environments." *arXiv preprint arXiv:2201.00523* (2022).

Power System Technology
ISSN:1000-3673
Received: 14-10-2023 Revised: 12-11-2023 Accepted: 20-12-2023

78- Mohamed, Ali Wagdy, Karam M. Sallam, Prachi Agrawal, Anas A. Hadi, and Ali Khater Mohamed. "Evaluating the performance of meta-heuristic algorithms on CEC 2021 benchmark problems." *Neural Computing and Applications* 35, no. 2 (2023): 1493-1517.