scientific reports

Check for updates

OPEN

Hierarchical guided manta ray foraging optimization for global continuous optimization problems and parameter estimation of solar photovoltaic models

Zhentao Tang[1,2,3], Kaiyu Wang[4], Lan Zhuang[1], Mingxin Zhu[1], Yongxuan Yao[1], Huiqin Chen[1,2,3], Jing Li[1,2,3], Xiaoxiang Wu[1,2,3] & Shangce Gao[4]✉

The manta ray foraging optimization (MRFO) algorithm has gained widespread application in engineering problems due to its notable performance and low computational cost. However, the algorithm is prone to getting trapped in local optima and struggles to effectively balance exploration and exploitation, primarily due to the use of fixed parameters and its somersault foraging mechanism, which focuses solely on the current best solution. To address these shortcomings, we propose an improved version called hierarchical guided manta ray foraging optimization (HGMRFO). In this version, the fixed parameters are replaced with an adaptive somersault factor, which helps balance exploration and exploitation during the evolutionary process. A novel hierarchical guidance mechanism is introduced in the somersault foraging strategy. Starting from the population structure, the mechanism guides the search direction of the population individuals through hierarchical interactions. To validate the effectiveness of HGMRFO, we compared it with seven state-of-the-art algorithms on 29 IEEE CEC2017 benchmark functions, achieving an average win rate of 73.15%. On 22 IEEE CEC2011 real-world optimization problems, HGMRFO obtained the most optimal solutions. Additionally, we analyzed the optimal parameter configurations, the impact of hierarchical interactions, and the computational complexity of the proposed method. Finally, when solving the parameter estimation problems for six multimodal photovoltaic models, HGMRFO outperformed other competing methods with a success rate of 97.62%, highlighting its superior performance in the photovoltaic field.

Keywords Manta ray foraging optimization, Exploration and exploitation, Hierarchical guidance mechanism, Hierarchical interactions, Photovoltaic models, Parameter estimation

Metaheuristic optimization algorithms have emerged as a prominent class of optimization techniques capable of addressing complex, nonlinear, and high-dimensional problems[1,2]. Unlike classical optimization methods, metaheuristics do not require derivative information and are more flexible in handling a wide range of objective functions[3]. By drawing inspiration from natural and physical processes, these algorithms effectively explore and exploit solution spaces. Prominent examples of metaheuristic algorithms include genetic algorithms (GA)[4], particle swarm optimization (PSO)[5], ant colony optimization (ACO)[6], and gravitational search algorithm (GSA)[7]. There are also the latest algorithms, such as draco lizard optimizer[8], eurasian lynx optimizer[9], and artificial meerkat algorithm[10].

One critical feature of many advanced metaheuristic methods is their hierarchical structure, which organizes the population into different layers or subgroups[11]. The hierarchical structure facilitates efficient information exchange and diversity preservation by promoting interactions between various segments of the population[12]. This layered approach is crucial for preventing premature convergence and ensuring comprehensive exploration of the solution space. By structuring the population hierarchically, metaheuristic algorithms can effectively

[1]Jiangsu Agri-animal Husbandry Vocational College, Taizhou 225300, China. [2]Jiangsu Agricultural Internet of Things Engineering Technology Research and Development Center, Taizhou 225300, China. [3]Taizhou Smart Agriculture Engineering Research Center, Taizhou 225300, China. [4]Faculty of Engineering, University of Toyama, Toyama-shi 930-8555, Japan. ✉email: gaosc@eng.u-toyama.ac.jp

balance the dual demands of exploration (searching new regions) and exploitation (intensifying search around promising solutions).

Manta ray foraging optimization (MRFO)[13] has emerged as a powerful technique inspired by the unique foraging behaviors of manta rays, including chain foraging, cyclone foraging, and somersault foraging strategies. These strategies enable manta rays to adapt dynamically to changing environmental conditions, allowing MRFO to maintain a balance between exploration and exploitation. The original MRFO has shown significant effectiveness in addressing a variety of practical optimization challenges. Nevertheless, as is common with many metaheuristic approaches, MRFO faces challenges such as premature convergence and reduced population diversity, which limit its capacity to reliably discover the global optimal solution. Addressing these challenges requires further enhancements to ensure effective exploration of the solution space and to prevent convergence to suboptimal solutions.

In recent years, numerous advancements have been made in the MRFO algorithm, aimed at overcoming its limitations and enhancing its performance. Researchers have focused on strategies to mitigate premature convergence and enhance population diversity. For instance, adaptive control mechanisms for parameters have been proposed, allowing the algorithm to dynamically adjust based on the state of the search process[14,15]. Hybrid approaches that integrate MRFO with other metaheuristic algorithms, such as GA and PSO, have also been introduced to enhance global exploration capabilities and improve solution quality[16,17].

Another line of research has focused on developing novel selection strategies to diversify the population and maintain a robust search process. These strategies often involve elite preservation, and opposition-based learning techniques to help balance exploration and exploitation[12,18]. Elite preservation ensures that the best solutions are retained across generations, opposition-based learning has been demonstrated to be particularly effective in enhancing the convergence rate of MRFO by periodically reinitializing certain members of the population with more promising candidates[19,20]. Additionally, the incorporation of chaotic maps and Levy flights has been demonstrated to further enhance MRFO's search efficiency, minimizing the probability of the algorithm getting trapped in local optima[21,22]. Chaotic maps help in better initial population distribution, which is crucial for global exploration, whereas Levy flights provide a mechanism for making larger, more diverse jumps in the search space, especially useful for avoiding local optima.

Moreover, multi-objective extensions of MRFO have been introduced to address problems involving multiple conflicting objectives. These multi-objective versions utilize Pareto-based selection mechanisms and crowding distance measures to maintain a diverse set of solutions while achieving an effective trade-off among objectives[23,24]. Recently, more advanced multi-objective methods such as inertia weight decreasing approaches and adaptive trade-off models have been incorporated to improve the flexibility and robustness of multi-objective MRFO[25,26]. These improvements allow MRFO to handle practical, real-world scenarios where multiple criteria must be optimized simultaneously, providing decision-makers with a set of Pareto-optimal solutions.

Another noteworthy advancement is the incorporation of memory-based strategies within MRFO, where successful solutions from previous iterations are stored and used to guide future search efforts[15,27]. Such strategies allow the algorithm to learn from historical data, which significantly improves the efficiency of the search, especially in dynamic environments. Furthermore, adaptive step-size mechanisms have been developed to refine the local search capabilities of MRFO. By adaptively adjusting the search step size based on the improvement rate in the fitness value, these methods help strike a better balance between exploration and exploitation[21,28].

MRFO also has been effectively integrated with advanced machine learning methodologies, leading to significant advancements in feature selection, hyperparameter optimization, and classification tasks, particularly in medical and biomedical applications. MRFO has been employed to optimize support vector machines (SVM) and convolutional neural networks (CNN), resulting in enhanced classification accuracy and reduced data dimensionality[29]. MRFO has been hybridized with CNNs for brain tumor detection, demonstrating superior performance compared to traditional optimization methods[30]. Collectively, MRFO substantially enhances the accuracy, robustness, and generalization capabilities of machine learning models in the medical and biomedical domains, providing a powerful tool for addressing complex classification and feature selection challenges.

Overall, these improvements have made MRFO a more versatile and powerful optimization tool capable of solving a broader range of complex problems. The advancements not only extend the applicability of MRFO to diverse fields such as engineering design, energy systems, and biomedicine but also enhance its effectiveness in handling practical challenges characterized by multiple objectives, dynamic environments, and computational constraints.

In recent years, numerous parameter estimation methods for solar photovoltaic models have emerged. In[31], experimental comparisons between hybrid methods and metaheuristic methods for photovoltaic parameter estimation revealed that hybrid methods provide more accurate parameter estimation. In[32], the trust-region dogleg (TRD) and trust-region reflective (TRR) algorithms were employed to estimate three unknown parameters in the solar photovoltaic model, while the remaining two parameters were determined using an analytical approach. In[33], a hybrid approach combining grey wolf optimization (GWO) and analytical methods achieved promising results in photovoltaic parameter estimation. In[34], a comparative analysis of existing photovoltaic parameter estimation techniques revealed that hybrid-based parameter estimation methods achieve more accurate photovoltaic characteristics. In[35], a multi-hybrid evolutionary algorithm based on the flower pollination algorithm (FPA), grey wolf optimizer (GWO), and differential evolution (DE) algorithm was used for precise and effective parameter estimation of photovoltaic systems. In[36], a new algorithm, the pelican optimization algorithm (POA), was proposed for the optimal parameter estimation of photovoltaic systems, achieving promising results. In[37], reinforcement learning and nonlinear hunting strategies were integrated into the golden jackal optimizer, achieving promising results in photovoltaic parameter estimation, making it a reliable tool for this task. In[38], a multi-strategy (self-adaptive weighting mechanism, chaotic elite learning approach, and experience-based learning method) tree seed algorithm was used for photovoltaic model

parameter estimation, achieving successful results. In[39], the study found that the hybrid approach combining the linear slope method and GWO yields photovoltaic parameters that are closer to the real characteristics. In[40], a nonlinear least squares (NLS) objective function was constructed using manufacturer datasheet information, and particle swarm optimization (PSO) was employed to estimate the unknown parameters of the photovoltaic model, achieving higher accuracy.

Within these study, we introduce the hierarchical guided manta ray foraging optimization (HGMRFO), which incorporates an adaptive somersault factor to replace the original fixed parameter S, and a hierarchical guidance search mechanism to replace the sole reliance on the current best individual during somersault foraging. These modifications enable effective directionality in individual searches and facilitate an improved equilibrium between exploration and exploitation. During the experimental phase, the optimal parameter configuration for HGMRFO was systematically determined, and the effectiveness of the complete hierarchical structure was demonstrated. Comparative analyses were conducted using the IEEE CEC2017 benchmark and IEEE CEC2011 real-world optimization problem sets, thereby validating the superior performance and practicality of HGMRFO. Furthermore, the proposed algorithm was applied to parameter estimation in solar photovoltaic models, where it exhibited promising real-world applicability and effectiveness.

The main contributions of this research are as follows:

1. Introduction of the hierarchical guidance mechanism: This paper innovatively alters the population structure, guiding individuals toward the global optimal solution through layered interactions. This approach overcomes the limitations of many existing algorithms that focus solely on searching around the current best solution. On the CEC2017 benchmark functions, the algorithm achieved an average win rate of 73.15%, and it obtained the most optimal solutions on the CEC2011 real-world optimization problems. These results effectively demonstrate that the mechanism significantly enhances the algorithm's global search capability in complex optimization problems.

2. Adaptive parameter adjustment strategy: An adaptive parameter tuning mechanism was implemented to achieve a dynamic balance between the exploration and exploitation stages of the search process. This strategy alleviates premature convergence issues caused by fixed parameter settings. The convergence curves obtained across various dimensional functions clearly show that this strategy significantly improves the algorithm's robustness and effective convergence.

3. Application in photovoltaic model parameter estimation: The improved algorithm was applied to parameter estimation for six photovoltaic (PV) models, achieving a 97.62% success rate. This not only verifies the algorithm's comprehensive performance in PV modeling but also demonstrates its potential practical utility and broad applicability in the renewable energy field.

The remainder of this paper is structured as follows: Section "Related work" provides an overview of related work, including a detailed introduction to the fundamental MRFO and classifications of population structures. Section "Hierarchical guided manta ray foraging optimization (HGMRFO)" identifies the limitations of MRFO and subsequently proposes the HGMRFO as an enhancement. Section "Experiments and analysis" presents the experimental and analytical aspects of the study, where the parameter sensitivity and hierarchical interactions of HGMRFO are comprehensively analyzed, and comparative experiments with state-of-the-art algorithms are conducted on benchmark functions and real-world optimization problems. Moreover, the practical applicability of HGMRFO is validated through its application to parameter estimation in photovoltaic models. Section "Conclusions" concludes the paper by summarizing key findings and outlining potential directions for future research.

Related work

Basic MRFO

MRFO algorithm is inspired by the unique foraging behavior of manta rays and employs three distinct mechanisms-chain foraging, cyclone foraging, and somersault foraging-to effectively drive the optimization process. These mechanisms collectively simulate the exploratory and exploitative capabilities of the swarm, ultimately enhancing optimization performance.

In the MRFO framework, a population of N manta ray individuals is represented as $X_i = \left(x_i^1, x_i^2, \ldots, x_i^d\right)$, where $i \in \{1, 2, \ldots, N\}$ and x_i^d denotes the position of the ith individual in the dth dimension. Each individual employs different foraging behaviors to navigate the search space, described in detail below.

The *chain foraging* mechanism replicates the behavior of manta rays moving in a coordinated chain formation, in which each individual follows the one ahead. The position of each manta ray is influenced by both the current best solution and the individual directly in front of it, which enhances both exploration and convergence toward the optimal solution. The update equation for chain foraging is formulated as:

$$x_i^d(t+1) = \begin{cases} x_i^d(t) + r \cdot \left(x^d_{\text{best}}(t) - x_i^d(t)\right) \\ \quad +\alpha \cdot \left(x^d_{\text{best}}(t) - x_i^d(t)\right), & i = 1 \\ x_i^d(t) + r \cdot \left(x_{i-1}^d(t) - x_i^d(t)\right) \\ \quad +\alpha \cdot \left(x^d_{\text{best}}(t) - x_i^d(t)\right), & i = 2, \ldots, N \end{cases} \tag{1}$$

$$\alpha = 2 \cdot r \cdot \sqrt{|\log(r)|} \tag{2}$$

where, $X_{\text{best}} = \left(x^1_{\text{best}}, x^2_{\text{best}}, \ldots, x^d_{\text{best}}\right)$ represents the current best individual, $x^d_{\text{best}}(t)$ denotes the position of the best individual in the dth dimension at time t. r is a random number within the range $[0, 1]$. The coefficient α is introduced to adjust the weight of the search direction, enhancing the effectiveness of the foraging behavior.

In the *cyclone foraging* phase, manta rays exhibit a spiraling behavior around the current best solution, which enhances their ability to perform local search. Unlike chain foraging, this behavior aims to increase the exploration capability of the population by generating a swirling motion around the optimal solution. The position update in cyclone foraging is modeled as:

$$x_i^d(t+1) = \begin{cases} x^d_{\text{best}} + r \cdot \left(x^d_{\text{best}}(t) - x_i^d(t)\right) \\ \quad +\beta \cdot \left(x^d_{\text{best}}(t) - x_i^d(t)\right), & i = 1 \\ x^d_{\text{best}} + r \cdot \left(x^d_{i-1}(t) - x_i^d(t)\right) \\ \quad +\beta \cdot \left(x^d_{\text{best}}(t) - x_i^d(t)\right), & i = 2, \ldots, N \end{cases} \tag{3}$$

$$\beta = 2e^{r_1 \frac{T-t+1}{T}} \cdot \sin(2\pi r_1) \tag{4}$$

where β is a dynamic coefficient dependent on the random number $r_1 \in [0, 1]$, and T represents the maximum number of iterations. The cyclone foraging mechanism includes an additional feature that selects a random individual as a reference point to preserve variability and mitigate the likelihood of getting trapped in local optima, formulated as:

$$x^d_{\text{rand}} = Lb^d + r \cdot \left(Ub^d - Lb^d\right) \tag{5}$$

$$x_i^d(t+1) = \begin{cases} x^d_{\text{rand}} + r \cdot \left(x^d_{\text{rand}}(t) - x_i^d(t)\right) \\ \quad +\beta \cdot \left(x^d_{\text{rand}}(t) - x_i^d(t)\right), & i = 1 \\ x^d_{\text{rand}} + r \cdot \left(x^d_{i-1}(t) - x_i^d(t)\right) \\ \quad +\beta \cdot \left(x^d_{\text{rand}}(t) - x_i^d(t)\right), & i = 2, \ldots, N \end{cases} \tag{6}$$

where x^d_{rand} denotes the position of a randomly selected individual, and Lb^d, Ub^d represent the lower and upper bounds for dimension d. This mechanism ensures improved solution diversity, effectively preventing premature convergence.

The final strategy, *somersault foraging*, simulates the somersault movements of manta rays around the optimal solution. This behavior aims to balance exploration and exploitation by allowing individuals to adjust their positions with respect to the best-known solution, thereby improving convergence accuracy. The somersault foraging update is expressed as:

$$x_i^d(t+1) = x_i^d(t) + S \cdot \left(r_2 \cdot x^d_{\text{best}} - r_3 \cdot x_i^d(t)\right), \quad i = 1, \ldots, N \tag{7}$$

where S is a constant parameter controlling the somersault magnitude, and $r_2, r_3 \in [0, 1]$ are random numbers. This mechanism progressively reduces the movement range of each individual as iterations proceed, thereby enhancing the precision of the solution search.

The overall workflow of MRFO is depicted in Fig. 1. Initially, either chain or cyclone foraging is selected based on a random criterion. During the cyclone foraging phase, the decision of whether to use the current best solution or a random reference is made based on the ratio t/T, thereby balancing global and local exploration. Ultimately, each individual performs somersault foraging to refine its position, ensuring steady convergence to the optimal solution. This coordinated use of chain, cyclone, and somersault foraging effectively balances the exploration and exploitation capabilities of the algorithm, thereby enhancing both its robustness and convergence efficiency.

Population structure

Population structure plays a crucial role in optimization algorithms, involving the allocation, organization, and management of individuals to identify superior solutions. The design of population structure significantly impacts the exchange of information among individuals and the maintenance of population diversity, both of which are critical for effective exploration and exploitation within the search space. Common population structures are generally classified into three categories:

(1) Panmictic structure: The panmictic structure is the most fundamental and widely adopted population configuration[41]. In this structure, individuals interact with each other by randomly selecting partners, which expedites the dissemination of information across the population. However, frequent and random interactions often result in insufficient population diversity, leading to premature convergence and hindering comprehensive exploration of the solution space.
(2) Neighborhood structures: To address the premature convergence problem inherent in panmictic structures, neighborhood-based configurations have been developed, leveraging both fitness values and spatial topology to enhance information exchange[42,43]. By promoting diversified neighborhood interactions, these structures facilitate evolutionary processes and enhance algorithmic performance. Neighborhood structures are predominantly categorized into cellular structures and distributed structures[44,45].

In cellular structures, individuals are organized into a grid, interacting exclusively with their immediate neighbors, which may partially overlap. This localized interaction helps maintain population diversity and mitigate premature convergence[42]. For instance, Dorronsoro and Alba introduced a cellular genetic algorithm that demonstrated the effective diffusion of solutions throughout the population[45]. Cellular structures have

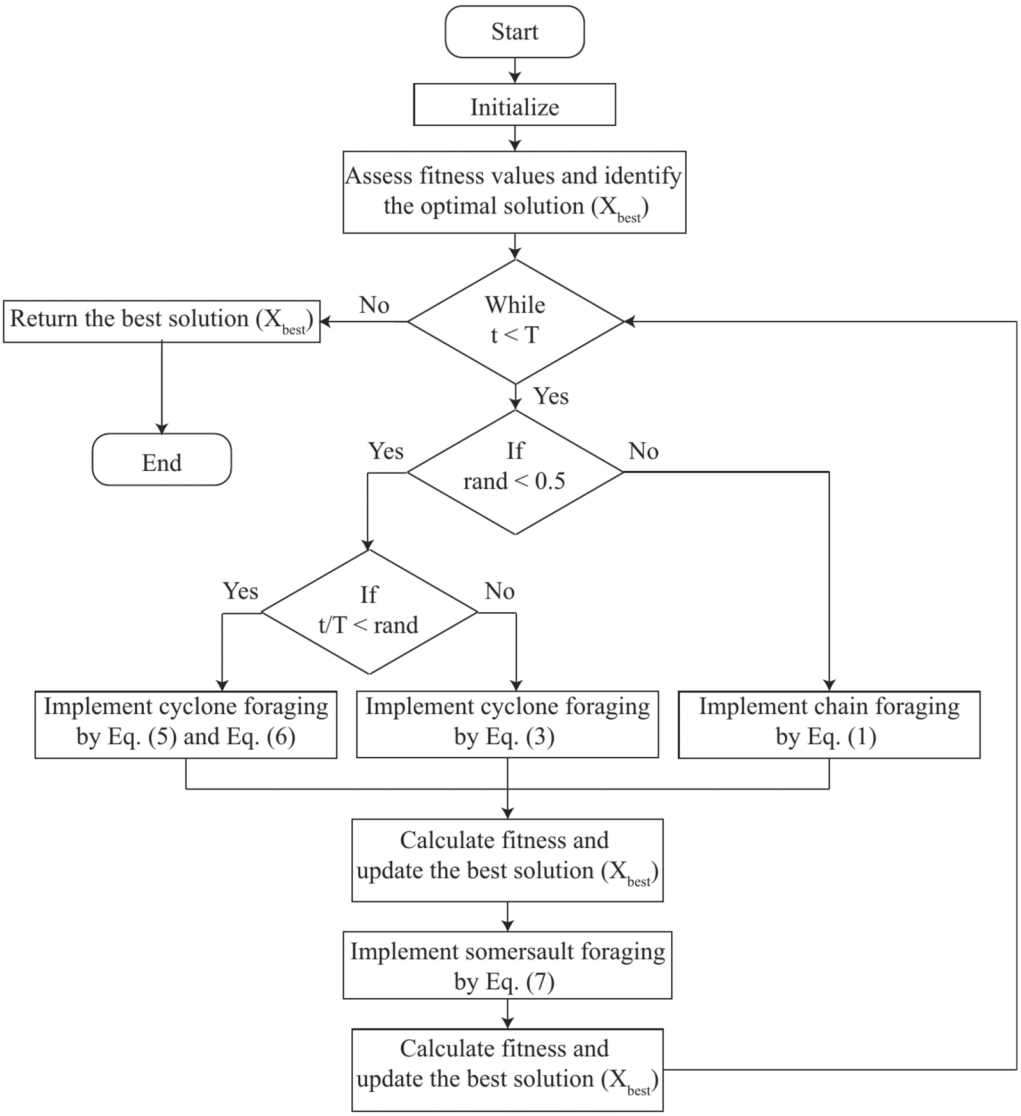


Fig. 1. Flowchart of MRFO.

also been employed to enhance differential evolution variants[46] and address dynamic optimization problems by integrating mechanisms such as clonal selection[47]. In the context of differential evolution, cellular direction information has been utilized to improve exploration and convergence rates through a combined approach[48].

In distributed structures, the population is partitioned into multiple subpopulations that evolve independently while sharing information via migration strategies. Each subpopulation searches the solution space in its specific region, and migration strategies facilitate effective information exchange among different subpopulations[49]. Through this migration mechanism, distributed structures effectively expand the exploration range and further enhance population diversity. For example, a PSO divided the population into two types of swarms-one composed of elite individuals and the other used for exploring alternative regions of the solution space[50]. In another study, a multicultural migration strategy was proposed to further increase population diversity in distributed differential evolution[51].

The hierarchical structure is a specific type of distributed configuration that allocates individuals across multiple layers, each serving distinct purposes, while enabling dynamic information exchange among the layers[52]. This stratification facilitates the evolution of subsequent generations by promoting inter-layer communication. For example, in dynamic hierarchical particle swarm optimization, particles can move vertically within the hierarchy, allowing the most promising individuals to ascend and optimize the population at multiple levels[53]. Hierarchical designs, such as the multi-layered GSA, significantly improve the balance between exploration and exploitation through hierarchical interactions[11]. In the context of MRFO, Tang et al. introduced an update mechanism that incorporates weighted fitness-distance balance selection, further enhancing diversity[12]. Similarly, Wang et al. proposed a deep-layered differential evolution approach comprising four distinct layers, effectively balancing diversity, exploration, and exploitation[54].

3) Structures with small-world and scale-free attributes: In addition to panmictic and neighborhood structures, some population configurations exhibit small-world and scale-free properties. The Watts-Strogatz and Barabási-Albert models are commonly employed to construct small-world networks and scale-free topologies, respectively[55]. Small-world networks uniquely blend features of random graphs and regular lattices, significantly improving the balance between exploration and exploitation during the search process[41]. For example, Zhang et al. used scale-free networks to adapt the PSO population structure, endowing it with scale-free properties that significantly improved population diversity and algorithm performance[56]. In another study, Yu et al. leveraged neighborhood information from scale-free networks to refine the mutation operator in differential evolution, thus improving the algorithm's capacity for both exploration and exploitation of the search space[57].

In conclusion, population structure is fundamental to the success of optimization algorithms. Through careful design of topological structures and strategic interactions among individuals, population structures can achieve an effective balance between exploration and exploitation, significantly improving both the efficiency and quality of solution searches[58–60]. Different population structures offer unique advantages in terms of information exchange and diversity maintenance, making them suitable for a wide range of optimization problems.

Hierarchical guided manta ray foraging optimization (HGMRFO)

Issues of MRFO

The somersault foraging mechanism in the MRFO algorithm, which aims to replicate the intricate movement behavior of manta rays during their foraging activities, presents notable limitations in its current form. These limitations are evident in the following update equation:

$$x_i^d(t+1) = x_i^d(t) + S \cdot \left(r_2 \cdot x_{\text{best}}^d - r_3 \cdot x_i^d(t)\right), \quad i = 1, \ldots, N$$

The main limitations are:

(a) Fixed parameter S: The parameter S, which controls the magnitude of the somersault movement, is kept constant throughout the optimization process. This static setting imposes significant constraints on the adaptability of the algorithm's search behavior. During the initial optimization stages, a larger value of S would be beneficial for promoting global exploration and preventing premature convergence to local optima. Conversely, in the latter stages, a smaller value would enhance the ability to fine-tune the search around promising regions. By keeping S fixed, MRFO struggles to dynamically balance exploration and exploitation, leading to a less efficient convergence rate and ultimately compromising the quality of the obtained solution.

(b) Limited exploration mode centered on the current best individual: The somersault foraging mechanism directs all individuals to perform movements around the current best solution, x_{best}. While this approach aims to accelerate convergence by intensifying focus on promising areas of the solution space, it inherently suffers from a significant drawback: it considerably reduces population diversity over time. As individuals concentrate around x_{best}, the search space becomes increasingly constrained, leading to rapid convergence to a localized region. This reduction in diversity weakens the algorithm's capacity for thorough exploration of the solution space, increasing the risk of MRFO getting stuck in local optima. This problem becomes especially evident in complex, multimodal landscapes, where the presence of numerous local optima makes identifying the global optimum more challenging.

In conclusion, the somersault foraging mechanism's dependence on a fixed parameter S and its singular focus on the current best individual substantially limit MRFO's capacity to balance exploration and exploitation effectively. These limitations heighten the risk of premature convergence and undermine the robustness of the search process. To address these issues, enhancements could include dynamically adapting the parameter S based on the current state of the search, as well as introducing alternative movement strategies that foster greater exploration diversity. Such improvements are anticipated to bolster MRFO's adaptability, enhance convergence efficiency, and improve overall solution quality.

Proposed HGMRFO

Adaptive somersault factor (S)

In the MRFO, the somersault factor S is generally a constant value that determines the range for somersault searches centered around the current best solution. However, a fixed value for S presents significant limitations at different optimization stages: a larger S value is beneficial in the early stages to enhance exploration and discover more unknown regions, while a smaller S is more suitable for fine-tuning solutions during the local exploitation phase. Therefore, a fixed S value lacks the flexibility needed to adapt to the dynamic requirements of various optimization phases, which ultimately restricts the effectiveness of global exploration and local exploitation.

To address this limitation, an adaptive somersault factor can be introduced to replace the fixed constant S, allowing it to be dynamically adjusted based on the evolving conditions of the search process. The adaptive S is calculated as follows:

$$S = 2 \times \arctan(|f_{\text{best}} - f_{\text{global}}|) \tag{8}$$

where f_{best} represents the best fitness value among the current population, and f_{global} denotes the global best fitness value found so far. By applying the arctangent function to the absolute difference between f_{best} and f_{global}, a dynamically adjusted value of S is obtained.

The arctangent function exhibits a characteristic where its output changes rapidly for small inputs and becomes more gradual for larger inputs. This property implies that when the difference between f_{best} and f_{global} is large, the computed value of S will also be relatively large, thereby enhancing exploration ability and helping the population escape local optima. Conversely, when the difference is small, S will decrease, allowing for more focused local search and improving exploitation during the later stages of optimization.

Through this adaptive adjustment mechanism, the somersault factor S varies dynamically according to the current search state, thereby autonomously achieving an appropriate equilibrium between global exploration and local refinement. This strategy effectively addresses the challenges associated with a fixed parameter S, substantially enhancing the rate of convergence as well as the algorithm's capability to perform a thorough global search, thereby enabling it to adapt more effectively to the changing requirements of the search process across different phases.

Hierarchical guidance mechanism in MRFO

Input: Parameters N, P_{rate}, A_{rate}, d, t, T
Output: Global optimum
Initialization: Randomly initialize a population of N individuals;
Fitness evaluation: Assess fitness values and identify the optimal solution (X_{best});
while $t < T$ **do**
  for $i = 1$ *to* N **do**
    if $rand < 0.5$ **then**
      if $t/T < rand$ **then**
        Update position by Eqs. (5) and (6);
      else
        Update position by Eq. (3);
    else
      Update position by Eq. (1);
    Check the boundary of the individual;
    Calculate fitness of individual;
    if $f(X_i) < f(X_{best})$ **then**
      $X_{\text{best}} = X_i$;
  Calculate somersault factor S by Eq. (8);
  for $i = 1$ *to* N **do**
    Update position by Eq. (9);
    Check the boundary of the individual;
    Calculate fitness of individual;
    if $f(X_i) < f(X_{best})$ **then**
      $X_{\text{best}} = X_i$;
  Update the archive;
  $t = t + 1$;

Algorithm 1. HGMRFO

In the enhanced MRFO, a hierarchical structure is incorporated to strengthen both the exploratory and exploitative abilities during the somersault foraging phase, thereby forming a hierarchical guidance mechanism. This hierarchical structure consists of four primary layers: the population layer, the personal-best layer, the archive layer, and the rank-based selective pressure layer. These layers play a crucial role in guiding the somersault foraging process.

Specifically, the original somersault foraging equation in MRFO is:

$$x_i^d(t+1) = x_i^d(t) + S \cdot \left(r_2 \cdot x^d_{\text{best}} - r_3 \cdot x_i^d(t)\right), \quad i = 1, \ldots, N$$

With the introduction of the hierarchical guidance mechanism, the somersault foraging equation is modified as follows:

$$\begin{aligned} x_i^d(t+1) = x_i^d(t) + S \cdot \big(r_2 \cdot (x^d_{\text{Pbest}}(t) - x_i^d(t) + x^d_{R1}(t) - x^d_{R2}(t)) \\ - r_3 \cdot x_i^d(t)\big), \quad i = 1, \ldots, N, \quad R1 \neq R2 \neq i \end{aligned} \tag{9}$$

In this new formula, the original x^d_{best} is replaced by $x^d_{\text{Pbest}}(t) - x^d_i(t) + x^d_{R1}(t) - x^d_{R2}(t)$, thereby significantly enhancing the population's ability to converge towards the global optimum. Specifically:

(1) Population layer: This foundational layer encompasses all individuals. During the somersault foraging phase, each individual in the population layer adjusts its movement direction based on information provided by the other layers.
(2) Personal-best layer: This layer is composed of the top P_N high-quality individuals from the population and serves as a replacement for the global best individual in the original algorithm. During the somersault foraging phase, one individual $x^d_{\text{Pbest}}(t)$ is randomly selected from this layer to provide guidance for individuals within the population, thereby enhancing exploitation capability and effectively preventing convergence to local optima.
(3) Archive layer: The archive layer stores individuals that were eliminated during the greedy selection process. These individuals provide diverse search directions, enriching population diversity. By randomly selecting individuals $x^d_{R2}(t)$ from the archive layer, individuals can obtain more diverse movement directions.
(4) Rank-based selective pressure layer: In this layer, all individuals are ranked based on their fitness, and one individual $x^d_{R1}(t)$ is randomly selected to participate in the somersault foraging process. High-quality individuals have a higher probability of being selected, which helps accelerate convergence, while the occasional selection of lower-quality individuals promotes solution diversity and prevents premature convergence.

To better illustrate the working principle of the hierarchical guidance mechanism, Fig. 2 visually depicts the movement trajectory during the somersault foraging phase. In the figure, ★ represents the global optimum, and the positions of X_{Pbest} and X_{R1} are closer to the global optimum. The arrows in the figure represent the direction of movement, including vectors $r_2(X_{\text{Pbest}} - X_i)$ and $r_2(X_{R1} - X_{R2})$. The combination of these vectors results in an individual's final movement direction that tends towards the global optimum, thereby enhancing the overall search capability and convergence performance of the population.

By incorporating the hierarchical layers, the somersault foraging phase is no longer confined to the original global best individual-based update strategy. Instead, it integrates information from the personal-best layer, archive layer, and rank-based selective pressure layer to dynamically adjust the individuals' trade-off between broad exploration of the solution space and focused exploitation of promising regions. This improvement not only significantly enhances MRFO's exploration capabilities but also bolsters its exploitation potential, allowing individuals to perform a more comprehensive exploration of the solution space and effectively circumvent local optima. The pseudocode for HGMRFO is provided in Algorithm 1.

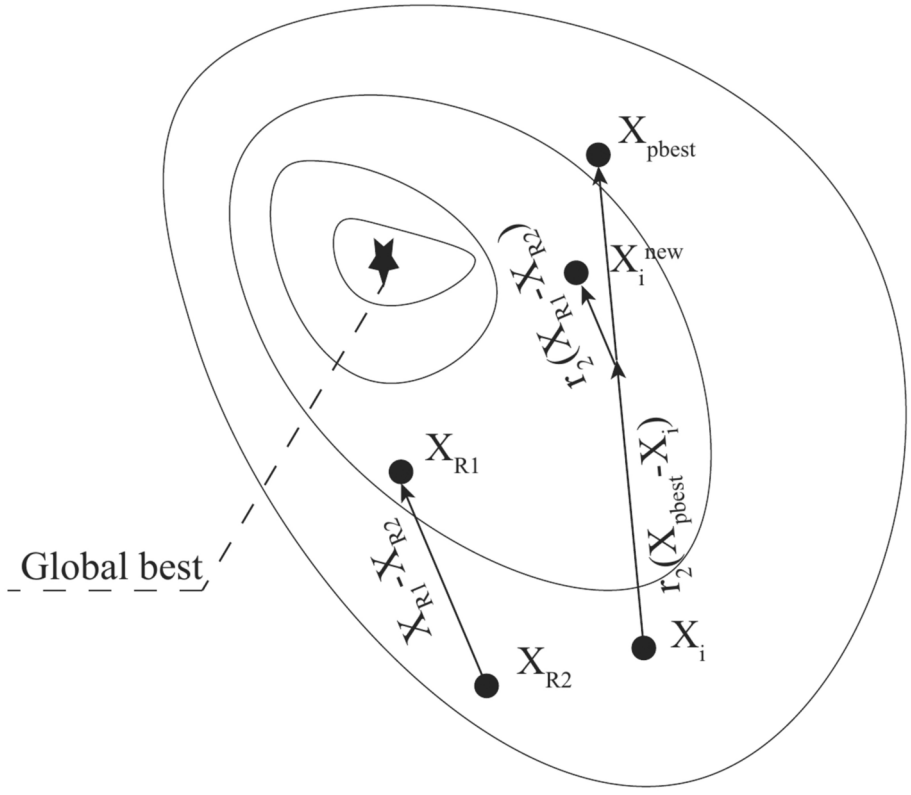


Fig. 2. Illustration of the search trajectory of HGMRFO.

Experiments and analysis

Experimental setup

To systematically evaluate the performance of HGMRFO, this study conducted simulations comparing HGMRFO with several state-of-the-art optimization algorithms on the IEEE CEC2017 benchmark function set. It should be noted that the F2 function was excluded from the experiments due to instability in the results. Additionally, to validate the practical applicability of HGMRFO, comparative experiments were carried out with other optimization algorithms on the IEEE CEC2011 real-world optimization problem set. Finally, to assess the performance of HGMRFO in parameter estimation for photovoltaic models, we selected six different photovoltaic models and compared HGMRFO with other methods for parameter estimation. For detailed descriptions of the IEEE CEC2017 benchmark function set, the IEEE CEC2011 real-world optimization problem set, and the photovoltaic models, please refer to references[2,61,62].

In this paper, we conducted an experimental comparison on the IEEE CEC2017 benchmark functions and IEEE CEC2011 real-world optimization problems. For all comparison algorithms, the population size $N = 100$, and the maximum number of function evaluations (MFEs) was 10,000D, where D represents the problem dimension. Each algorithm was run independently 51 times on each function to ensure statistical significance of the results. In the parameter estimation experiments for photovoltaic (PV) models, $N = 100$ and $MFEs = 50,000$. Each algorithm was independently run 30 times for each parameter estimation problem to thoroughly evaluate its stability and effectiveness in practical application scenarios. The experiments were conducted in MATLAB R2021b, utilizing a computer featuring an Intel Core i7-9750H processor running at 2.60 GHz and 16 GB of RAM.

For the evaluation of experimental data, three distinct criteria were employed to comprehensively assess the performance of HGMRFO, as defined below:

1) Mean optimization error and standard deviation: The performance of each algorithm is evaluated by analyzing the optimization error generated in each iteration, which measures the difference between the solution obtained by the algorithm and the known global optimum. Each algorithm is executed independently across multiple trials, and the mean and standard deviation (std) of the optimization error are calculated to assess the robustness and reliability of its performance. The algorithm with the lowest mean error value is considered the most accurate, with this value highlighted in bold to indicate the best performance among all compared algorithms.

2) Wilcoxon rank-sum test[63]: It represents a non-parametric statistical approach employed to determine if a significant difference exists between the proposed algorithm and the comparison algorithm, using a significance level of $\alpha = 0.05$. In the resulting test outcomes, the symbol "+" indicates that the proposed algorithm significantly outperforms its competitor, while the symbol "–" signifies that the proposed algorithm performs significantly worse. The symbol "≈" implies that there is no significant difference between the two algorithms. Furthermore, "W/T/L" represents the number of times the proposed algorithm wins, ties, and loses against its competitors, providing a comprehensive assessment of the relative performance of the algorithms.

3) Friedman test[64]: It is also a non-parametric statistical approach that serves to evaluate the performance of all tested algorithms over the entire problem set and generate a comprehensive ranking. In this test, the mean optimization error of each algorithm is used as the test data, with a lower Friedman rank indicating superior algorithm performance. For clarity and ease of comparison, the smallest Friedman rank is highlighted in bold, emphasizing the best-performing algorithm.

Parameter sensitivity analysis

In the HGMRFO, the number of individuals in the personal-best layer is determined by the parameter $\mathrm{P}_{\mathrm{rate}}$, calculated as $P_N = \mathrm{P}_{\mathrm{rate}} \times N$. To evaluate the impact of $\mathrm{P}_{\mathrm{rate}}$ on the algorithm's performance, this study considers 10 possible values: $\mathrm{P}_{\mathrm{rate}} \in \{0.02, 0.05, 0.1, 0.15, 0.2, 0.3, 0.4, \mathrm{rand}[0.02, 0.15], \mathrm{rand}[0.02, 0.2], \mathrm{rand}[0.02, 0.3]\}$. In the parameter sensitivity analysis, these values are used to regulate the size of the personal-best layer, enabling an assessment of the algorithm's performance on the 30-dimensional IEEE CEC2017 benchmark function set under different configurations. By comparing the optimization results across different $\mathrm{P}_{\mathrm{rate}}$ settings, we can gain insight into how this parameter influences HGMRFO's convergence performance and global search capability.

The number of individuals in the archive layer is controlled by the parameter $\mathrm{A}_{\mathrm{rate}}$, calculated as $A_N = \mathrm{A}_{\mathrm{rate}} \times N$. To comprehensively evaluate the effect of archive size on the performance of HGMRFO, $\mathrm{A}_{\mathrm{rate}}$ is set to 7 different values: $\mathrm{A}_{\mathrm{rate}} \in \{1, 1.5, 2, 2.5, 3, 3.5, 4\}$. In the experiments, these values are used to adjust the size of the archive layer, and the impact on performance is assessed using the IEEE CEC2017 benchmark function set. Proper configuration of the archive layer is crucial for maintaining population diversity and preventing premature convergence. By comparing the results for different $\mathrm{A}_{\mathrm{rate}}$ settings, we aim to identify the optimal archive size that best balances exploration and exploitation at various stages of the optimization process.

The sensitivity analysis of parameters $\mathrm{P}_{\mathrm{rate}}$ and $\mathrm{A}_{\mathrm{rate}}$ aims to determine the optimal parameter configurations that enhance the performance of HGMRFO on complex optimization problems, improve its global search ability, and increase its local exploitation precision, thereby improving the algorithm's overall solution efficiency and robustness. Tables 1 and 2 present the experimental results of HGMRFO across different parameter settings on the IEEE CEC2017 benchmark functions. Analysis of the W/T/L metrics in these tables indicates that the algorithm demonstrates limited sensitivity to the parameters $\mathrm{P}_{\mathrm{rate}}$ and $\mathrm{A}_{\mathrm{rate}}$. However, according to the Friedman test rankings, HGMRFO achieves the best performance with $\mathrm{P}_{\mathrm{rate}} = \mathrm{rand}[0.02, 0.2]$ and $\mathrm{A}_{\mathrm{rate}} = 3.5$. This implies that, although the algorithm is generally robust to variations in these parameters, specific configurations can still lead to statistically significant improvements in performance.

P_{rate}	W/T/L	Rank
0.02	5/24/0	5.931
0.05	4/25/0	4.8621
0.1	3/26/0	5.5862
0.15	1/28/0	6.1724
0.2	4/25/0	5.7586
0.3	5/24/0	5.6552
0.4	7/22/0	6.1379
rand[0.02, 0.15]	5/23/1	6.5517
rand[0.02, 0.2]	–/–/–	**3.0345**
rand[0.02, 0.3]	2/27/0	5.3103

Table 1. Statistical results of HGMRFO with P_{rate} on CEC2017 benchmark functions with 30 dimensions.

A_{rate}	W/T/L	Rank
1	6/23/0	4.8276
1.5	2/27/0	4.4138
2	2/26/1	3.5172
2.5	3/26/0	4.5862
3	3/26/0	3.8621
3.5	–/–/–	**2.6552**
4	2/27/0	4.1379

Table 2. Statistical results of HGMRFO with A_{rate} on CEC2017 benchmark functions with 30 dimensions.

Layer	W/T/L	Rank
L1	7/18/4	3.1724
L2	14/12/3	3.3448
L3	7/22/0	3.2759
L4	–/–/–	**2.1034**
L5	3/26/0	3.1034

Table 3. Statistical results of HGMRFO (i.e., L4) and other hierarchical algorithms on CEC2017 functions with 30 dimensions.

Analysis of hierarchical interactions

To comprehensively evaluate the efficacy of various hierarchical structures within the HGMRFO, five distinct hierarchical schemes, denoted as L1 through L5, were systematically compared as follows:

L1: Personal-best layer only
L2: Personal-best layer + Population layer
L3: Rank-based selective pressure layer + Archive layer
L4: Personal-best layer + Population layer + Rank-based selective pressure layer + Archive layer (i.e., HGMRFO)
L5: Current best individual + Population layer + Rank-based selective pressure layer + Archive layer

These five schemes were subjected to experimental evaluation on the CEC2017 suite to thoroughly assess their relative performances, as shown in Table 3.

The experimental results unequivocally demonstrate that L4, the complete version of HGMRFO, which integrates the personal-best layer, population layer, rank-based selective pressure layer, and archive layer, consistently outperformed the other hierarchical schemes in terms of optimization accuracy. Specifically, HGMRFO achieved a more optimal balance between global exploration and local exploitation, enhancing its capability to prevent premature convergence and obtain superior overall solutions.

Notably, the comparison between L4 and L1 underscores the critical importance of incorporating additional layers, such as the rank-based selective pressure layer and archive layer, in enhancing the robustness and search diversity of the algorithm. Similarly, comparisons with L5 reveal that leveraging a dynamically updated personal-best layer, rather than merely relying on the current best individual, provides more effective guidance, ultimately leading to enhanced optimization outcomes.

The comparison between L4 and L2 highlights the impact of adding both the rank-based selective pressure layer and the archive layer. While L2 already incorporates the population layer to complement the personal-best layer, L4 further enhances the search process by utilizing rank-based selective pressure to guide the population more effectively towards promising regions and by incorporating an archive layer to maintain search diversity. This combination enhances resilience against early convergence and optimizes the utilization of the solution space.

Furthermore, the comparison between L4 and L3 reveals the critical role of the personal-best layer in conjunction with the rank-based selective pressure layer and archive layer. L3, which lacks the personal-best layer, showed limited guidance efficiency, leading to suboptimal convergence. By incorporating the personal-best layer, L4 substantially enhanced the algorithm's capacity to fine-tune the exploration process and attain superior-quality solutions, underscoring the importance of integrating all four layers to achieve optimal performance.

Overall, the experimental analysis substantiates that the complete hierarchical configuration of HGMRFO (L4) significantly improves the algorithm's performance, emphasizing the importance of each layer in facilitating a comprehensive and effective exploration of the solution space.

Comparison for competitive algorithms

To rigorously evaluate the performance of the HGMRFO, comparative experiments were conducted against several state-of-the-art algorithms, including the original MRFO[13], the fractional-order Caputo MRFO (FCMRFO)[27], chaotic wind driven optimization with fitness distance balance strategy (CFDBWDO)[65], spatial information sampling algorithm (SIS)[66], reptile search algorithm (RSA)[67], the fractional-order flower pollination algorithm (FOFPA)[68], and spherical evolution (SE)[69], on the IEEE CEC2017 benchmark functions with dimensions of 10, 30, 50, and 100. The statistical results derived from these experiments are presented in Table 5, with the experimental data provided in Tables 8, 9, 10, and 11, while the specific parameter settings for each algorithm are detailed in Table 4.

In the statistical results presented in Table 5, HGMRFO demonstrated superior performance across various dimensions of the IEEE CEC2017 benchmark functions. Specifically, in the 10-dimensional benchmark, HGMRFO outperformed MRFO, FCMRFO, CFDBWDO, SIS, RSA, FOFPA, and SE on 15, 11, 27, 22, 29, 23, and 13 problems, respectively, achieving the top rank according to the Friedman test. These results indicate that HGMRFO exhibits remarkable performance in low-dimensional optimization tasks.

In the 30-dimensional benchmark, HGMRFO outperformed the aforementioned algorithms on 21, 19, 28, 21, 28, 24, and 17 problems, respectively, with the highest Friedman ranking. This signifies the strong competitiveness of HGMRFO in medium-dimensional problems.

For the 50-dimensional benchmark, the W/T/L (Wins/Ties/Losses) metric demonstrates that HGMRFO performed better than MRFO, FCMRFO, CFDBWDO, SIS, RSA, FOFPA, and SE on 17, 15, 27, 18, 29, 26, and 17 problems, respectively. It also achieved the smallest Friedman rank, indicating that HGMRFO excels in high-dimensional optimization scenarios.

In the 100-dimensional benchmark, HGMRFO outperformed the competing algorithms on 17, 17, 25, 18, 28, 24, and 18 problems, respectively, while maintaining the smallest rank value, thereby showcasing its effectiveness in ultra-high-dimensional optimization tasks.

Algorithms	Year	Parameters
HGMRFO	2024	$\mathrm{P_{rate}} = \mathrm{rand}[0.02, 0.2], \mathrm{A_{rate}} = 3.5$
MRFO	2020	$S = 2$
FCMRFO	2022	$\alpha = 1$
HMRFO	2023	$S = 2, \mu = 3/4, \sigma = 1/12, PR_1 = 0.8, PR_2 = 0.6$
CFDBWDO	2022	$\alpha = 0.4, g = 0.2, RT = 3, c = 0.4, w = 0.5, r = 0.0001, L = 50, \varepsilon = 0.01, J = 12$
SIS	2022	$b \in (0, 0.1]$
RSA	2022	$\alpha = 0.1, \beta = 0.005$
FOFPA	2020	$S = 0.8, \lambda = 1.5, \alpha = 0.6, r = 8$
SE	2019	DSF = $[5 \sim 10]$
disGSA	2023	$G_0 = 100, \alpha = 20, k = -900, s = 8$
AHA	2022	$M_c = 2n$
DGSA	2021	$G_0 = 100, D_N = 5$
XPSO	2020	$\sigma = 0.1, Stag_{max} = 5, \eta = 0.2, p = 0.5$
FOWFO	2024	$e = 12, \epsilon = 0.9999999, P_e = 0.7$
WFO	2022	$p_l = 0.3, p_e = 0.7$
WDO	2010	$\alpha = 0.4, g = 0.2, RT = 3, c = 0.4, u_{\max} = 0.1$
GWO	2014	α linearly decreases from 2 to 0

Table 4. Parameter settings of HGMRFO and various algorithms on IEEE CEC2017, IEEE CEC2011, and PV models.

	Dim=10		Dim=30		Dim=50		Dim=100	
Algorithms	**W/T/L**	**Rank**	**W/T/L**	**Rank**	**W/T/L**	**Rank**	**W/T/L**	**Rank**
HGMRFO	–/–/–	**2.4138**	–/–/–	**1.931**	–/–/–	**2.2414**	–/–/–	**2.4828**
MRFO	15/9/5	4.3103	21/5/3	4.2069	17/8/4	4.069	17/6/6	3.931
FCMRFO	11/14/4	3.2414	19/6/4	3.1379	15/9/5	3.1724	17/6/6	3.2069
CFDBWDO	27/2/0	6.1379	28/0/1	6.6897	27/0/2	6.6552	25/1/3	6.5172
SIS	22/5/2	4.9828	21/3/5	3.7586	18/3/8	3.2069	18/4/7	3.1379
RSA	29/0/0	7.9655	28/1/0	7.9655	29/0/0	7.8966	28/1/0	7.931
FOFPA	23/2/4	4.1379	24/4/1	5	26/2/1	5.3448	24/1/4	5.1724
SE	13/4/12	2.8103	17/6/6	3.3103	17/6/6	3.4138	18/2/9	3.6207

Table 5. Statistical results of HGMRFO and seven algorithms on CEC2017 functions.

The statistical analysis across different dimensions conclusively demonstrates that HGMRFO consistently achieves superior performance, affirming its robustness and high efficiency in handling complex optimization problems of varying dimensionality.

In Table 8, HGMRFO achieved the best solution on 6 out of 10-dimensional functions, while SE obtained the best solution on 12 functions, ranking first in terms of the number of best solutions, with HGMRFO ranking second. Therefore, for low-dimensional functions, the proposed HGMRFO still requires further improvement in future research.

In Tables 9, 10, and 11, HGMRFO achieved the best solution on 15 out of 30-dimensional functions, 10 out of 50-dimensional functions, and 11 out of 100-dimensional functions, respectively, ranking first in terms of the number of best solutions obtained. Thus, HGMRFO demonstrates the best performance on 30-dimensional, 50-dimensional, and 100-dimensional functions.

Fig. 4 shows the convergence curves of the average optimization errors for HGMRFO and other competing algorithms on the F8 and F11 functions across 10-dimensional, 30-dimensional, 50-dimensional, and 100-dimensional spaces. As can be seen from the figure, HGMRFO demonstrates rapid convergence and consistently finds the optimal solution in all dimensions. Compared to the original MRFO algorithm, the figure clearly illustrates the effectiveness of the hierarchical guidance mechanism proposed in this paper.

The computational complexity of each iteration in the HGMRFO algorithm is primarily determined by the following components:

(1) Initialization: $O(N \times D)$.
(2) Fitness evaluation: $O(N)$.
(3) Best individual update: $O(N)$.
(4) Position update: $O(N \times D)$.
(5) Boundary detection: $O(N \times D)$.
(6) Archive update: $O(N \times D)$.
(7) Probability calculation and selection: $O(N)$.

Therefore, the overall computational complexity for a single iteration is: $O(N \times D)$. If the algorithm runs for T iterations, the total computational complexity becomes: $O(T \times N \times D)$.

Furthermore, Fig. 3 illustrates a bar chart that shows the computational running times for each algorithm tested on the IEEE CEC2017 benchmark set with dimensions of 10, 30, 50, and 100. As illustrated in Fig. 3, both HGMRFO and MRFO exhibit the shortest CPU running times across all dimensions, with computational complexities that are nearly identical. These results suggest that the introduction of the hierarchical guidance structure in HGMRFO imposes negligible computational overhead on MRFO, thus affirming the scalability and efficiency of HGMRFO. Additionally, the low computational cost of HGMRFO highlights its potential for application in high-dimensional and intricate engineering optimization problems.

Real-world optimization problems

To comprehensively assess the practicality of the HGMRFO, we conducted extensive simulation experiments comparing HGMRFO with eight state-of-the-art algorithms on real-world optimization problems from the IEEE CEC2011 benchmark suite. The algorithms considered in this comparison include: MRFO[13], FCMRFO[27], HMRFO[12], disGSA[70], AHA[71], SIS[66], DGSA[72], and XPSO[73]. Among them, HMRFO is an hierarchical MRFO with weighted fitness-distance balance selection. disGSA is an adaptive position-guided GSA. AHA denotes the artificial hummingbird algorithm. DGSA is a GSA with hierarchy and distributed framework. XPSO is an expanded PSO based on multi-exemplar and forgetting ability. The specific parameter settings for each algorithm are detailed in Table 4. The experimental data and statistical results are presented in Tables 6 and 7.

From the results summarized in Tables 6 and 7, it can be observed that HGMRFO achieved the best mean values on eight distinct problems, outperforming the competing algorithms by achieving the highest number of optimal solutions. Furthermore, the W/T/L analysis indicates that HGMRFO outperformed MRFO, FCMRFO, HMRFO, disGSA, AHA, SIS, DGSA, and XPSO on 7, 8, 7, 12, 14, 16, 11, and 11 problems, respectively. These results highlight the effectiveness and practical applicability of HGMRFO in addressing real-world optimization challenges.

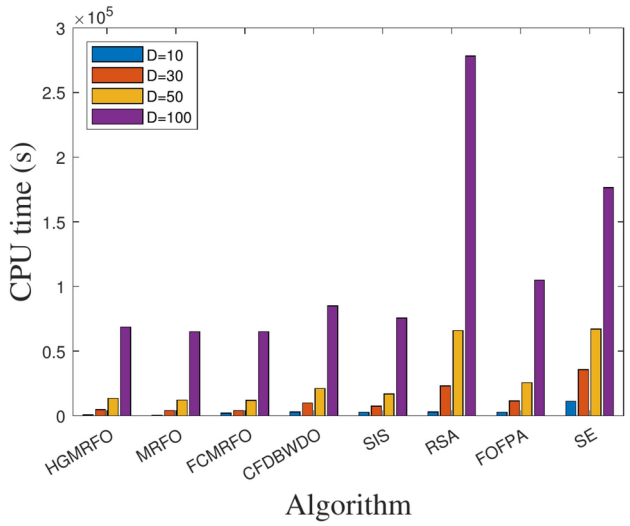


Fig. 3. Bar chart depicting CPU running time.

	HGMRFO			MRFO			FCMRFO			HMRFO		
	Mean	**Std**		**Mean**	**Std**		**Mean**	**Std**		**Mean**	**Std**	
F1	**7.5456799**	6.063434487		11.52754644	6.671964153	+	12.16287364	6.492594563	+	7.910302253	6.964252096	≈
F2	− 23.01947004	3.20106643		− 27.05577523	0.789345135	−	− 26.97969875	0.708184122	−	**-27.1136843**	0.697683597	−
F3	1.15149E−05	3.72892E−19		1.15149E−05	3.5452E−19	≈	1.15149E−05	4.54812E−19	≈	1.15149E−05	4.74335E−19	≈
F4	**14.62881228**	0.404634964		15.09523461	1.19169975	+	15.26533098	1.615056422	+	14.97962873	0.903800601	+
F5	− 29.70145264	3.963645014		− 33.62315824	2.362654249	−	− 33.93982932	2.515292796	−	− 33.68356837	2.325791771	−
F6	− 23.48964581	3.018486		− 22.92609813	3.558278066	≈	− 23.5420097	3.086914794	−	− 23.51322387	3.493240166	≈
F7	1.558316585	0.230807357		1.080087031	0.219278802	−	1.225713566	0.208887407	−	1.421367953	0.256899262	−
F8	**220**	0		**220**	0	≈	**220**	0	≈	**220**	0	≈
F9	1684.88436	390.0363253		3299.976318	1873.12238	+	2189.072421	642.1303436	+	3201.710113	4175.863125	+
F10	− 21.12672575	0.270788485		− 21.06051286	0.718593846	≈	− 21.16441676	0.553417875	−	− 21.22538369	0.377189433	−
F11	52281.72083	408.6091835		52361.21988	507.4312215	≈	52342.33079	573.7655236	≈	**y51083.92565**	451.5186784	−
F12	**17438772.08**	53271.73069		17473244.28	77794.28038	+	17462065.8	64567.29224	+	17467272.7	74181.07783	+
F13	**15456.00698**	7.566640902		15463.53821	15.82038458	+	15465.91837	15.26447283	+	15463.6698	17.14061091	+
F14	19389.92521	187.3404768		19451.53642	230.4080043	≈	19433.67346	209.9553977	≈	19518.98255	289.4900846	+
F15	**33010.606**	42.64311277		33020.68977	61.77541827	≈	33014.30516	55.1280023	≈	33011.61616	41.22291414	≈
F16	**137129.1608**	1949.311177		139708.8363	3362.010505	+	138723.1336	3334.506386	+	139042.6578	2837.210139	+
F17	1962074.985	20707.96482		2004437.071	173135.5062	≈	1980867.385	51581.73882	+	1985538.538	72309.70451	≈
F18	958377.509	15100.37473		962834.2717	63699.63641	≈	954842.1609	26778.40648	−	961566.6442	22354.7607	≈
F19	1349038.031	100020.3833		1334130.856	165484.7438	−	1346747.927	152401.1149	≈	1286675.208	121693.9524	−
F20	956393.9212	15074.20454		950693.625	13937.1094	−	951034.4423	10300.26768	≈	964528.4917	29455.25172	≈
F21	14.250842	1.978687815		13.94759085	2.50107824	≈	14.63627829	2.68771994	≈	**13.35929869**	2.844019668	≈
F22	**20.06059281**	2.259480694		21.68497396	3.118904942	+	21.20452144	2.816317233	+	20.96414888	2.730117268	+
W/T/L	−/−/−			7/10/5			8/8/6			7/9/6		

Table 6. Analysis and comparative performance of HGMRFO and eight competitive algorithms on IEEE CEC2011 real-world optimization problems.

The demonstrated superior performance, combined with the low computational overhead of HGMRFO, motivated us to apply this algorithm to a practical engineering problem-parameter estimation in solar photovoltaic models.

Comparison on parameter estimation of photovoltaic model

To thoroughly analyze the nonlinear output characteristics of photovoltaic (PV) systems, the adoption of accurate mathematical models for PV cells is crucial. The output characteristics of PV cells are significantly influenced by various factors, including temperature and internal as well as external circuit parameters. Consequently, the

	disGSA			AHA			SIS			DGSA			XPSO		
	Mean	**Std**		**Mean**	**Std**		**Mean**	**Std**		**Mean**	**Std**		**Mean**	**Std**	
F1	17.47391973	5.470206253	+	14.49837455	8.171808684	+	18.39017045	4.832461831	+	15.36003099	5.171082772	+	11.87523968	5.524929846	+
F2	− 26.3244037	1.656718405	−	− 26.88276512	0.99276452	−	− 16.18868877	4.294438424	+	− 25.94040242	2.455154294	−	− 26.62955583	0.749051232	−
F3	1.15149E−05	5.42962E−19	+	1.15149E−05	2.25642E−13	≈	1.15149E−05	3.15322E−16	+	1.15149E−05	1.66472E−12	+	**-0.00016673**	0.001272924	−
F4	14.63862887	1.455780439	−	16.40534169	2.437398369	+	19.6290455	3.270347476	+	16.36766023	2.308586321	+	14.9065653	1.800079766	−
F5	− 33.09055389	1.913171334	−	− 33.23537396	2.298288735	−	− 32.8725704	2.805451308	−	− 33.47936543	1.744126034	−	**-34.38493325**	1.869106511	−
F6	− 21.75572359	2.998252639	≈	**-25.80920512**	2.896806136	−	− 13.71199977	1.259428287	+	− 21.5756295	2.988086601	+	− 24.63935566	3.217541092	−
F7	0.902931326	0.16389891	−	1.363284765	0.203944019	−	0.902780281	0.297667563	−	**0.679256235**	0.140544396	−	0.862423633	0.137602163	−
F8	222.2352941	6.932786555	+	**220**	0	≈	226.1960784	9.325276635	+	220.1960784	1.400280084	≈	222.5598923	18.28128751	≈
F9	1589904.934	159370.1404	+	2135.206489	993.4579601	+	168422.6584	52202.81824	+	**1283.549936**	840.0984967	−	19127.71596	20848.31813	+
F10	− 18.12383684	1.624224768	+	− 15.49218405	2.975753264	+	− 15.83479707	5.102266813	+	**-21.43541163**	0.068391611	−	− 20.60571549	1.872118363	+
F11	52058.8457	501.9985769	−	52529.47876	505.5468325	+	52610.30649	605.8632139	+	52119.66365	585.5883124	−	52418.09374	612.1166024	≈
F12	25182957.66	459011.4329	+	17440882.59	56153.82026	≈	23000560.47	494246.0606	+	23783090.49	1097011.939	+	20710781.14	312942.3379	+
F13	15486.55437	25.32639738	+	15468.22705	19.4346151	+	15478.33871	18.48017631	+	20987.3529	10339.77788	+	15467.65744	16.77429917	+
F14	19204.57794	176.424332	−	**18932.32268**	234.841745	−	19293.25358	197.0702334	−	19148.02808	147.896134	−	19169.49537	145.1695674	−
F15	34815.04219	9335.104487	+	33098.41705	88.44948903	+	33139.7153	71.13062863	+	33209.61569	36.76770821	+	33077.44101	78.75436034	+
F16	140010.8218	1792.983414	+	141479.4314	5022.125221	+	140007.4902	3321.066885	+	139806.732	1628.401082	+	138409.456	3958.554788	≈
F17	**1942026.469**	6446.837188	−	2871549.196	1311885.532	+	2011168.457	105698.1168	+	2367006.454	486255.7322	+	1977864.331	56206.98396	≈
F18	1110518.393	503000.1294	−	1258159.344	420559.2879	+	951569.1447	31903.50043	−	**944986.0217**	2523.787517	−	4474834.607	645974.5764	+
F19	1784687.323	659327.8104	+	1820408.143	347330.8087	+	**1116859.444**	125359.5343	−	1125227.721	107804.1035	−	4995847.583	565237.9004	+
F20	1312959.317	1299154.349	≈	1198325.572	351096.6118	+	947681.2809	2995.376903	−	**945006.8768**	2615.209042	−	4607782.125	749556.5714	+
F21	18.39059825	3.209004489	+	19.94380741	4.515853198	+	18.50395419	3.316156684	+	15.65184742	1.738959283	+	15.31140339	2.333974974	+
F22	30.88547681	5.371732874	+	23.97320647	4.397468261	+	22.62070558	2.348901878	+	23.64428422	1.889889231	+	21.46801535	2.180518032	+
W/T/L	12/2/8			14/3/5			16/0/6			11/1/10			11/4/7		

Table 7. Analysis and comparative performance of HGMRFO and eight competitive algorithms on IEEE CEC2011 real-world optimization problems (continue).

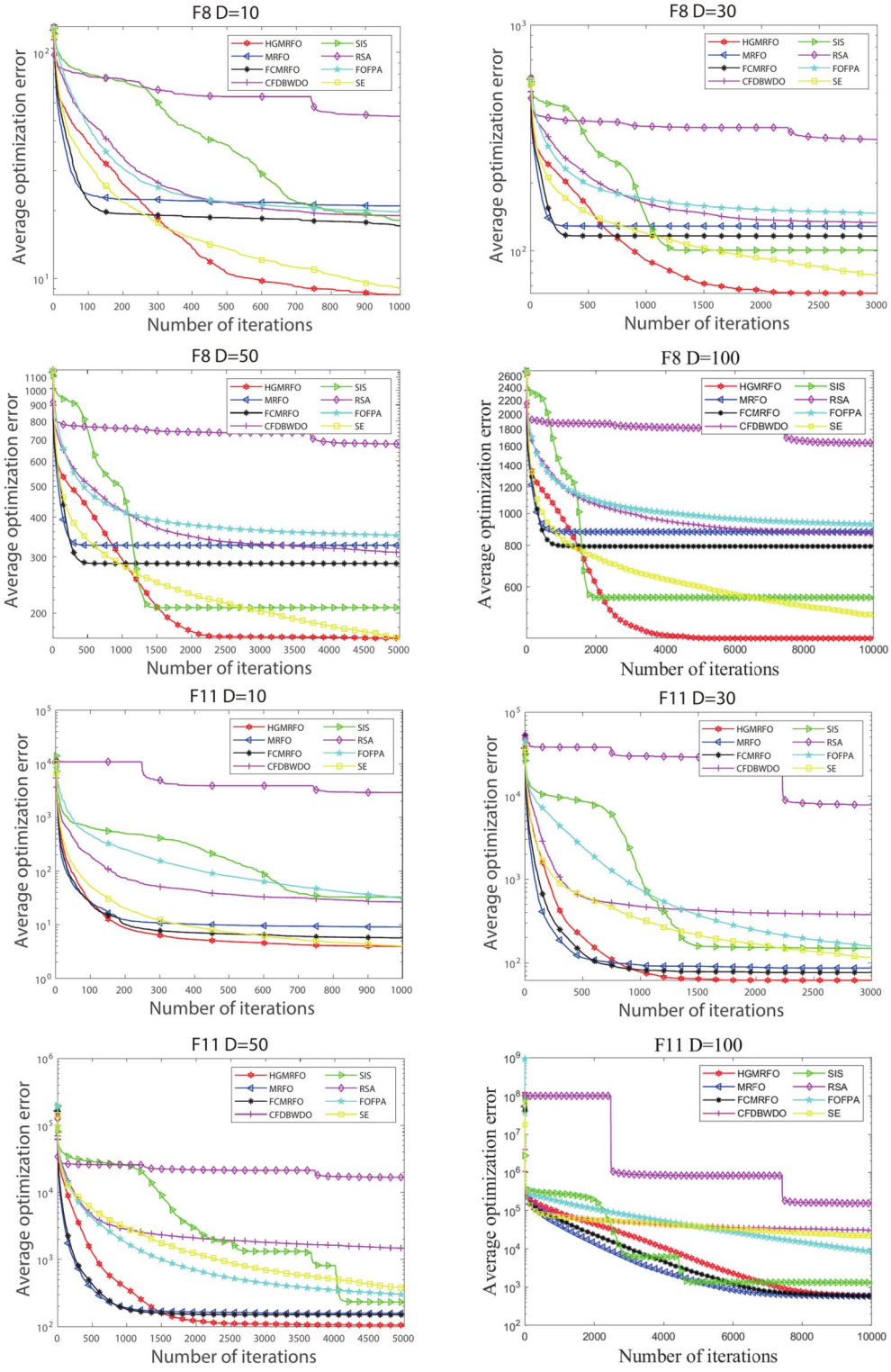


Fig. 4. Convergence curves of average optimization error performance by eight algorithms on F8 and F11 (dimensions 10, 30, 50 and 100).

development of precise mathematical models is essential for predicting system performance, optimizing system design, and improving energy conversion efficiency. To meet diverse research and application requirements, various PV models have been developed, spanning from simple to more complex structures. These models include the single diode model (SDM)[74], the double diode model (DDM)[75], the triple diode model (TDM)[76], and more complex PV module models (MM)[77]. The structures of these models are shown in Fig. 5, with a detailed description provided below.

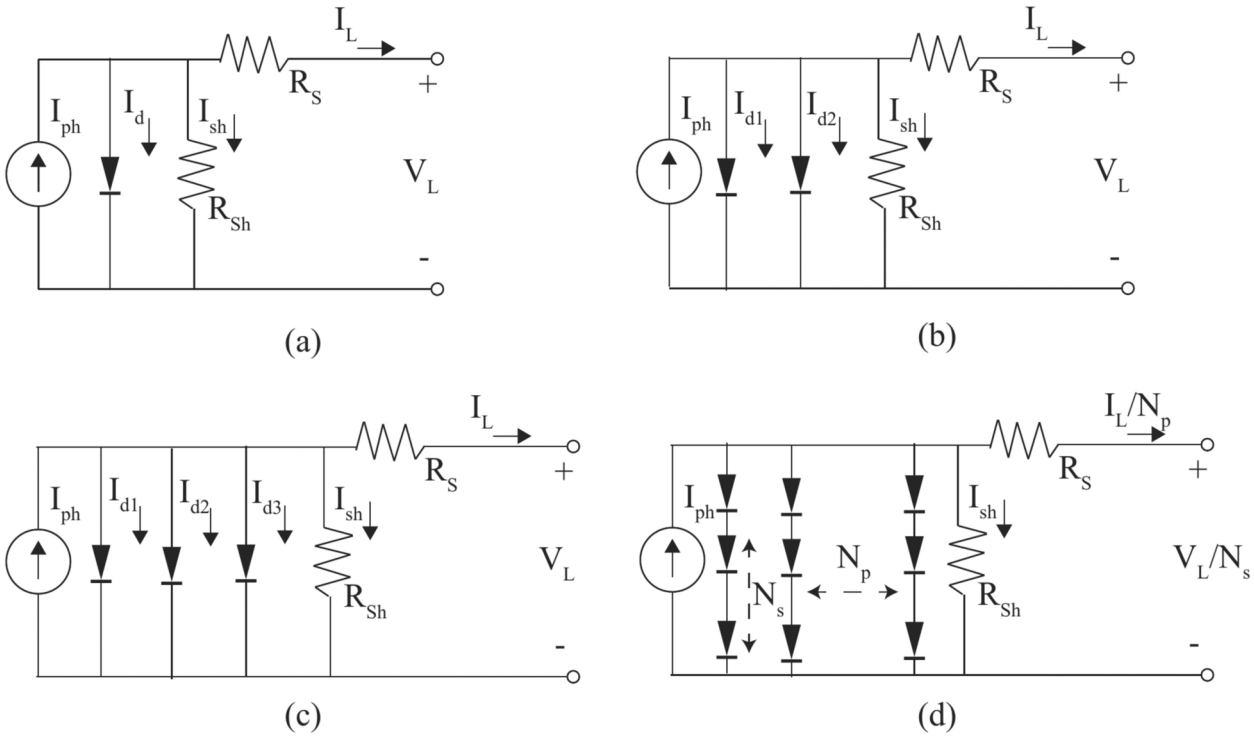


Fig. 5. Photovoltaic equivalent circuit: (a) single diode model, (b) double diode model, (c) triple diode model, and (d) photovoltaic module model.

The output current I_L of the SDM can be determined using the equation below:

$$I_L = I_{ph} - I_d - I_{sh} = I_{ph} - I_{sd}\left[\exp\left(\frac{q(V_L + R_s I_L)}{akT}\right) - 1\right] - \frac{V_L + R_s I_L}{R_{sh}} \tag{10}$$

where, I_{ph} denotes the photo-generated current, which is the current generated due to illumination, I_d is the diode current, I_{sh} is the current through the shunt resistance, I_{sd} denotes the reverse saturation current of the diode, V_L is the output voltage, R_s is the series resistance, R_{sh} is the shunt resistance, a is the diode ideality factor, q is the electron charge ($1.60217646 \times 10^{-19}$ C), k is the Boltzmann constant ($1.3806503 \times 10^{-23}$ J/K), and T is the cell temperature in Kelvin. In the SDM, five parameters need to be estimated: I_{ph}, I_{sd}, R_s, R_{sh}, and a.

The output current I_L of the DDM can be determined using the equation below:

$$I_L = I_{\mathrm{ph}} - I_{d1} - I_{d2} - I_{\mathrm{sh}} = I_{\mathrm{ph}} - I_{\mathrm{sd1}}\left[\exp\left(\frac{q(V_L + R_s I_L)}{a_1 kT}\right) - 1\right] - I_{\mathrm{sd2}}\left[\exp\left(\frac{q(V_L + R_s I_L)}{a_2 kT}\right) - 1\right] - \frac{V_L + R_s I_L}{R_{\mathrm{sh}}} \tag{11}$$

where I_{sd1} is the diffusion current, I_{sd2} is the saturation current, and a_1 and a_2 are the ideality factors of the two diodes. In the DDM, seven parameters need to be estimated: I_{ph}, I_{sd1}, I_{sd2}, R_s, R_{sh}, a_1 and a_2.

The output current I_{L} of the TDM can be determined using the equation below:

$$\begin{aligned} I_L = I_{\mathrm{ph}} - I_{d1} - I_{d2} - I_{d3} - I_{\mathrm{sh}} = I_{\mathrm{ph}} - I_{\mathrm{sd1}}\left[\exp\left(\frac{q(V_L + R_s I_L)}{a_1 kT}\right) - 1\right] - I_{\mathrm{sd2}}\left[\exp\left(\frac{q(V_L + R_s I_L)}{a_2 kT}\right) - 1\right] \\ - I_{\mathrm{sd3}}\left[\exp\left(\frac{q(V_L + R_s I_L)}{a_3 kT}\right) - 1\right] - \frac{V_L + R_s I_L}{R_{\mathrm{sh}}} \end{aligned} \tag{12}$$

where I_{sd3} is leakage current. In the TDM, nine parameters need to be estimated: I_{ph}, I_{sd1}, I_{sd2}, I_{sd3}, R_s, R_{sh}, a_1, a_2 and a_3.

The PV current of MM can be determined by the following equation:

$$\frac{I_L}{N_p} = I_{\mathrm{ph}} - I_{\mathrm{sd}}\left[\exp\left(\frac{q\left(\frac{V_L}{N_s} + R_s\frac{I_L}{N_p}\right)}{akT}\right) - 1\right] - \frac{\frac{V_L}{N_s} + R_s\frac{I_L}{N_p}}{R_{\mathrm{sh}}} \tag{13}$$

	HGMRFO			MRFO			FCMRFO			CFDBWDO		
	Mean	Std		Mean	Std		Mean	Std		Mean	Std	
F1	**8.6683E+02**	1.2267E+03		2.1391E+03	2.2607E+03	+	1.4390E+03	1.6856E+03	+	2.9078E+08	3.2182E+08	+
F3	1.6099E−07	3.5662E−07		**0.0000E+00**	0.0000E+00	−	**0.0000E+00**	0.0000E+00	−	9.6125E+01	3.4366E+01	+
F4	1.0595E+00	6.4060E−01		4.5960E−01	3.0959E−01	−	8.0051E−01	5.8132E−01	−	2.6648E+01	− 8.6780E+01	+
F5	9.6075E+00	4.2259E+00		1.8975E+01	9.2856E+00	+	1.3435E+01	7.1257E+00	+	3.9411E+01	− 8.6301E+01	+
F6	3.3411E−05	1.3797E−04		1.3743E−01	5.2944E−01	≈	6.9261E−06	3.5272E−05	≈	9.8519E+00	− 9.6736E+01	+
F7	2.1261E+01	5.8644E+00		3.8274E+01	1.3229E+01	+	2.9335E+01	9.5837E+00	+	4.0592E+01	− 8.6953E+01	+
F8	**8.4670E+00**	4.4534E+00		2.0916E+01	7.8338E+00	+	1.7061E+01	7.9085E+00	+	1.8946E+01	− 9.2168E+01	+
F9	1.7817E−02	8.9065E−02		7.3773E−01	1.6240E+00	+	7.4777E−02	1.6626E−01	+	3.3138E+01	− 6.6699E+01	+
F10	8.9371E+02	3.6271E+02		6.9056E+02	2.8553E+02	−	5.7381E+02	2.4583E+02	−	9.8173E+02	2.2739E+02	≈
F11	**3.9538E+00**	2.7002E+00		9.0631E+00	7.2642E+00	+	5.6738E+00	4.3209E+00	+	2.6683E+01	− 7.9023E+01	+
F12	**1.0481E+04**	9.3136E+03		1.4007E+04	1.4340E+04	≈	1.3380E+04	1.1107E+04	≈	3.6156E+06	5.5755E+06	+
F13	8.1528E+02	6.5802E+02		**5.5364E+02**	3.9581E+02	−	7.4775E+02	6.5096E+02	≈	1.3126E+04	1.3402E+04	+
F14	5.4780E+01	1.2855E+01		5.3370E+01	1.4746E+01	≈	5.5322E+01	1.3296E+01	≈	1.5092E+02	2.9695E+01	+
F15	1.1020E+02	6.2245E+01		**8.3502E+01**	4.7708E+01	−	1.0481E+02	5.6166E+01	≈	3.8884E+02	1.8017E+02	+
F16	4.7362E+00	1.8420E+01		4.4551E+01	7.2817E+01	≈	3.4733E+01	6.7734E+01	≈	9.6849E+01	− 2.8928E+01	+
F17	2.1034E+01	1.4086E+01		3.6949E+01	2.6993E+01	+	2.4964E+01	1.7624E+01	≈	5.2097E+01	− 8.1448E+01	+
F18	3.0874E+03	2.8832E+03		3.8506E+03	4.1892E+03	≈	3.3755E+03	4.5487E+03	≈	9.6431E+04	1.3356E+05	+
F19	**5.0447E+01**	3.6361E+01		6.8366E+01	4.0008E+01	+	7.7273E+01	9.4133E+01	+	8.8857E+02	1.2132E+03	+
F20	5.6400E+00	7.4941E+00		1.0803E+01	1.0352E+01	+	7.8981E+00	9.8061E+00	≈	7.0253E+01	− 7.5543E+01	+
F21	1.0008E+02	4.3074E−01		1.1221E+02	3.4553E+01	≈	**9.6376E+01**	1.9681E+01	≈	1.0635E+02	− 9.5544E+01	+
F22	**8.5595E+01**	3.3210E+01		9.5237E+01	2.0674E+01	≈	9.6959E+01	1.9299E+01	≈	1.0640E+02	− 7.9196E+01	+
F23	3.1220E+02	5.4232E+00		3.2371E+02	1.1675E+01	+	**2.9770E+02**	7.5336E+01	+	3.3531E+02	1.4766E+01	+
F24	1.6727E+02	1.1556E+02		2.4005E+02	1.2523E+02	+	2.1898E+02	1.2736E+02	+	**1.1414E+02**	− 7.5293E+01	+
F25	4.2531E+02	2.2570E+01		4.2692E+02	2.2906E+01	≈	4.3201E+02	2.0885E+01	+	4.2233E+02	− 7.9955E+01	≈
F26	2.9504E+02	4.2654E+01		3.2154E+02	7.0915E+01	+	2.9101E+02	4.9507E+01	≈	4.1408E+02	− 2.4751E+01	+
F27	3.9532E+02	2.3788E+00		4.0115E+02	1.3473E+01	+	3.9973E+02	6.4132E+00	+	4.4057E+02	− 9.0851E+01	+
F28	3.1168E+02	5.8445E+01		4.0899E+02	1.3246E+02	+	3.6977E+02	1.2028E+02	≈	4.0513E+02	− 2.0984E+01	+
F29	2.6882E+02	1.6980E+01		2.8829E+02	3.4224E+01	+	2.6969E+02	2.4738E+01	≈	3.3722E+02	− 6.2970E+01	+
F30	2.0845E+04	8.4126E+04		2.6334E+05	4.8761E+05	≈	1.2791E+05	3.4531E+05	−	7.8889E+05	8.6737E+05	+
W/T/L	−/−/−			15/9/5			11/14/4			27/2/0		

	SIS			RSA			FOFPA			SE		
	Mean	Std		Mean	Std		Mean	Std		Mean	Std	
F1	2.8508E+03	3.2684E+03	+	1.0959E+10	3.5843E+09	+	3.3990E+07	3.1633E+07	+	9.4515E+02	1.0747E+03	+
F3	1.8415E+03	4.1076E+03	+	8.6206E+03	2.4497E+03	+	1.9081E+02	6.5715E+01	+	5.7662E+02	2.3915E+02	+
F4	**3.3205E−01**	1.9930E−01	−	7.5599E+02	4.8506E+02	+	2.8533E+00	1.8370E+00	+	5.6375E+00	4.8616E−01	+
F5	1.7357E+01	9.2791E+00	+	9.0559E+01	1.4777E+01	+	2.6142E+01	5.4053E+00	+	**8.6108E+00**	1.5538E+00	≈
F6	6.3752E−02	2.3733E−01	+	4.8054E+01	8.0604E+00	+	1.8584E+01	3.6507E+00	+	**1.5604E−14**	3.9511E−14	−
F7	2.7258E+01	6.6245E+00	+	1.0736E+02	9.9270E+00	+	3.6236E+01	5.2421E+00	+	**2.0692E+01**	1.7287E+00	≈
F8	1.8040E+01	7.6703E+00	+	5.2143E+01	6.7164E+00	+	1.9637E+01	4.0122E+00	+	9.0938E+00	1.6881E+00	+
F9	6.0602E−02	2.6142E−01	≈	6.4497E+02	2.9999E+02	+	1.3895E+02	4.4131E+01	+	**2.3718E−10**	1.6133E−09	+
F10	9.0434E+02	4.2094E+02	≈	1.5393E+03	2.3004E+02	+	6.7542E+02	9.6887E+01	−	**4.0393E+02**	1.1324E+02	−
F11	3.2850E+01	2.3526E+01	+	2.9215E+03	2.0410E+03	+	3.1319E+01	9.3211E+00	+	4.0058E+00	8.1692E−01	≈
F12	1.4551E+06	9.0130E+06	+	5.4526E+08	4.5993E+08	+	1.7481E+04	1.0061E+04	+	3.8420E+05	2.0437E+05	+
F13	1.3000E+04	1.1576E+04	+	3.1522E+07	3.9151E+07	+	6.0343E+02	3.0963E+02	≈	1.4072E+03	9.0899E+02	+
F14	1.7482E+02	9.3844E+01	+	3.3118E+03	6.8091E+03	+	9.4497E+01	3.4725E+01	+	**3.0181E+01**	2.6405E+01	−
F15	2.4609E+03	1.4848E+03	+	7.8233E+03	8.6978E+03	+	2.7223E+02	1.5065E+02	+	1.3021E+02	1.9794E+02	−
F16	8.2981E+01	8.5457E+01	+	4.7874E+02	1.0948E+02	+	3.6972E+01	3.0550E+01	+	**1.7444E+00**	3.3481E−01	+
F17	4.9245E+01	2.5084E+01	+	1.4140E+02	4.5116E+01	+	3.5911E+01	5.6231E+00	+	**3.4226E+00**	9.1089E−01	−
F18	1.3725E+04	9.4649E+03	+	1.8568E+07	2.1789E+07	+	**5.9357E+02**	3.5131E+02	−	1.8131E+03	1.2739E+03	−
F19	7.0635E+02	1.2398E+03	+	5.4020E+05	1.2020E+06	+	1.0529E+02	7.1417E+01	+	1.3103E+02	4.0827E+02	−
F20	7.3304E+01	3.4593E+01	+	2.7456E+02	6.1862E+01	+	6.0629E+01	1.1664E+01	+	**1.0306E+00**	4.7335E−01	−
F21	1.2272E+02	2.9913E+01	+	1.6398E+02	4.0283E+01	+	1.0779E+02	3.4707E+00	+	1.3544E+02	4.0624E+01	+
F22	9.6542E+01	2.0003E+01	≈	9.2294E+02	3.1656E+02	+	9.5384E+01	2.0281E+01	+	1.0022E+02	3.3708E−01	−

Continued

	SIS			RSA			FOFPA			SE		
	Mean	Std		Mean	Std		Mean	Std		Mean	Std	
F23	3.1782E+02	8.1816E+00	+	3.9339E+02	1.1618E+01	+	3.1227E+02	4.1383E+01	+	3.0940E+02	2.0349E+00	–
F24	3.0932E+02	5.3478E+01	+	4.3425E+02	3.5143E+01	+	1.2101E+02	1.2597E+01	+	2.4729E+02	1.0364E+02	+
F25	4.1516E+02	2.2212E+01	–	9.8654E+02	2.7705E+02	+	**3.8129E+02**	4.7239E+01	–	4.0716E+02	1.5669E+01	–
F26	3.0000E+02	5.6522E−08	+	1.3374E+03	3.1668E+02	+	**2.2046E+02**	4.7471E+01	–	3.0000E+02	3.0368E−11	+
F27	3.9557E+02	2.9103E+00	≈	4.3282E+02	1.0592E+01	+	3.9480E+02	5.5676E+00	≈	**3.9145E+02**	2.0970E+00	–
F28	**3.0000E+02**	1.2696E−08	+	8.9230E+02	1.7949E+02	+	3.2666E+02	1.9394E+01	+	3.2223E+02	6.5358E+01	+
F29	2.6964E+02	2.2067E+01	≈	4.6404E+02	8.0907E+01	+	2.9019E+02	1.5425E+01	+	**2.6856E+02**	8.2235E+00	≈
F30	2.8853E+05	8.4205E+05	+	3.9376E+06	7.7581E+06	+	2.1289E+04	2.9084E+04	+	**1.5277E+04**	9.5982E+03	+
W/T/L	22/5/2			29/0/0			23/2/4			13/4/12		

Table 8. Experimental and statistical results of HGMRFO and seven competitive algorithms on IEEE CEC2017 benchmark functions with 10 dimensions, where HGMRFO is the main algorithm in statistical results (W/T/L).

where N_{p} is the number of solar cells connected in parallel, and N_{s} is the number of solar cells connected in series. In the MM, five parameters need to be estimated: I_{ph}, I_{sd}, R_s, R_{sh}, and a. The objective function for parameter estimation is achieved by minimizing the error between the measured current data and the experimental current data[2].

To comprehensively evaluate the performance of the proposed HGMRFO, six groups of experiments were designed to encompass different numbers of series diodes (Ns) and parallel diodes (Np). For the SDM, DDM, and TDM, experimental data were obtained from a commercial silicon solar cell produced by R.T.C. France, with 26 sets of current-voltage data measured at a temperature of 33°C[78]. For the photovoltaic MM, the experimental data were sourced from Photowatt-PWP201, STP6-120/36, and STM6-40/36 PV modules, under varying temperature and irradiance conditions[78,79]. Table 12 presents a comprehensive overview of the true value sources, measurement conditions, and the number of experimental data points for each model. Additionally, Table 13 presents the lower bounds (LB) and upper bounds (UB) for parameter extraction in each model[80–82]. In these six experimental setups, the HGMRFO was systematically compared against seven other advanced optimization algorithms. These algorithms include the original MRFO[13], FCMRFO[27], HMRFO[12], water flow optimizer (WFO)[83], fractional-order WFO (FOWFO)[84], wind driven optimization (WDO)[85], and grey wolf optimizer (GWO)[86]. The configuration of parameters for all algorithms is summarized in Table 4. The experimental data and statistical results are presented in Table 14.

As presented in Table 14, HGMRFO achieved the best mean values across all photovoltaic (PV) models. Specifically, HGMRFO outperformed MRFO, FCMRFO, HMRFO, FOWFO, WFO, WDO, and GWO in 6, 6, 5, 6, 6, 6, and 6 PV models, respectively. According to the Friedman test ranks, HGMRFO is ranked first, indicating that HGMRFO outperforms other competing algorithms. These results clearly demonstrate the superior performance of HGMRFO in addressing the parameter estimation challenges associated with solar PV models, underscoring its efficacy and robustness in tackling such complex optimization problems.

Figure 6 presents the box-and-whisker diagrams of errors for HGMRFO and seven other algorithms across six photovoltaic models. The median, indicated by the red horizontal line, represents the average performance of the algorithms. From the figure, it can be seen that HGMRFO has the lowest median on the SDM, DDM, TDM, STP6-120/36, and STM6-40/36 models, indicating that HGMRFO performs best on these five models. However, on the Photowatt-PWP201 model, HGMRFO slightly underperforms compared to HMRFO, although its overall performance remains strong. The box-and-whisker diagrams demonstrate that HGMRFO exhibits excellent robustness and accuracy on PV models.

Conclusions

In this study, we address the limitations of the MRFO during the somersault foraging phase, specifically the fixed parameter setting and the exclusive focus on the current best individual during the somersault search. To overcome these limitations, we introduce an adaptive somersault factor and a hierarchical guidance mechanism, enhancing the information exchange among population individuals. Through the synergistic effect of a multi-layered structure, individuals are effectively guided towards the global optimal solution, thereby proposing an enhanced algorithm named hierarchical guided manta ray foraging optimization (HGMRFO). During the experimental phase, we initially identified the optimal parameter configurations for HGMRFO and conducted a systematic analysis of how different hierarchical structures influence the algorithm's performance. Extensive experiments on the IEEE CEC2017 benchmark functions and IEEE CEC2011 real-world optimization problems confirmed the significant advantages of HGMRFO in terms of optimization performance and practical applicability, demonstrating its lower computational cost and superior efficiency. Finally, HGMRFO was utilized for parameter estimation in solar photovoltaic models with multimodal characteristics, yielding excellent experimental outcomes and further demonstrating the potential and efficacy of this algorithm in addressing complex real-world problems.

In future research, we intend to focus on the following five areas:

	HGMRFO			MRFO			FCMRFO			CFDBWDO		
	Mean	Std		Mean	Std		Mean	Std		Mean	Std	
F1	2.8733E+03	3.3983E+03		3.7384E+03	4.4790E+03	≈	3.0502E+03	3.6798E+03	≈	1.7146E+09	1.1246E+09	+
F3	5.4023E+03	1.6083E+03		**4.7739E+00**	1.0342E+01	–	2.5560E+01	3.2147E+01	–	1.2670E+04	8.9234E+03	+
F4	**3.3121E+01**	3.4733E+01		4.6676E+01	3.7281E+01	≈	5.5588E+01	3.9557E+01	+	8.1562E+02	4.0561E+02	+
F5	**6.7175E+01**	2.1944E+01		1.5350E+02	4.3442E+01	+	1.3270E+02	4.0950E+01	+	1.6470E+02	4.3923E+01	+
F6	2.4752E−01	3.5103E−01		1.7815E+01	1.0132E+01	+	1.0386E+01	8.9397E+00	+	4.4069E+01	6.5509E+00	+
F7	**9.4761E+01**	1.9898E+01		2.5624E+02	6.9890E+01	+	1.6886E+02	4.4777E+01	+	2.7902E+02	6.0857E+01	+
F8	**6.4893E+01**	1.9153E+01		1.2886E+02	3.5590E+01	+	1.1641E+02	2.7480E+01	+	1.3355E+02	4.1496E+01	+
F9	5.3708E+01	6.3118E+01		2.2953E+03	1.0480E+03	+	1.0669E+03	9.4409E+02	+	2.9867E+03	1.2348E+03	+
F10	6.8983E+03	8.9017E+02		3.6595E+03	6.4142E+02	–	**3.4772E+03**	5.8910E+02	–	5.8775E+03	1.3076E+03	–
F11	**6.1740E+01**	3.2343E+01		8.6673E+01	3.6544E+01	+	7.6515E+01	3.6304E+01	+	3.7778E+02	6.1640E+01	+
F12	**4.5702E+04**	3.0292E+04		5.9688E+04	2.7587E+04	+	5.0525E+04	2.8510E+04	≈	2.8812E+08	2.2742E+08	+
F13	**1.1895E+04**	1.2034E+04		1.2701E+04	1.5277E+04	≈	1.3791E+04	1.3027E+04	≈	3.6731E+07	4.3282E+07	+
F14	**2.7243E+03**	2.1197E+03		2.7805E+03	3.2620E+03	≈	2.7566E+03	5.6662E+03	–	7.3610E+04	1.6813E+05	+
F15	3.2196E+03	4.0649E+03		7.1380E+03	9.1883E+03	+	5.7870E+03	6.0623E+03	+	1.2859E+05	2.2319E+05	+
F16	**5.6293E+02**	2.2147E+02		9.0200E+02	2.4113E+02	+	8.9713E+02	2.6164E+02	+	1.7389E+03	3.3959E+02	+
F17	1.6326E+02	1.1804E+02		4.6555E+02	2.1949E+02	+	3.0781E+02	1.8318E+02	+	4.3076E+02	1.7774E+02	+
F18	1.8222E+05	1.3458E+05		**8.5543E+04**	3.7115E+04	–	1.2147E+05	1.3642E+05	–	5.6785E+05	6.0264E+05	+
F19	4.8620E+03	5.5320E+03		8.2127E+03	9.9957E+03	≈	7.8671E+03	9.2365E+03	≈	7.7929E+05	7.7910E+05	+
F20	1.9678E+02	7.0967E+01		4.4438E+02	1.7426E+02	+	3.6037E+02	1.6785E+02	+	4.4312E+02	1.4110E+02	+
F21	**2.5805E+02**	1.6197E+01		3.1198E+02	3.4358E+01	+	2.9524E+02	2.8782E+01	+	4.0029E+02	4.8580E+01	+
F22	**1.0022E+02**	7.6361E−01		1.0042E+02	1.2042E+00	+	1.0028E+02	8.7263E−01	≈	5.6896E+02	8.8201E+01	+
F23	**4.1913E+02**	2.2177E+01		4.8741E+02	3.6661E+01	+	4.7078E+02	4.1095E+01	+	8.8305E+02	4.9124E+01	+
F24	**4.8605E+02**	2.8690E+01		5.7733E+02	5.3777E+01	+	5.4401E+02	4.1898E+01	+	1.0001E+03	5.9431E+01	+
F25	3.8730E+02	5.8341E+00		3.9438E+02	1.8613E+01	+	3.9153E+02	1.1722E+01	+	5.2113E+02	4.6526E+01	+
F26	1.2660E+03	9.7677E+02		1.9367E+03	1.4642E+03	+	1.8636E+03	1.0345E+03	+	3.3320E+03	1.9105E+03	+
F27	5.3630E+02	1.6493E+01		5.7030E+02	2.7188E+01	+	5.5385E+02	2.4958E+01	+	1.0904E+03	9.4137E+01	+
F28	**3.1422E+02**	3.5997E+01		3.3967E+02	5.7760E+01	+	3.2923E+02	5.1411E+01	+	7.2888E+02	7.3843E+01	+
F29	6.5629E+02	1.4963E+02		9.1283E+02	2.0666E+02	+	8.1832E+02	1.8844E+02	+	1.6473E+03	3.8267E+02	+
F30	**4.3869E+03**	1.5409E+03		5.9003E+03	2.5925E+03	+	4.6928E+03	2.2838E+03	≈	1.0823E+07	1.6080E+07	+
W/T/L	–/–/–			21/5/3			19/6/4			28/0/1		

	SIS			RSA			FOFPA			SE		
	Mean	Std		Mean	Std		Mean	Std		Mean	Std	
F1	5.1460E+03	5.6314E+03	+	5.3838E+10	9.2915E+09	+	2.4255E+07	1.0559E+07	+	**2.5172E+03**	2.0169E+03	≈
F3	7.1410E+02	5.0997E+03	–	7.4231E+04	5.5046E+03	+	2.2184E+04	4.4090E+03	+	5.4633E+04	7.6445E+03	+
F4	8.9081E+01	2.7812E+01	+	1.4546E+04	4.5609E+03	+	8.6346E+01	1.2506E+01	+	1.0510E+02	7.0488E+00	+
F5	9.9652E+01	3.0169E+01	+	3.8913E+02	3.3091E+01	+	1.8150E+02	2.0876E+01	+	7.5201E+01	7.6463E+00	+
F6	2.9762E+00	3.9293E+00	+	8.6338E+01	7.4613E+00	+	5.2801E+01	4.9568E+00	+	**1.8279E−13**	5.6058E−14	–
F7	1.4625E+02	2.6679E+01	+	6.7249E+02	6.7304E+01	+	3.0828E+02	3.5163E+01	+	1.1192E+02	8.5281E+00	+
F8	1.0076E+02	2.3524E+01	+	3.1165E+02	2.2062E+01	+	1.4680E+02	1.5907E+01	+	7.7764E+01	7.0849E+00	+
F9	2.1975E+01	3.6454E+01	–	8.5330E+03	1.1963E+03	+	4.1369E+03	9.2322E+02	+	**2.4584E+00**	1.9576E+00	–
F10	3.5043E+03	5.7230E+02	–	7.0211E+03	3.5932E+02	≈	3.7406E+03	2.0997E+02	–	3.7243E+03	2.4641E+02	–
F11	1.4958E+02	5.2693E+01	+	7.7699E+03	2.8062E+03	+	1.5798E+02	1.9698E+01	+	1.1566E+02	2.0547E+01	+
F12	6.5648E+05	5.5450E+05	+	1.7025E+10	4.3633E+09	+	5.3672E+05	2.5644E+05	+	2.0775E+06	6.4319E+05	+
F13	1.0107E+05	6.4281E+04	+	1.1868E+10	4.9062E+09	+	1.2354E+04	7.5592E+03	+	4.3738E+04	1.6766E+04	+
F14	3.1476E+03	4.3240E+03	≈	3.0735E+06	3.5882E+06	+	1.0125E+04	5.4452E+03	+	5.7333E+04	3.3255E+04	+
F15	5.8721E+04	3.4562E+04	+	6.7358E+08	5.7473E+08	+	**2.2305E+03**	1.5967E+03	≈	1.6491E+04	1.0272E+04	+
F16	7.1844E+02	2.4772E+02	+	3.8976E+03	6.8620E+02	+	1.0419E+03	1.0626E+02	+	5.7780E+02	1.1249E+02	≈
F17	2.7824E+02	1.6578E+02	+	5.3062E+03	6.8661E+03	+	3.1946E+02	7.8078E+01	+	**1.1896E+02**	2.8329E+01	≈
F18	1.4749E+05	9.2222E+04	≈	3.2772E+07	3.0709E+07	+	1.4638E+05	4.6420E+04	≈	2.5257E+05	1.2494E+05	+
F19	2.6815E+05	7.8325E+05	+	2.3230E+09	1.6940E+09	+	**3.7171E+03**	4.9079E+03	≈	2.2225E+04	1.0628E+04	+
F20	4.4304E+02	1.4685E+02	+	8.6362E+02	1.4256E+02	+	4.4067E+02	6.6277E+01	+	**1.6039E+02**	6.3034E+01	–
F21	2.9762E+02	2.3166E+01	+	6.4314E+02	4.2694E+01	+	3.1502E+02	8.6750E+01	+	2.7736E+02	1.4284E+01	+
F22	1.0034E+02	8.6008E−01	+	5.2531E+03	1.0078E+03	+	3.6966E+02	6.4764E+02	+	1.3624E+02	1.5243E+01	+

Continued

	SIS			RSA			FOFPA			SE		
	Mean	Std		Mean	Std		Mean	Std		Mean	Std	
F23	4.4659E+02	2.9306E+01	+	1.0393E+03	1.0894E+02	+	5.6695E+02	6.3230E+01	+	4.2387E+02	6.8250E+00	≈
F24	4.9935E+02	2.6257E+01	+	1.1775E+03	2.4527E+02	+	6.2125E+02	2.0691E+02	+	5.1274E+02	1.3466E+01	+
F25	**3.8651E+02**	1.0272E+00	−	2.2241E+03	8.6049E+02	+	4.0277E+02	8.7350E+00	+	3.8725E+02	1.6211E−01	≈
F26	1.7733E+03	4.4535E+02	+	7.9333E+03	1.1240E+03	+	**5.0349E+02**	6.8285E+01	≈	1.3656E+03	2.3990E+02	≈
F27	5.1974E+02	1.3804E+01	−	9.4089E+02	2.3127E+02	+	5.5949E+02	1.1191E+01	+	**5.1853E+02**	3.7328E+00	−
F28	3.5401E+02	6.0951E+01	≈	3.9854E+03	8.8503E+02	+	4.1538E+02	1.2010E+01	+	4.3049E+02	6.9779E+00	+
F29	8.3981E+02	1.8964E+02	+	4.1455E+03	1.6091E+03	+	9.5949E+02	8.5626E+01	+	**5.9090E+02**	4.8039E+01	−
F30	4.7924E+05	3.5917E+05	+	2.2386E+09	9.2590E+08	+	3.3446E+04	1.2479E+04	+	2.7492E+04	1.1025E+04	+
W/T/L	21/3/5			28/1/0			24/4/1			17/6/6		

Table 9. Experimental and statistical results of HGMRFO and seven competitive algorithms on IEEE CEC2017 benchmark functions with 30 dimensions, where HGMRFO is the main algorithm in statistical results (W/T/L).

1. Further enhancement for low-dimensional optimization problems: Based on the optimization results on the CEC2017 benchmark functions with 10 dimensions, particularly in comparison with SE, it is evident that HGMRFO requires further refinement for better performance in low-dimensional optimization tasks.
2. Improvement in selection methods for HGMRFO: In the boxplot analysis of the PV models, HGMRFO performed weaker than HMRFO on the Photowatt-PWP201 model. In the future, we could draw inspiration from the fitness-distance balance selection strategy in HMRFO[12] to improve the selection method in HGMRFO.
3. Extension of hierarchical guidance mechanism: We will explore the integration of the hierarchical guidance mechanism into other optimization algorithms to further enhance their performance and robustness.
4. Investigation of alternative population structures: We aim to study various alternative population structures in an effort to identify more sophisticated configurations that could improve the structural characteristics of the original algorithm, thereby contributing to overall optimization efficiency.
5. Application of HGMRFO to additional real-world problems: We plan to apply HGMRFO to other complex real-world problems, such as the protein structure prediction[87,88] and dendritic neural model[89–91], to validate its versatility and effectiveness in diverse domains.

	HGMRFO			MRFO			FCMRFO			CFDBWDO		
	Mean	Std		Mean	Std		Mean	Std		Mean	Std	
F1	2.8983E+03	3.4531E+03		5.2780E+03	5.2915E+03	+	**2.4110E+03**	3.1230E+03	≈	6.3178E+09	3.7639E+09	+
F3	5.9769E+04	8.9597E+03		1.6242E+04	4.6228E+03	−	2.8114E+04	6.3107E+03	−	4.3762E+04	1.5275E+04	−
F4	8.8922E+01	5.1504E+01		8.9105E+01	5.2308E+01	≈	**7.9753E+01**	4.8823E+01	≈	2.4301E+03	4.2825E+02	+
F5	**1.5722E+02**	3.6049E+01		3.2504E+02	3.8372E+01	+	2.9920E+02	4.4810E+01	+	2.8811E+02	− 3.3467E+01	+
F6	5.3091E+00	5.3419E+00		3.9001E+01	1.2274E+01	+	3.1434E+01	1.0894E+01	+	5.4654E+01	− 9.3791E+01	+
F7	**2.2124E+02**	4.8635E+01		6.3666E+02	1.5264E+02	+	4.5513E+02	1.2490E+02	+	6.9210E+02	5.4069E+01	+
F8	**1.6675E+02**	5.0528E+01		3.2629E+02	5.3327E+01	+	2.8608E+02	4.9351E+01	+	3.1024E+02	− 4.2001E+01	+
F9	9.8935E+02	1.0049E+03		8.2821E+03	2.5680E+03	+	5.6003E+03	3.4264E+03	+	1.2625E+04	3.8959E+03	+
F10	1.2438E+04	1.7360E+03		6.5747E+03	8.3820E+02	−	6.4619E+03	7.2637E+02	−	1.0006E+04	2.6815E+03	−
F11	**1.0491E+02**	3.1906E+01		1.5685E+02	3.9700E+01	+	1.4931E+02	3.8247E+01	+	1.4727E+03	3.4275E+02	+
F12	7.2181E+05	4.4740E+05		7.6245E+05	4.8366E+05	≈	**5.6938E+05**	3.6498E+05	−	3.1702E+09	1.4171E+09	+
F13	**2.4529E+03**	3.3796E+03		4.2083E+03	4.9110E+03	≈	4.0775E+03	4.7260E+03	≈	3.0138E+08	4.8371E+08	+
F14	3.6017E+04	2.4702E+04		2.0915E+04	2.0423E+04	−	2.6240E+04	2.0410E+04	−	1.6989E+06	1.9876E+06	+
F15	**4.6738E+03**	4.4349E+03		7.1083E+03	6.5137E+03	+	6.7152E+03	4.8648E+03	+	2.2851E+07	2.3651E+07	+
F16	1.2445E+03	4.4677E+02		1.6473E+03	4.6933E+02	+	1.6030E+03	4.5434E+02	+	2.8639E+03	5.7565E+02	+
F17	9.3279E+02	2.4280E+02		1.3505E+03	3.4141E+02	+	1.2046E+03	2.9992E+02	+	1.3843E+03	2.3460E+02	+
F18	5.7797E+05	3.8127E+05		**1.4713E+05**	8.6593E+04	−	2.4855E+05	1.9159E+05	−	3.2529E+06	1.8288E+06	+
F19	1.4017E+04	7.4756E+03		1.4547E+04	1.0327E+04	≈	**1.2729E+04**	9.3426E+03	≈	5.1108E+06	8.0553E+06	+
F20	8.0934E+02	4.6369E+02		1.0993E+03	3.5821E+02	+	9.9825E+02	3.1125E+02	+	1.1846E+03	3.0114E+02	+
F21	**3.3041E+02**	3.4227E+01		4.4890E+02	5.1858E+01	+	4.1641E+02	4.1589E+01	+	6.2562E+02	1.3609E+01	+
F22	5.7970E+03	6.4795E+03		6.9282E+03	1.6827E+03	≈	6.6503E+03	1.8022E+03	≈	9.8474E+03	3.9436E+03	+
F23	**5.9544E+02**	4.6470E+01		7.9393E+02	9.8260E+01	+	7.2727E+02	9.0568E+01	+	1.6205E+03	− 7.7462E+00	+
F24	**6.5170E+02**	4.8957E+01		8.6298E+02	9.6044E+01	+	7.8525E+02	1.0068E+02	+	1.8190E+03	1.0918E+01	+
F25	5.5282E+02	3.6089E+01		5.5539E+02	3.6850E+01	≈	5.5662E+02	3.9702E+01	≈	1.7569E+03	5.2701E+01	+
F26	1.5167E+03	1.9077E+03		3.8756E+03	3.7991E+03	+	2.0325E+03	2.9152E+03	≈	5.7099E+03	2.9044E+03	+
F27	7.2364E+02	7.4540E+01		9.0307E+02	1.5341E+02	+	8.3117E+02	1.1097E+02	+	2.6880E+03	1.4534E+02	+
F28	5.0137E+02	2.9873E+01		4.9911E+02	3.2913E+01	≈	4.9771E+02	1.6912E+01	≈	2.0690E+03	6.6164E+01	+
F29	1.0462E+03	2.8073E+02		1.5544E+03	3.0702E+02	+	1.3335E+03	4.4006E+02	+	3.6665E+03	4.3589E+02	+
F30	**8.8391E+05**	1.0071E+05		9.5632E+05	2.1236E+05	≈	1.0002E+06	2.7389E+05	≈	1.9511E+08	7.0613E+07	+
W/T/L	−/−/−			17/8/4			15/9/5			27/0/2		

	SIS			RSA			FOFPA			SE		
	Mean	Std		Mean	Std		Mean	Std		Mean	Std	
F1	1.0093E+04	9.2626E+03	+	9.6354E+10	9.8835E+09	+	1.9139E+07	7.7080E+06	+	1.9495E+04	2.3701E+04	+
F3	**1.9814E+03**	1.3306E+04	−	1.4873E+05	1.0424E+04	+	1.5878E+05	1.2952E+04	+	1.5025E+05	1.6633E+04	+
F4	1.5304E+02	5.6066E+01	+	2.7103E+04	6.7603E+03	+	1.1109E+02	1.9581E+01	+	1.5174E+02	1.6103E+01	+
F5	2.0986E+02	4.1882E+01	+	6.3363E+02	2.8984E+01	+	3.3962E+02	2.3860E+01	+	1.7057E+02	1.0971E+01	+
F6	1.6084E+01	7.2260E+00	+	9.8267E+01	4.6805E+00	+	6.4303E+01	4.6559E+00	+	**3.3660E−13**	2.2287E−14	−
F7	2.8881E+02	4.5968E+01	+	1.3023E+03	5.9024E+01	+	6.8090E+02	5.2656E+01	+	2.3168E+02	1.3679E+01	+
F8	2.0811E+02	4.3225E+01	+	6.7911E+02	2.6501E+01	+	3.4995E+02	2.5616E+01	+	1.6730E+02	1.2430E+01	≈
F9	1.0598E+03	1.0417E+03	≈	3.2043E+04	2.5802E+03	+	1.5323E+04	3.0686E+03	+	**2.6562E+02**	1.0576E+02	−
F10	**5.7782E+03**	8.7675E+02	−	1.3150E+04	4.8002E+02	+	6.9699E+03	2.8953E+02	−	6.8537E+03	3.3037E+02	−
F11	2.3126E+02	5.2205E+01	+	1.6818E+04	2.8463E+03	+	2.9962E+02	4.2684E+01	+	3.7420E+02	1.3437E+02	+
F12	4.7872E+06	2.5756E+06	+	7.6674E+10	1.7083E+10	+	2.8762E+06	8.6669E+05	+	1.2706E+07	2.7079E+06	+
F13	1.0792E+05	5.7404E+04	+	4.5896E+10	1.2704E+10	+	1.4700E+04	1.6003E+04	+	8.1687E+04	4.7146E+04	+
F14	**1.4347E+04**	1.2863E+04	−	3.6046E+07	2.8641E+07	+	6.4848E+04	2.4776E+04	+	5.7709E+05	2.5290E+05	+
F15	4.9485E+04	2.6529E+04	+	6.6284E+09	4.8002E+09	+	6.3323E+03	5.2919E+03	+	1.6543E+04	7.1668E+03	+
F16	1.2498E+03	4.7228E+02	≈	7.1723E+03	1.3780E+03	+	1.7038E+03	1.5531E+02	+	**1.1560E+03**	1.8081E+02	≈
F17	9.9582E+02	3.4084E+02	≈	1.8194E+04	3.4088E+04	+	1.2753E+03	1.5210E+02	+	**7.7779E+02**	1.4202E+02	−
F18	1.8011E+05	8.8506E+04	−	9.3450E+07	5.2370E+07	+	5.2321E+05	2.1611E+05	≈	1.4491E+06	6.3307E+05	+
F19	1.7238E+05	6.2041E+04	+	6.8885E+09	2.8016E+09	+	1.8972E+04	1.0705E+04	+	2.2708E+04	6.1905E+03	+
F20	9.1993E+02	3.0532E+02	+	1.8251E+03	1.9615E+02	+	1.1692E+03	1.2745E+02	+	**6.0702E+02**	1.4265E+02	≈
F21	3.9314E+02	3.8134E+01	+	1.0318E+03	9.7335E+01	+	5.8779E+02	3.7828E+01	+	3.7776E+02	9.5461E+00	+
F22	**2.9429E+03**	3.6154E+03	−	1.4076E+04	5.1507E+02	+	7.7114E+03	2.2566E+02	≈	5.1137E+03	3.2910E+03	≈

Continued

	SIS			RSA			FOFPA			SE		
	Mean	Std		Mean	Std		Mean	Std		Mean	Std	
F23	6.4121E+02	4.8093E+01	+	1.6539E+03	1.5821E+02	+	9.6865E+02	6.2814E+01	+	6.0154E+02	1.2685E+01	≈
F24	6.7374E+02	4.2945E+01	+	2.0220E+03	4.1296E+02	+	1.2737E+03	8.0214E+01	+	7.2947E+02	2.0034E+01	+
F25	**5.1487E+02**	3.8884E+01	–	1.0315E+04	1.2693E+03	+	5.7695E+02	2.0371E+01	+	5.5722E+02	7.9234E+00	≈
F26	3.2254E+03	5.1078E+02	+	1.3410E+04	1.0356E+03	+	**5.9122E+02**	1.1284E+02	+	2.6982E+03	2.1562E+02	+
F27	6.3518E+02	5.7235E+01	–	1.9051E+03	3.2647E+02	+	8.8472E+02	5.1563E+01	+	**6.1808E+02**	1.3974E+01	–
F28	**4.7654E+02**	2.2494E+01	–	8.9502E+03	1.1330E+03	+	5.4836E+02	2.4664E+01	+	5.1360E+02	1.1637E+01	+
F29	1.2794E+03	2.7048E+02	+	6.1887E+04	4.4509E+04	+	1.6492E+03	1.3134E+02	+	**6.9952E+02**	8.7430E+01	–
F30	1.9538E+07	5.0749E+06	+	8.6047E+09	2.7177E+09	+	1.7037E+06	3.0742E+05	+	9.8329E+05	8.9598E+04	+
W/T/L	18/3/8			29/0/0			26/2/1			17/6/6		

Table 10. Experimental and statistical results of HGMRFO and seven competitive algorithms on IEEE CEC2017 benchmark functions with 50 dimensions, where HGMRFO is the main algorithm in statistical results (W/T/L).

	HGMRFO			MRFO			FCMRFO			CFDBWDO		
	Mean	Std		Mean	Std		Mean	Std		Mean	Std	
F1	**4.3950E+03**	5.1323E+03		4.7446E+03	7.0201E+03	≈	8.1291E+03	1.0361E+04	+	2.9504E+10	1.0848E+10	+
F3	2.2707E+05	1.9755E+04		8.8374E+04	1.4356E+04	–	1.1692E+05	1.5175E+04	–	1.6744E+05	4.4682E+04	–
F4	**1.9908E+02**	4.5853E+01		2.4411E+02	4.5461E+01	+	2.2328E+02	3.9187E+01	+	1.2160E+04	1.4642E+03	+
F5	**4.2818E+02**	8.7618E+01		7.9387E+02	5.3833E+01	+	7.6260E+02	8.3190E+01	+	7.6707E+02	4.8749E+01	+
F6	1.7818E+01	6.9169E+00		5.2876E+01	6.5346E+00	+	4.9528E+01	7.2767E+00	+	6.1798E+01	− 9.4708E+01	+
F7	**6.0963E+02**	1.5289E+02		2.0444E+03	3.7554E+02	+	1.6132E+03	3.4024E+02	+	1.9436E+03	2.0617E+02	+
F8	**4.1954E+02**	6.7210E+01		8.7951E+02	9.6885E+01	+	7.9650E+02	1.2433E+02	+	8.6846E+02	6.7901E+01	+
F9	1.1120E+04	7.3753E+03		2.0954E+04	1.5224E+03	+	1.9962E+04	2.5695E+03	+	3.1248E+04	4.9407E+03	+
F10	2.9138E+04	2.1329E+03		1.4000E+04	1.1428E+03	–	1.3863E+04	1.5173E+03	–	2.1764E+04	6.7448E+03	–
F11	5.8925E+02	1.1622E+02		**5.8149E+02**	9.9737E+01	≈	5.8807E+02	1.0790E+02	≈	3.0723E+04	1.0581E+04	+
F12	**9.3751E+05**	3.7590E+05		1.1140E+06	4.9563E+05	+	9.8180E+05	4.2781E+05	≈	1.8982E+10	4.6177E+09	+
F13	4.6517E+03	4.2497E+03		5.1765E+03	5.4638E+03	≈	**3.4921E+03**	4.0192E+03	≈	1.1341E+09	8.0140E+08	+
F14	3.3422E+05	1.5901E+05		1.2103E+05	5.9541E+04	–	1.6130E+05	8.8965E+04	–	1.1653E+06	1.1701E+06	+
F15	**1.7504E+03**	2.2805E+03		2.6142E+03	2.9806E+03	≈	2.8844E+03	3.8767E+03	+	5.3062E+07	8.7422E+07	+
F16	3.0525E+03	6.1009E+02		4.0348E+03	6.4469E+02	+	3.9949E+03	6.8760E+02	+	8.5857E+03	1.3469E+03	+
F17	2.4440E+03	5.5677E+02		3.0686E+03	5.9598E+02	+	2.9326E+03	5.7286E+02	+	5.2203E+03	2.3099E+03	+
F18	1.0250E+06	4.1142E+05		3.1414E+05	1.3299E+05	–	4.2797E+05	1.8470E+05	–	1.7729E+06	7.9590E+05	+
F19	1.9931E+03	2.0981E+03		4.4468E+03	5.8955E+03	≈	2.4934E+03	2.8485E+03	≈	2.1815E+08	1.5182E+08	+
F20	4.0276E+03	1.0135E+03		3.0617E+03	5.7650E+02	–	2.9411E+03	5.4850E+02	–	3.8093E+03	7.1281E+02	–
F21	**5.8208E+02**	6.3854E+01		9.6218E+02	1.0817E+02	+	8.1776E+02	9.9167E+01	+	1.8832E+03	1.0356E+02	+
F22	2.4909E+04	1.1832E+04		1.6716E+04	1.7654E+03	–	1.6349E+04	1.8077E+03	–	2.5555E+04	5.6587E+03	≈
F23	9.8449E+02	8.2821E+01		1.3800E+03	1.2843E+02	+	1.2053E+03	1.0267E+02	+	3.6213E+03	5.5015E+01	+
F24	1.3794E+03	1.4244E+02		2.1546E+03	2.5059E+02	+	1.8367E+03	1.8433E+02	+	6.5874E+03	3.5037E+02	+
F25	**7.7228E+02**	6.7245E+01		7.7320E+02	6.8997E+01	≈	7.8338E+02	6.9839E+01	≈	5.1983E+03	3.5974E+02	+
F26	8.5370E+03	5.0055E+03		1.6200E+04	7.7381E+03	+	1.3303E+04	7.2289E+03	+	2.0903E+04	4.2852E+03	+
F27	8.9186E+02	7.0583E+01		1.2403E+03	2.2460E+02	+	1.0989E+03	1.6959E+02	+	6.0325E+03	7.2409E+02	+
F28	**5.6072E+02**	3.6083E+01		5.7281E+02	3.1286E+01	+	5.7067E+02	3.2488E+01	+	9.9749E+03	8.6770E+02	+
F29	3.2878E+03	5.1092E+02		3.7893E+03	5.1347E+02	+	3.6916E+03	6.7887E+02	+	1.0266E+04	1.7646E+03	+
F30	**5.6875E+03**	2.3273E+03		7.8957E+03	4.1346E+03	+	6.3306E+03	2.9860E+03	≈	2.3276E+09	8.1724E+08	+
W/T/L	–/–/–			17/6/6			17/6/6			25/1/3		

	SIS			RSA			FOFPA			SE		
	Mean	Std		Mean	Std		Mean	Std		Mean	Std	
F1	1.4163E+04	1.6427E+04	+	2.4353E+11	9.4459E+09	+	1.5212E+06	5.3324E+05	+	2.2507E+04	1.5263E+04	+
F3	**1.3736E+04**	3.9250E+04	–	3.1547E+05	1.6834E+04	+	2.9592E+05	1.6622E+04	+	4.8042E+05	3.4697E+04	+
F4	2.6601E+02	3.4809E+01	+	7.5174E+04	9.7383E+03	+	2.5183E+02	3.0718E+01	+	2.6818E+02	2.2653E+01	+
F5	5.4974E+02	5.8386E+01	+	1.4827E+03	4.7431E+01	+	8.5012E+02	5.1810E+01	+	4.9956E+02	2.2169E+01	+
F6	4.0448E+01	7.7577E+00	+	1.0715E+02	3.5622E+00	+	7.0256E+01	4.2542E+00	+	**7.5122E−13**	5.6058E−14	–
F7	7.4762E+02	7.5307E+01	+	3.3607E+03	1.0665E+02	+	2.0193E+03	1.0791E+02	+	6.4528E+02	2.5673E+01	+
F8	5.5742E+02	8.4283E+01	+	1.6362E+03	4.0523E+01	+	9.2606E+02	6.2942E+01	+	4.9286E+02	2.6490E+01	+
F9	1.9330E+04	5.9329E+03	+	7.4593E+04	6.5991E+03	+	3.6867E+04	5.7495E+03	+	**1.0386E+04**	2.2831E+03	≈
F10	**1.2577E+04**	1.4565E+03	–	2.9763E+04	7.0758E+02	≈	1.7515E+04	6.3382E+02	–	1.7092E+04	4.9547E+02	–
F11	1.3179E+03	2.5099E+02	+	1.5772E+05	2.5005E+04	+	8.4576E+03	1.3054E+03	+	2.1558E+04	5.4449E+03	+
F12	9.2281E+06	4.1137E+06	+	1.6698E+11	2.3229E+10	+	1.3718E+07	2.5983E+06	+	5.3478E+07	1.0606E+07	+
F13	6.2175E+04	1.9198E+04	+	4.1173E+10	8.7188E+09	+	5.8909E+03	3.0086E+03	+	4.3257E+04	1.6541E+04	+
F14	**8.3174E+04**	3.7249E+04	–	6.2005E+07	2.3708E+07	+	7.3321E+05	1.7261E+05	+	6.3763E+06	1.6150E+06	+
F15	5.2094E+04	2.3334E+04	+	2.2956E+10	6.8841E+09	+	3.9567E+03	6.5635E+03	+	2.3022E+04	9.7701E+03	+
F16	**2.8568E+03**	5.9430E+02	≈	1.9723E+04	3.3278E+03	+	4.4461E+03	2.1387E+02	+	3.3913E+03	3.6300E+02	+
F17	**2.3544E+03**	4.4182E+02	≈	6.7270E+06	5.8727E+06	+	3.2879E+03	1.8120E+02	+	2.3892E+03	2.8523E+02	≈
F18	**2.1967E+05**	6.6718E+04	–	9.2932E+07	3.8790E+07	+	1.2549E+06	2.8228E+05	+	5.3515E+06	1.5958E+06	+
F19	3.6051E+05	1.0849E+05	+	2.3053E+10	7.1824E+09	+	**1.7766E+03**	1.6915E+03	≈	5.0503E+04	2.2719E+04	+
F20	2.6299E+03	4.3728E+02	–	5.1285E+03	2.1535E+02	+	3.3583E+03	1.8651E+02	–	**2.2639E+03**	2.3195E+02	–
F21	7.8552E+02	8.5750E+01	+	3.0143E+03	2.4895E+02	+	1.2683E+03	7.5088E+01	+	7.2794E+02	2.5761E+01	+
F22	**1.4044E+04**	3.1171E+03	–	3.1298E+04	5.6394E+02	+	1.9622E+04	4.5425E+02	–	1.7728E+04	2.4653E+03	–

Continued

	SIS			RSA			FOFPA			SE		
	Mean	Std		Mean	Std		Mean	Std		Mean	Std	
F23	1.1953E+03	1.0053E+02	+	2.9617E+03	1.5603E+02	+	1.5634E+03	6.0424E+01	+	**7.6215E+02**	1.5801E+01	–
F24	1.4682E+03	1.0248E+02	+	6.7656E+03	2.2875E+03	+	2.0977E+03	5.0636E+01	+	**1.3076E+03**	3.0359E+01	–
F25	7.9192E+02	7.3711E+01	≈	2.2055E+04	2.2289E+03	+	8.4930E+02	3.1929E+01	+	8.5147E+02	1.4579E+01	+
F26	8.8894E+03	8.4418E+02	≈	4.4698E+04	3.9093E+03	+	**7.6315E+02**	3.4708E+02	–	7.8411E+03	2.8319E+02	–
F27	8.0199E+02	6.8438E+01	–	6.0583E+03	1.7301E+03	+	1.0369E+03	5.5347E+01	+	**7.6293E+02**	1.6812E+01	–
F28	6.2088E+02	4.3364E+01	+	2.7273E+04	2.4450E+03	+	7.0635E+02	3.1676E+01	+	6.5017E+02	2.6015E+01	+
F29	3.4800E+03	5.0835E+02	+	5.5957E+05	4.3816E+05	+	4.5971E+03	2.5399E+02	+	**2.7248E+03**	2.0647E+02	–
F30	2.4452E+06	1.0612E+06	+	3.8886E+10	6.2514E+09	+	6.5680E+04	8.1794E+04	+	1.8381E+04	5.0119E+03	+
W/T/L	18/4/7			28/1/0			24/1/4			18/2/9		

Table 11. Experimental and statistical results of HGMRFO and seven competitive algorithms on IEEE CEC2017 benchmark functions with 100 dimensions, where HGMRFO is the main algorithm in statistical results (W/T/L).

Model	Source of true values	Series diodes (Ns)	Parallel diodes (Np)	Measurement conditions	Number of experimental data sets
SDM	R.T.C. France commercial silicon solar cell	1	1	33°C	26 sets
DDM	R.T.C. France commercial silicon solar cell	1	2	33°C	26 sets
TDM	R.T.C. France commercial silicon solar cell	1	3	33°C	26 sets
MM (Photowatt-PWP201)	Photowatt-PWP201 PV module	36	1	45°C, 100 W/mÂ²	25 sets
MM (STP6-120/36)	Schutten Solar STP6-120/36 PV module	36	1	55°C	24 sets
MM (STM6-40/36)	Schutten Solar STM6-40/36 PV module	36	1	51°C	20 sets

Table 12. Sources of experimental true current-voltage values, measurement conditions, and number of experimental data sets for six models.

	SDM or DDM or TDM		MM					
			Photowatt-PWP201		STP6-120/36		STM6-40/36	
Parameters	LB	UB	LB	UB	LB	UB	LB	UB
I_{ph}(A)	0	1	0	2	0	8	0	2
I_{sd}, I_{sd1}, I_{sd2}, I_{sd3}(μA)	0	1	0	50	0	50	0	50
a, *a*1, *a*2, *a*3	1	2	1	50	1	50	1	60
R_s(Ω)	0	0.5	0	2	0	0.36	0	0.36
R_{sh}(Ω)	0	100	0	2000	0	1500	0	1000

Table 13. Parameters ranges for PV models.

	HGMRFO			MRFO			FCMRFO			HMRFO		
	Mean	Std		Mean	Std		Mean	Std		Mean	Std	
SDM	**1.263E−02**	3.830E−03		1.934E−02	4.381E−03	+	2.390E−02	6.270E−03	+	2.160E−02	5.990E−03	+
DDM	**1.402E−02**	3.313E−03		1.826E−02	4.855E−03	+	2.340E−02	4.928E−03	+	2.025E−02	4.842E−03	+
TDM	**1.414E−02**	4.416E−03		1.836E−02	3.449E−03	+	2.114E−02	5.273E−03	+	2.221E−02	5.577E−03	+
Photowatt-PWP201	**9.136E−02**	2.485E−02		1.353E−01	6.923E−02	+	1.550E−01	5.778E−02	+	1.099E−01	4.619E−02	≈
STP6-120/36	**6.660E−01**	1.956E−01		1.084E+00	6.967E−02	+	1.075E+00	6.289E−02	+	9.859E−01	1.365E−01	+
STM6-40/36	**8.166E−02**	2.374E−02		1.409E−01	8.339E−02	+	1.387E−01	5.356E−02	+	1.106E−01	4.278E−02	+
W/T/L	−/−/−			6/0/0			6/0/0			5/1/0		
Rank	**1**			3.5			5.5			4.6667		
	FOWFO			WFO			WDO			GWO		
	Mean	Std		Mean	Std		Mean	Std		Mean	Std	
SDM	1.787E−02	6.530E−03	+	2.087E−02	8.145E−03	+	1.720E−02	4.984E−03	+	2.101E−02	7.975E−03	+
DDM	1.853E−02	3.999E−03	+	1.748E−02	6.324E−03	+	1.841E−02	4.358E−03	+	1.976E−02	6.648E−03	+
TDM	2.074E−02	7.382E−03	+	2.072E−02	9.361E−03	+	2.048E−02	7.471E−03	+	2.207E−02	1.005E−02	+
Photowatt-PWP201	2.129E−01	7.681E−02	+	2.272E−01	1.098E−01	+	2.442E−01	6.449E−02	+	2.495E−01	8.019E−02	+
STP6-120/36	1.104E+00	3.290E−01	+	1.069E+00	3.796E−01	+	1.126E+00	3.187E−01	+	1.191E+00	3.673E−01	+
STM6-40/36	1.479E−01	2.921E−02	+	1.490E−01	5.589E−02	+	1.678E−01	3.474E−02	+	1.774E−01	6.148E−02	+
W/T/L	6/0/0			6/0/0			6/0/0			6/0/0		
Rank	4.8333			4.3333			5			7.1667		

Table 14. Analysis and comparative performance of HGMRFO and other competitive algorithms on the PV models.

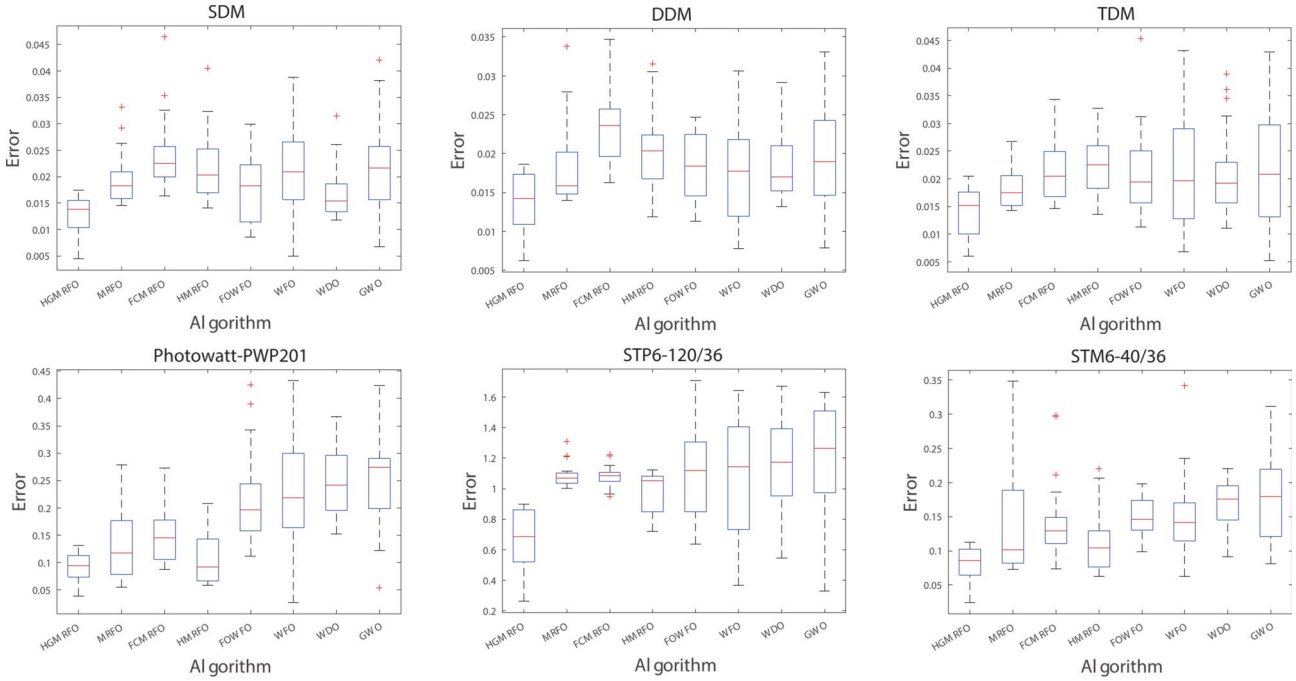


Fig. 6. Box-and-whisker diagrams of errors obtained by eight algorithms on six PV models.

Data availability

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

Received: 2 January 2025; Accepted: 17 February 2025
Published online: 21 July 2025

References

1. Zhang, Z. et al. Triple-layered chaotic differential evolution algorithm for layout optimization of offshore wave energy converters. *Expert Syst. Appl.* **239**, 122439 (2024).
2. Gao, S. et al. A state-of-the-art differential evolution algorithm for parameter estimation of solar photovoltaic models. *Energy Convers. Manage.* **230**, 113784 (2021).
3. Zhang, B. et al. Information gain-based multi-objective evolutionary algorithm for feature selection. *Inf. Sci.* **677**, 120901 (2024).
4. Kramer, O. *Genetic Algorithm Essentials* Vol. 679 (Springer, 2017).
5. Wang, D., Tan, D. & Liu, L. Particle swarm optimization algorithm: An overview. *Soft. Comput.* **22**, 387–408 (2018).
6. Dorigo, M. & Stützle, T. Ant colony optimization: overview and recent advances. In *Handbook of Metaheuristics*, 311–351 (Springer, 2019).
7. Rashedi, E., Nezamabadi-Pour, H. & Saryazdi, S. GSA: A gravitational search algorithm. *Inf. Sci.* **179**, 2232–2248 (2009).
8. Wang, X. Draco lizard optimizer: A novel metaheuristic algorithm for global optimization problems. *Evol. Intel.* **18**, 10 (2025).
9. Wang, X. Eurasian lynx optimizer: A novel metaheuristic optimization algorithm for global optimization and engineering applications. *Phys. Scr.* **99**, 115275 (2024).
10. Wang, X. Artificial meerkat algorithm: A new metaheuristic algorithm for solving optimization problems. *Phys. Scr.* **99**, 125280 (2024).
11. Wang, Y., Gao, S., Zhou, M. & Yu, Y. A multi-layered gravitational search algorithm for function optimization and real-world problems. *IEEE/CAA J. Autom. Sin.* **8**, 94–109 (2021).
12. Tang, Z. et al. Hierarchical manta ray foraging optimization with weighted fitness-distance balance selection. *Int. J. Comput. Intell. Syst.* **16**, 114 (2023).
13. Zhao, W., Zhang, Z. & Wang, L. Manta ray foraging optimization: An effective bio-inspired optimizer for engineering applications. *Eng. Appl. Artif. Intell.* **87**, 103300 (2020).
14. Tang, A., Zhou, H., Han, T. & Xie, L. A modified manta ray foraging optimization for global optimization problems. *IEEE Access* **9**, 128702–128721 (2021).
15. Xu, X. et al. A novel calibration method for robot kinematic parameters based on improved manta ray foraging optimization algorithm. *IEEE Trans. Instrum. Meas.* **72**, 1–11 (2023).
16. Rizk-Allah, R. M. et al. On a novel hybrid manta ray foraging optimizer and its application on parameters estimation of lithium-ion battery. *Int. J. Comput. Intell. Syst.* **15**, 62 (2022).
17. El-Shorbagy, M. A., Omar, H. A. & Fetouh, T. Hybridization of manta-ray foraging optimization algorithm with pseudo parameter-based genetic algorithm for dealing optimization problems and unit commitment problem. *Mathematics* **10**, 2179 (2022).
18. Ekinci, S., Izci, D. & Kayri, M. An effective controller design approach for magnetic levitation system using novel improved manta ray foraging optimization. *Arab. J. Sci. Eng.* **47**, 9673–9694 (2022).
19. Abdul Razak, A. A., Nasir, A. N. K., Abdul Ghani, N. M. & Tokhi, M. O. Opposition-based manta ray foraging algorithm for global optimization and its application to optimize nonlinear type-2 fuzzy logic control. *J. Low Freq. Noise Vib. Active Control* **43**, 1339–1362 (2024).
20. Ekinci, S., Izci, D. & Hekimoğlu, B. Optimal FOPID speed control of DC motor via opposition-based hybrid manta ray foraging optimization and simulated annealing algorithm. *Arab. J. Sci. Eng.* **46**, 1395–1409 (2021).
21. Zhang, K. et al. IBMRFO: Improved binary manta ray foraging optimization with chaotic tent map and adaptive somersault factor for feature selection. *Expert Syst. Appl.* **251**, 123977 (2024).
22. Qu, P., Yuan, Q., Du, F. & Gao, Q. An improved manta ray foraging optimization algorithm. *Sci. Rep.* **14**, 10301 (2024).
23. Got, A., Zouache, D. & Moussaoui, A. MOMRFO: Multi-objective manta ray foraging optimizer for handling engineering design problems. *Knowl.-Based Syst.* **237**, 107880 (2022).
24. Zhong, C., Li, G., Meng, Z., Li, H. & He, W. Multi-objective SHADE with manta ray foraging optimizer for structural design problems. *Appl. Soft Comput.* **134**, 110016 (2023).
25. Singh, S., Kumar, R. & Rao, U. P. Multi-objective adaptive manta-ray foraging optimization for workflow scheduling with selected virtual machines using time-series-based prediction. *Int. J. Softw. Sci. Comput. Intell.* **14**, 1–25 (2022).
26. Haddadian Nezhad, E., Ebrahimi, R. & Ghanbari, M. Fuzzy multi-objective allocation of photovoltaic energy resources in unbalanced network using improved manta ray foraging optimization algorithm. *Expert Syst. Appl.* **234**, 121048 (2023).
27. Yousri, D. et al. Discrete fractional-order Caputo method to overcome trapping in local optima: Manta ray foraging optimizer as a case study. *Expert Syst. Appl.* **192**, 116355 (2022).
28. Zhu, D., Wang, S., Zhou, C. & Yan, S. Manta ray foraging optimization based on mechanics game and progressive learning for multiple optimization problems. *Appl. Soft Comput.* **145**, 110561 (2023).
29. Got, A., Zouache, D., Moussaoui, A., Abualigah, L. & Alsayat, A. Improved manta ray foraging optimizer-based SVM for feature selection problems: A medical case study. *J. Bionic Eng.* **21**, 409–425 (2024).
30. Karrupusamy, P. Hybrid manta ray foraging optimization for novel brain tumor detection. *J. Soft Comput. Paradigm (JSCP)* **02**, 175–185 (2020).
31. Rawat, N. & Thakur, P. Comparative analysis of hybrid and metaheuristic parameter estimation methods of solar PV. *Mater. Today Proc.* **65**, 3748–3756 (2022).
32. Rawat, N., Thakur, P. & Singh, A. K. A novel hybrid parameter estimation technique of solar PV. *Int. J. Energy Res.* **46**, 4919–4934 (2022).
33. Rawat, N. et al. A new grey wolf optimization-based parameter estimation technique of solar photovoltaic. *Sustain. Energy Technol. Assess.* **57**, 103240 (2023).
34. Rawat, N., Thakur, P., Singh, A. K. & Bansal, R. Performance analysis of solar PV parameter estimation techniques. *Optik* **279**, 170785 (2023).
35. Sharma, P., Raju, S., Salgotra, R. & Gandomi, A. H. Parametric estimation of photovoltaic systems using a new multi-hybrid evolutionary algorithm. *Energy Rep.* **10**, 4447–4464 (2023).
36. Rathod, A. A. & Subramanian, B. Efficient approach for optimal parameter estimation of PV using Pelican Optimization Algorithm. *Cogent Eng.* **11**, 2380805 (2024).
37. Sundar Ganesh, C. S., Kumar, C., Premkumar, M. & Derebew, B. Enhancing photovoltaic parameter estimation: Integration of non-linear hunting and reinforcement learning strategies with golden jackal optimizer. *Sci. Rep.* **14**, 2756 (2024).
38. Beskirli, A., Dag, I. & Kiran, M. S. A tree seed algorithm with multi-strategy for parameter estimation of solar photovoltaic models. *Appl. Soft Comput.* **167**, 112220 (2024).
39. Rawat, N., Thakur, P., Dixit, A., Charu, K. & Goyal, S. Grey wolf optimisation based modified parameter estimation technique of solar PV. In *2024 IEEE Third International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES)*, 755–760 (IEEE, 2024).
40. Charu, K. M. et al. An efficient data sheet based parameter estimation technique of solar PV. *Sci. Rep.* **14**, 6461 (2024).
41. Dorronsoro, B. & Bouvry, P. Improving classical and decentralized differential evolution with new mutation operator and population topologies. *IEEE Trans. Evol. Comput.* **15**, 67–98 (2011).
42. Lynn, N., Ali, M. Z. & Suganthan, P. N. Population topologies for particle swarm optimization and differential evolution. *Swarm Evol. Comput.* **39**, 24–35 (2018).
43. Cantú-Paz, E. *Efficient and Accurate Parallel Genetic Algorithms* Vol. 1 (Kluwer Academic Publishers, 2000).
44. Alba, E. & Tomassini, M. Parallelism and evolutionary algorithms. *IEEE Trans. Evol. Comput.* **6**, 443–462 (2002).

45. Dorronsoro, B. & Alba, E. *Cellular Genetic Algorithms* Vol. 42 (Springer, 2008).
46. Dorronsoro, B. & Bouvry, P. Differential evolution algorithms with cellular populations. In *Parallel Problem Solving from Nature, PPSN XI*, 320–330 (Springer, 2010).
47. Nabizadeh, S., Rezvanian, A. & Meybodi, M. R. A multi-swarm cellular PSO based on clonal selection algorithm in dynamic environments. In *2012 International Conference on Informatics, Electronics & Vision (ICIEV)*, 482–486 (IEEE, 2012).
48. Liao, J., Cai, Y., Wang, T., Tian, H. & Chen, Y. Cellular direction information based differential evolution for numerical optimization: An empirical study. *Soft. Comput.* **20**, 2801–2827 (2016).
49. Liang, J. & Suganthan, P. Dynamic multi-swarm particle swarm optimizer. In *Proceedings 2005 IEEE Swarm Intelligence Symposium, 2005. SIS 2005.*, 124–129 (IEEE, 2005).
50. El Dor, A., Clerc, M. & Siarry, P. A multi-swarm PSO using charged particles in a partitioned search space for continuous optimization. *Comput. Optim. Appl.* **53**, 271–295 (2012).
51. Cheng, J., Zhang, G. & Neri, F. Enhancing distributed differential evolution with multicultural migration for global numerical optimization. *Inf. Sci.* **247**, 72–93 (2013).
52. Gong, Y.-J. et al. Distributed evolutionary algorithms and their models: A survey of the state-of-the-art. *Appl. Soft Comput.* **34**, 286–300 (2015).
53. Janson, S. & Middendorf, M. A hierarchical particle swarm optimizer and its adaptive variant. *IEEE Trans. Syst. Man Cybern. B* **35**, 1272–1282 (2005).
54. Wang, K., Lei, Z., Wang, Z., Zhang, Z. & Gao, S. Deep-layered differential evolution. In *Advances in Swarm Intelligence*, 503–515 (Springer, 2023).
55. Giacobini, M., Preuss, M. & Tomassini, M. Effects of scale-free and small-world topologies on binary coded self-adaptive CEA. In *Evolutionary Computation in Combinatorial Optimization*, 86–98 (Springer, 2006).
56. Zhang, C. & Yi, Z. Scale-free fully informed particle swarm optimization algorithm. *Inf. Sci.* **181**, 4550–4568 (2011).
57. Yu, Y. et al. Scale-free network-based differential evolution to solve function optimization and parameter estimation of photovoltaic models. *Swarm Evol. Comput.* **74**, 101142 (2022).
58. Lieberman, E., Hauert, C. & Nowak, M. A. Evolutionary dynamics on graphs. *Nature* **433**, 312–316 (2005).
59. Gao, S., Wang, Y., Wang, J. & Cheng, J. Understanding differential evolution: A poisson law derived from population interaction network. *J. Comput. Sci.* **21**, 140–149 (2017).
60. Wang, Y., Gao, S., Yu, Y. & Xu, Z. The discovery of population interaction with a power law distribution in brain storm optimization. *Mem. Comput.* **11**, 65–87 (2019).
61. Awad, N., Ali, M., Liang, J., Qu, B. & Suganthan, P. Problem definitions and evaluation criteria for the CEC 2017 special session and competition on single objective bound constrained real-parameter numerical optimization. In *Technical Report* (Nanyang Technological University Singapore, 2016).
62. Das, S. & Suganthan, P. N. Problem definitions and evaluation criteria for CEC 2011 competition on testing evolutionary algorithms on real world optimization problems. *Jadavpur University, Nanyang Technological University, Kolkata* 341–359 (2010).
63. Luengo, J., García, S. & Herrera, F. A study on the use of statistical tests for experimentation with neural networks: Analysis of parametric test conditions and non-parametric tests. *Expert Syst. Appl.* **36**, 7798–7808 (2009).
64. Carrasco, J., García, S., Rueda, M., Das, S. & Herrera, F. Recent trends in the use of statistical tests for comparing swarm and evolutionary computing algorithms: Practical guidelines and a critical review. *Swarm Evol. Comput.* **54**, 100665 (2020).
65. Tang, Z. et al. Chaotic wind driven optimization with fitness distance balance strategy. *Int. J. Comput. Intell. Syst.* **15**, 46 (2022).
66. Yang, H. et al. An intelligent metaphor-free spatial information sampling algorithm for balancing exploitation and exploration. *Knowl.-Based Syst.* **250**, 109081 (2022).
67. Abualigah, L., Elaziz, M. A., Sumari, P., Geem, Z. W. & Gandomi, A. H. Reptile search algorithm (RSA): A nature-inspired meta-heuristic optimizer. *Expert Syst. Appl.* **191**, 116158 (2022).
68. Yousri, D., Elaziz, M. A. & Mirjalili, S. Fractional-order calculus-based flower pollination algorithm with local search for global optimization and image segmentation. *Knowl.-Based Syst.* **197**, 105889 (2020).
69. Tang, D. Spherical evolution for solving continuous optimization problems. *Appl. Soft Comput.* **81**, 105499 (2019).
70. Guo, A. et al. An adaptive position-guided gravitational search algorithm for function optimization and image threshold segmentation. *Eng. Appl. Artif. Intell.* **121**, 106040 (2023).
71. Zhao, W., Wang, L. & Mirjalili, S. Artificial hummingbird algorithm: A new bio-inspired optimizer with its engineering applications. *Comput. Methods Appl. Mech. Eng.* **388**, 114194 (2022).
72. Wang, Y., Gao, S., Yu, Y., Cai, Z. & Wang, Z. A gravitational search algorithm with hierarchy and distributed framework. *Knowl.-Based Syst.* **218**, 106877 (2021).
73. Xia, X. et al. An expanded particle swarm optimization based on multi-exemplar and forgetting ability. *Inf. Sci.* **508**, 105–120 (2020).
74. Ram, J. P. et al. Analysis on solar PV emulators: A review. *Renew. Sustain. Energy Rev.* **81**, 149–160 (2018).
75. Ishaque, K., Salam, Z. & Syafaruddin, A. A comprehensive MATLAB simulink PV system simulator with partial shading capability based on two-diode model. *Sol. Energy* **85**, 2217–2227 (2011).
76. Allam, D., Yousri, D. & Eteiba, M. Parameters extraction of the three diode model for the multi-crystalline solar cell/module using moth-flame optimization algorithm. *Energy Convers. Manage.* **123**, 535–548 (2016).
77. El-Naggar, K., AlRashidi, M., AlHajri, M. & Al-Othman, A. Simulated annealing algorithm for photovoltaic parameters identification. *Sol. Energy* **86**, 266–274 (2012).
78. Easwarakhanthan, T., Bottin, I. B. J. & Boutrit, C. Nonlinear minimization algorithm for determining the solar cell parameters with microcomputers. *Int. J. Sol. Energy* **4**, 1–12 (1986).
79. Tong, N. T. & Pora, W. A parameter extraction technique exploiting intrinsic properties of solar cells. *Appl. Energy* **176**, 104–115 (2016).
80. Liang, J. et al. Parameters estimation of solar photovoltaic models via a self-adaptive ensemble-based differential evolution. *Sol. Energy* **207**, 336–346 (2020).
81. Yang, X., Gong, W. & Wang, L. Comparative study on parameter extraction of photovoltaic models via differential evolution. *Energy Convers. Manage.* **201**, 112113 (2019).
82. Askarzadeh, A. & Rezazadeh, A. Parameter identification for solar cell models using harmony search-based algorithms. *Sol. Energy* **86**, 3241–3249 (2012).
83. Luo, K. Water flow optimizer: A nature-inspired evolutionary algorithm for global optimization. *IEEE Trans. Cybern.* **52**, 7753–7764 (2022).
84. Tang, Z. et al. Fractional-order water flow optimizer. *Int. J. Comput. Intell. Syst.* **17**, 84 (2024).
85. Bayraktar, Z., Komurcu, M. & Werner, D. H. Wind driven optimization (WDO): A novel nature-inspired optimization algorithm and its application to electromagnetics. In *2010 IEEE Antennas and Propagation Society International Symposium*, 1–4 (IEEE, 2010).
86. Mirjalili, S., Mirjalili, S. M. & Lewis, A. Grey wolf optimizer. *Adv. Eng. Softw.* **69**, 46–61 (2014).
87. Lei, Z., Gao, S., Zhang, Z., Zhou, M. & Cheng, J. MO4: A many-objective evolutionary algorithm for protein structure prediction. *IEEE Trans. Evol. Comput.* **26**, 417–430 (2022).
88. Zhang, Y., Gao, S., Cai, P., Lei, Z. & Wang, Y. Information entropy-based differential evolution with extremely randomized trees and lightGBM for protein structural class prediction. *Appl. Soft Comput.* **136**, 110064 (2023).

89. Gao, S. et al. Dendritic neuron model with effective learning algorithms for classification, approximation, and prediction. *IEEE Trans. Neural Netw. Learn. Syst.* **30**, 601–614 (2019).
90. Gao, S. et al. Fully complex-valued dendritic neuron model. *IEEE Trans. Neural Netw. Learn. Syst.* **34**, 2105–2118 (2023).
91. Yu, Y. et al. Improving dendritic neuron model with dynamic scale-free network-based differential evolution. *IEEE/CAA J. Autom. Sin.* **9**, 99–110 (2022).

Acknowledgements
This work was mainly supported by the Research on Deep-layered Swarm Intelligence Algorithms and Their Application in Solar Photovoltaic Models under Grant NSF2024CB14.

Author contributions
Zhentao Tang: Writing- original draft, Methodology, Conceptualization, Software. Kaiyu Wang: Formal analysis, Writing- review & editing, Conceptualization. Lan Zhuang: Formal analysis, Writing- review & editing, Conceptualization. Mingxin Zhu: Data curation, Writing- review & editing. Yongxuan Yao: Data curation, Writing- review & editing. Huiqin Chen: Data curation, Writing- review & editing. Jing Li: Formal analysis, Validation. Xiaoxiang Wu: Formal analysis, Validation. Shangce Gao: Writing- review & editing, Resources, Project administration, Supervision. All authors read and approved the final manuscript.

Declarations

Competing interests
The authors declare no competing interests.

Ethical approval
No human or animal subjects were involved in this experiment.

Additional information
Correspondence and requests for materials should be addressed to S.G.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by-nc-nd/4.0/.

© The Author(s) 2025