Hindawi
International Journal of Energy Research
Volume 2024, Article ID 4322211, 25 pages
https://doi.org/10.1155/2024/4322211

Research Article

Wind Farm Layout Optimization Based on Dynamic Opposite Learning-Enhanced Sparrow Search Algorithm

Yun Zhu,[1] Yahui Guo,[1] Tianyu Hu,[2] Chengke Wu,[2] and Lidong Zhang[3]

[1]*School of Electrical Engineering, Guangxi University, Nanning 530004, China*
[2]*Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518172, China*
[3]*School of Energy and Power Engineering, Northeast Electric Power University, Jilin 132012, China*

Correspondence should be addressed to Lidong Zhang; lidongzhang@neepu.edu.cn

Received 4 October 2023; Revised 19 December 2023; Accepted 30 January 2024; Published 29 February 2024

Academic Editor: Abdulkeri Okbaz

In recent years, the proportion of wind power in new energy generation has gradually increased. The natural wind in wind farms is subject to velocity attenuation by the wake effect, so improving the efficiency of wind farm power generation has become a problem that must be solved for wind power generation. Considering the uncertainty of wind farms, we regard wind farm layout optimization (WFLO) as a strongly nonlinear problem. In this paper, we improve the sparrow search algorithm (SSA) using dynamic opposite learning (DOL) strategy. Twenty-eight benchmark test results prove that compared with other algorithms, the improved algorithm DOLSSA has excellent robustness and the ability of searching for a better solution when solving a strongly nonlinear optimization problem, and the DOL strategy effectively improves the shortcomings of the original algorithm which is prone to local optimization and space limitation. In this paper, the authors establish the dynamic rotational coordinates of wind farms and set six different physical scenarios by considering the wind direction and wind speed variables, and the results prove that the performance of DOLSSA is optimal.

1. Introduction

With the continuous advance and innovation of science, the dominant position of energy as a necessary consumable for the progress and development of human society is more difficult to shake than at any other time [1]. In the development process, with the continuous depletion of traditional fuel resources and environmental degradation, emerging clean and renewable energy as a substitute for traditional energy is gradually entering the stage [2]. Among the emerging renewable energy sources, wind power is more widely used than other clean energy due to its more mature technology, low cost, and environmentally friendly characteristics [3]. The vigorous construction of wind farms is beneficial to reduce the demand for fossil fuels [4], promoting the gradual replacement of nonrenewable energy by renewable energy in the electricity market [5] and establishing a new type of electricity system dominated by clean energy.

One of the existing problems in the construction of wind farms is the wake effect of the upstream wind turbine on the downstream wind turbine [6]. The wake effect of the wind turbine will seriously weaken the natural wind speed passing through the wind turbine [7], thus affecting the power generation of the downstream wind turbine, resulting in the loss of the wind farm economy and a significant increase in the cost of power generation [8]. In recent years, for the WFLO problem, many scholars have made improvements in the physical model to meet practical needs. In 2020, Reddy proposed a new framework with six wake models and four wake superposition scenarios, which provided new ideas for the experiments [9]. In 2021, Moreno et al. introduced a new objective function that considers the minimum cost of electricity, the total area of the wind farm, and the power loss in the wake effect [10]. In the same year, Guo et al. improved the effectiveness of the model by considering the influence of environmental factors based on the original wake model [11]. In 2022, Chen et al. developed a new two-stage WFLO model for offshore wind farms [12]. To improve the convergence accuracy of the computation, many excellent algorithms have been proposed [13]. In

2020, Long et al. improved the evolutionary success rate of the generalized regression neural algorithm by introducing an adaptive mechanism and reduced the computational cost of wind farm layout optimization through a learning process with a large amount of data [14]. In 2021, Yeghikian et al. used a particle swarm algorithm to optimize the treatment of multiple scenarios for the variation of the wind turbine heights as well as the number of wind turbines [15]. In 2022, Bai et al. used a Monte Carlo tree search to improve the adaptive algorithm, and the improved algorithm confirmed its optimality in the design of a certain wind farm in New Zealand [16]. In literature [17], the authors proposed the normal approximation method to determine the probability of future wind speed occurrence using historical data to balance the relationship between the maximum annual power production of a wind farm and the expectation of historical wind speed in the case of uncertain or unmeasured real-time wind speed. Lei et al. [18] solved the WFLO problem using PSO algorithm with a GA and achieved an average of 92.24 percent energy conversion under different scenarios. In 2023, Rizk-Allah and Hassanien proposed an algorithm combining balanced optimizer and pattern search, which improved the optimization algorithm's handling of irregular wind farm terrain capabilities [19]. Lei et al. [20] proposed a new variant of PSO (CGPSO) to improve the program quality of WFLO. Yu et al. [21] proposed the CLSHADE algorithm for profile optimization of wind energy conversion to reduce carbon emissions. Yang et al. used the improved SE algorithm (ISE) [22] to optimize the total power generation from wind farms. Although the above studies provide diversified solutions to the WFLO problem, the lack of introducing novel algorithms to solve the problem tends to cause researchers to easily fall into traditional thinking, which is not innovative enough, and the traditional optimization algorithms are prone to make the results fall into the dilemma of local optimums during the computation process. Therefore, the aim of this paper is to propose a new variant of the sparrow search algorithm (SSA) for application to WFLO.

At this stage, AI has been widely used in the energy field [23], and the heuristic optimization algorithm as a kind of AI has also played a great role [24]. WFLO problem as a typical single-objective optimization problem, SSA algorithm has good processing potential for this type of problem, so in this paper, SSA algorithm is introduced to solve this type of problem. SSA was first proposed by Xue and Shen [25] in 2020 and has been widely used in many areas of power system [26] due to its high accuracy and fast convergence speed. He et al. proposed a new optimization method for distribution network protection based on SSA, which improves the overall performance of the relay protection device [27]. Fathy et al. used it to achieve optimal energy management in microgrids from different perspectives [28]. In addition, applications of the SSA include research on transformer fault diagnosis [29], maximizing the energy efficiency of battery charging and discharging in modern communication networks [30], localization of voltage temporary sources in large-scale distribution networks with high penetration of distributed power sources [31], optimal configuration of the capacity of electric vehicle charging stations in a new type of power system [32], and cluster head selection in wireless sensor networks [33]. However, since the change of individual position in the iterative process of SSA is deeply affected by the previous foraging strategy, it will easily fall into the local optimal solution. To improve its shortcomings, in recent years, many scholars have introduced an improvement strategy based on SSA and put forward a variety of variations of the sparrow search algorithm for different physical scenarios [34]. Yuan et al. proposed an improved SSA (ISSA) to solve the distributed maximum power point tracking problem [35]. Wang et al. developed a chaotic SSA (CSSA), which demonstrated its advantages in solving cluster optimization problems compared to various other optimization algorithms [36]. Li et al. proposed an adaptive SSA to analyze the impact of the number of cars on microgrid scheduling for a particular islanded microgrid [37]. Hou et al. developed a chaotic quantum SSA (CQSSA) to solve the problem of identifying the parameters of a physical model of a lithium battery [38]. Li et al. proposed a multiobjective SSA (MOSSA) and proved its effectiveness by comparing it with four multiobjective optimization algorithms such as MOPSO and MOGWO [39]. Gao et al. used an extreme learning machine with SSA optimization to solve the medium-term load-side electricity demand forecasting problem [40]. Wang et al. developed a new variant of the SSA algorithm, PESSA, which proved its superiority in algorithmic improvement through several function tests and was successfully applied to the optimal path search problem in three-dimensional space [41]. SSA as a novel algorithm has appeared many variants in a short time, but many new variants are still unable to achieve real-time updates of iteration range. Therefore, in this paper, the DOL strategy is used to improve the original algorithm to achieve the improvement of population initialization adaptation and real-time change of population range to avoid local optimization, and the improved algorithm has significantly improved robustness and convergence accuracy compared with the original algorithm and other comparative algorithms. The idea of DOLSSA to solve the WFLO problem is analyzed in Figure 1.

The main research of this paper is to propose a novel variant of the SSA algorithm (DOLSSA) for solving the WFLO problem within a given wind farm. The main contributions of this paper are as follows: (1) A wind farm layout model combining the Jensen wake model, wind farm constraints, and rotation coordinates is introduced. (2) The new algorithm DOLSSA is proposed by introducing the DOL strategy to improve the shortcomings of the SSA algorithm. (3) Experiments are conducted for all benchmark functions in CEC2014 and WFLO problems in six different physical scenarios. The comparative analysis of the results proves that the new algorithm combines the advantages of SSA and DOL strategies and outperforms the existing algorithms in many aspects.

The rest of the paper is organized as follows: Section 2 describes the wake impact calculation method based on the Jensen wake model, the two-dimensional wind farm rotating coordinate model, the objective function, and the wind farm

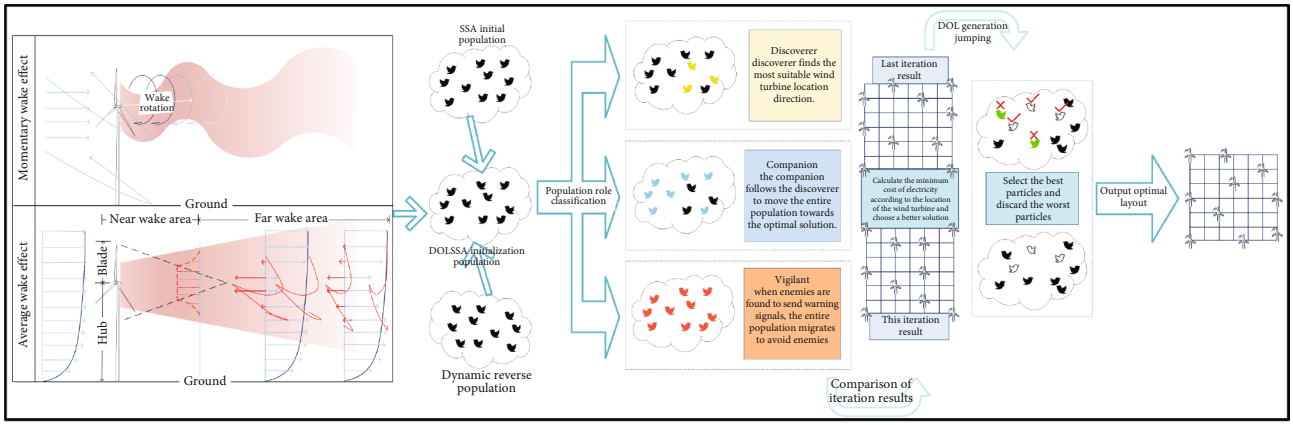


Figure 1: DOLSSA solution concept map.

constraints. Section 3 presents the principle and steps of DOLSSA. In Section 4, the results of the DOLSSA are compared with the other seven optimization algorithms both numerically and statistically in a benchmark function test environment and a real physical application environment. Finally, Section 5 summarizes the entire paper.

2. Physical Modeling in WFLO

2.1. Jensen Wake Model. Wind turbine power generation [42], which extracts energy from the wind, creates a wake downstream of the wind turbine, and wind speeds within the wake are attenuated. The area affected by the wake effect from the upstream wind turbine increases during the downstream propagation phase, and the wind speed gradually returns to the natural wind speed [43]. The wake effect is the macroscopic effect on wind power generation caused by the change in wind speed due to the interaction between WTGs. It is important to take into account the wake effects of neighboring wind turbines as well as the possible wake effects of the way the wind turbines are arranged in new wind farms. At this stage, the Jensen wake model has been widely used compared to other wake models because it can be applied to onshore and offshore wind farms with different turbine types and layouts with minimal computational resources [44]. By considering the effect of the upstream wind turbine wake on the downstream wind turbine, three scenarios can be considered in the Jensen wake model [45]. In the first scenario, the downstream wind turbine is completely outside the range of the upstream wake effect and can be considered unaffected by the upstream wake effect.

$$u_i = u, \tag{1}$$

where u is the initial wind speed of the incoming wind, unaffected by the wake effect, and u_i is the wind speed at the location of the downstream wind turbine i under the influence of the wake effect of the upstream wind turbine.

In the second case, the downstream wind turbine is completely within the range of the upstream wind turbine wake effect.

$$\begin{aligned} u_i &= u\left[1 - \frac{2a}{1 + E\left(D_T/R_p\right)^2}\right], \\ a &= 0.5\left(1 - \sqrt{1 - C_t}\right), \\ E &= \frac{1}{2\ln\left(H/Z_0\right)}, \end{aligned} \tag{2}$$

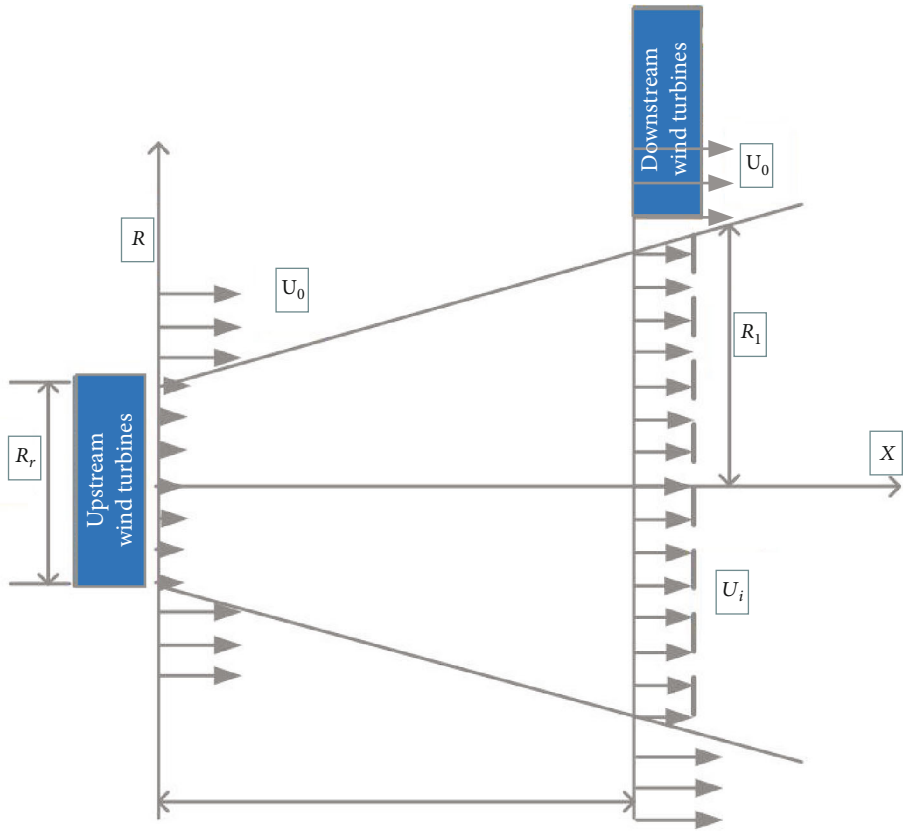
where a is the axis induction coefficient of the wind turbine; E is the entrainment coefficient; D_T is the distance between the upstream and downstream wind turbines relative to the x-axis on the two-dimensional map surface; R_p is the radius of the wake formed by the upstream wind turbine at the location of the downstream wind turbine due to the wake effect; C_t is the thrust coefficient of the wind turbine, which determines its efficiency in converting wind energy into electricity; H is the hub height of the wind turbine; and Z_0 is the length of the surface roughness and refers to the height of the downstream wind turbine at the location where the wind speed is zero.

In the third case, the downstream turbine is partially within the wake of the upstream turbine.

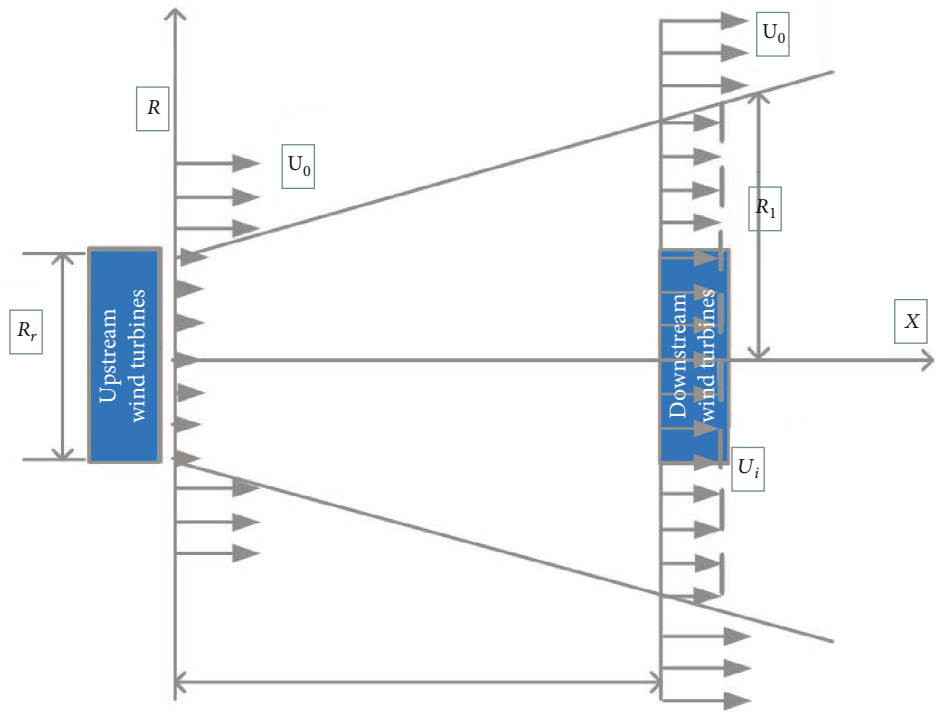
$$u_i = u\left(1 - \frac{2a}{E\left(D_T/R_p\right)^2}\right)\frac{S_w}{S_t}, \tag{3}$$

where S_t and S_w in the above equation are the downstream wind turbine rotor's rotating area and the area of the downstream wind turbine rotor's rotating area affected by the upstream wind turbine, respectively, and E is the entrainment coefficient. The Jensen wake model is shown in Figure 2.

Since the downstream wind turbine may be affected by the wake effect of multiple upstream wind turbines during

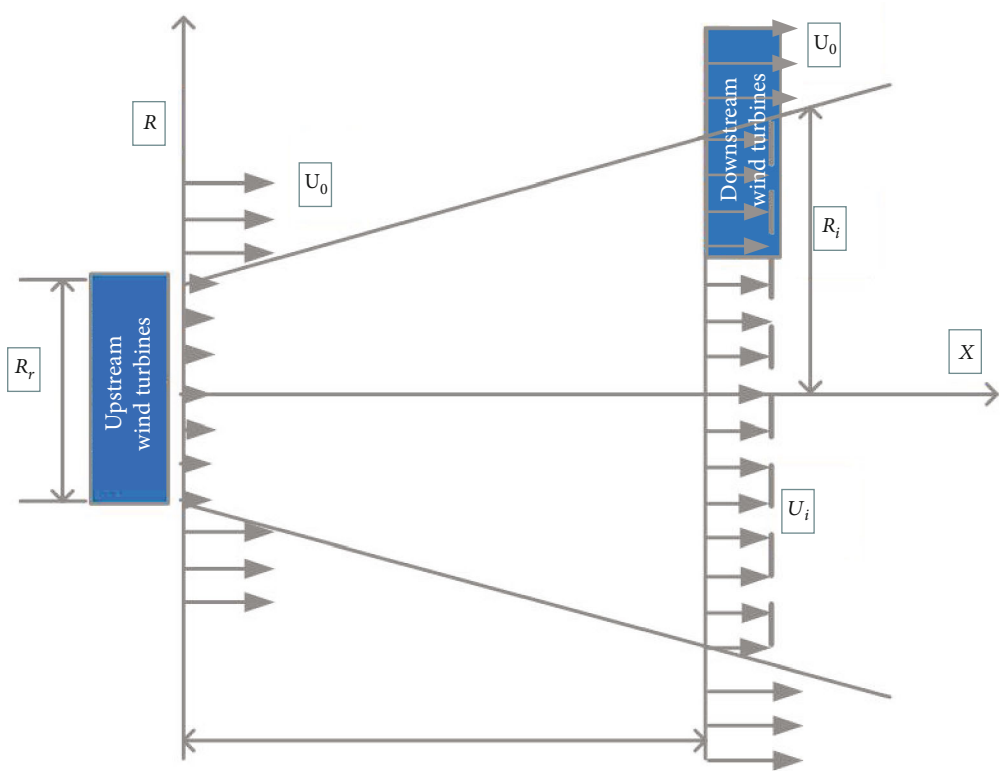


(a) No effect



(b) Partial effect

Figure 2: Continued.



(c) Full effect

Figure 2: Wind turbine wake effect impact map.

the generation process, the introduction of a wake superposition equation is considered, where m_j is the number of upstream wind turbines acting on the downstream wind turbine.

$$u_i = u\left[1 - \sqrt{\sum_{j=1}^{m_j}\left(1 - \frac{2a}{1 + E\left(D_T/R_p\right)^2}\right)_j}\right]. \tag{4}$$

2.2. Definition of Two-Dimensional Coordinate Rotation for Wind Farms. A wind farm can be considered as a certain defined area on a two-dimensional plane, and we can define the location of the wind turbines in this area using square parcellations and using two-dimensional coordinates [46].

$$L = \begin{bmatrix} x_1 & x_2 & \cdots & x_n \\ y_1 & y_2 & \cdots & y_n \end{bmatrix}. \tag{5}$$

Considering that the wind direction is not constant, the definition of rotation is added to the original 2D model. When the wind direction changes along angle ε, we can assume that the two-dimensional model in which the wind farm is located also changes with the angle, so that the wind direction is always consistent with the direction of the nacelle of the wind turbine, and the two-dimensional coordinates of the wind turbine can be expressed as follows in the new rotation space:

$$L = \begin{bmatrix} \cos\varepsilon & -\sin\varepsilon \\ \sin\varepsilon & \cos\varepsilon \end{bmatrix} \begin{bmatrix} x_1 & x_2 & \cdots & x_n \\ y_1 & y_2 & \cdots & y_n \end{bmatrix}. \tag{6}$$

2.3. Objective Function. The overall objective of this paper is to achieve the maximum annual power generation at a certain installation and maintenance cost, which can also be interpreted as achieving the minimum cost of electricity per unit, where the main decision variable is the location of the wind turbines in the wind farm. Under the condition of determination of the number of generators, the objective function of this study is as follows:

$$O_b = \min\left(\frac{C_T + M_a}{P_t}\right), \tag{7}$$

where M_a is the annual maintenance cost of the wind turbine, P_t is the annual electricity production of the whole wind farm, and C_T is the construction cost of the wind turbine, where M_a and C_T mainly depend on the number of wind turbines, which is calculated by the following formulas:

$$C_T = N\left(\frac{2}{3} + \frac{1}{3}e^{-0.00174N^2}\right),$$
$$M_a = C_m N, \tag{8}$$

where N is the number of wind turbines and C_m is the one-year maintenance cost of a single wind turbine.

In this paper, all the wind turbines in the wind farm are divided into two categories according to their location wind speeds, where the location wind speeds are equal to the natural wind speeds, and we consider them as the upstream wind turbines, totaling v, and the remaining w wind turbines are located in locations where the wind speeds are lower than the natural winds due to wind speed attenuation:

$$N = v + w,$$
$$P_t = P_{wt} + P_{wa}. \tag{9}$$

According to the classification of wind turbines, in this paper, the annual power generation P_t of wind farms is divided into two components, P_{wt} and P_{wa}, where P_{wt} refers to the total power generation of v wind turbines that are not affected by the wake effect and P_{wa} is the total power generation of w wind turbines that are affected by the wake effect.

$$P_w = \frac{1}{2}\eta\rho\pi r^2 u^3,$$
$$\eta = C_p\eta_m\eta_e, \tag{10}$$

where ρ is the air density and η is the power efficiency of the wind turbine, which can also be interpreted as the ratio of the electrical energy output of the wind turbine to the wind energy input. The size of the value of η depends on the three factors of C_p, η_m, and η_e, which represent the power coefficient of the wind turbine, the efficiency of the mechanical transmission, and the efficiency of the conversion of electrical energy, respectively, which are set to $C_p = 0.4$, $\eta_m = 0.75$, and $\eta_e = 0.3$, and the values can be introduced to obtain the following equations:

$$P_{wt} = 24 \times 365 \times \sum_{j=1}^{v}\frac{1}{2}\eta\rho\pi r^2 u_j^3,$$
$$P_{wa} = 24 \times 365 \times \sum_{i=1}^{w}\frac{1}{2}\eta\rho\pi r^2 u_i^3. \tag{11}$$

2.4. Restrictions. Considering the uncertainty of the wind farm layout, several related constraints are introduced here. The first is the wind turbine distance constraint, where the spacing between wind turbines is a factor in the design of a wind farm, and the smaller the spacing between wind turbines, the greater the impact of the interaction between wind turbines (the wake effect). Conversely, arbitrarily increasing the spacing between wind turbines can increase the cost of laying roads and extending cables, as well as increasing the amount of land required to install a given capacity of wind turbine units. The minimum spacing between wind turbines is generally greater than or equal to 3 times the rotor diameter of the wind turbine [47], which is chosen as 5 times in this paper. The second constraint is the wind farm boundary constraint [48], which does not allow the coordinates of the selected wind turbine locations to extend beyond the length of the boundary, ensuring that each wind turbine location is within the wind farm. The third constraint is the wind turbine overlap constraint [49], which prevents two wind turbine locations from overlapping in the wind farm. The fourth constraint ensures that each wind turbine in the wind farm is of the same model to prevent variability in hub height, rotor diameter, energy conversion efficiency, and other factors between wind turbines.

Wind turbine distance constraint:

$$\sqrt{\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2} \geq 5R_r, \tag{12}$$

where $i = 1, 2, \cdots, v$ and $j = 1, 2, \cdots, w$.

Wind farm boundary constraint:

$$0 \leq x_k \leq W,$$
$$0 \leq y_k \leq L, \tag{13}$$

where $k = 1, 2, \cdots, N$.

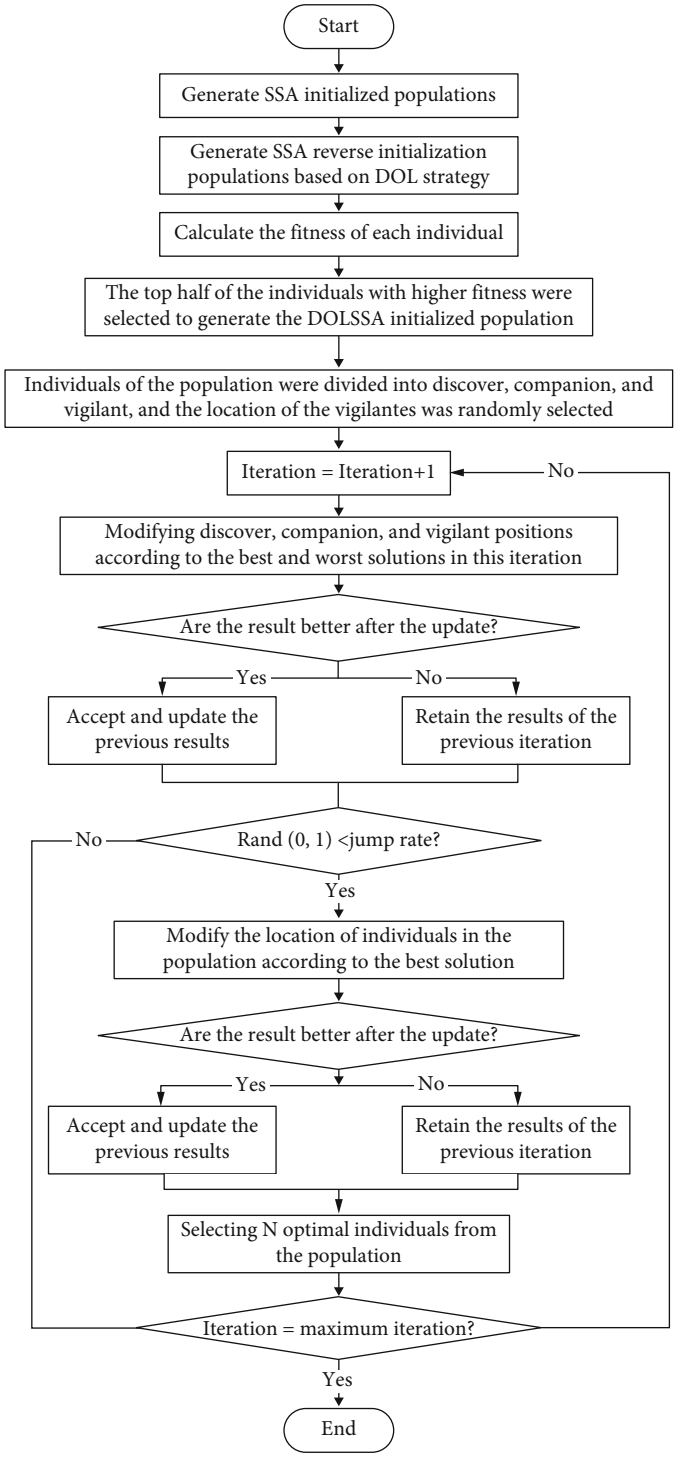
Wind turbine overlap constraint:

$$x_i \neq x_j\,; y_i \neq y_j, \tag{14}$$

where R_r is the rotor diameter of the wind turbine; x_k and y_k are the horizontal and vertical coordinates of the k^{th} wind turbine in the two-dimensional wind farm, respectively; and W and L are the boundaries of the wind farm.

3. Algorithm Preparation

3.1. Sparrow Search Algorithm. SSA is an optimization algorithm summarized by Xue and Shen in 2020 [25] to summarize the predation of sparrow populations in nature as well as the escape from predation. In this optimization algorithm, all members of the sparrow population can be divided into three roles: discoverer, companion, and vigilant. The discoverer, as the pioneer of the whole population of sparrows in the food search, is only a small part of the whole population, and its main function is to spread out in all directions to determine the optimal direction for the population to search for food in the process of foraging; in addition to the discoverer, all the other sparrows in the population can be regarded as companions, and the companions will listen to the discoverer's guidance to obtain food for the population; in addition, each individual in the population can be vigilant, and when they encounter danger, they will



Algorithm 1 The framework of DOLSSA algorithm

Randomly generate an initial population P;
for $i = 1; i \le N; i++$ **do**
  $r1_i = rand(0,1), r2_i = rand(0,1)$;
  for $j = 1; j \le D; j++$ **do**
    $oP_{i,j}^{DO} = oP_{i,j} + r1_i * (r2_i * (a_j + b_j - oP_{i,j}) - oP_{i,j})$;
    Check the boundaries;
  end for
end for
Select N number of the fittest individuals from $P \cup oP^{DO}$;
Set $G = 0$;
Input:
  Gm: the maximum iterations
  D: the number of Discoverers
  V: the number of vigilants
  n: the number of Sparrows
  ST: the warning range
  Initialize the sparrow population and define its related parameters.
while $G < Gm$
  Rank the fitness values and find the current best-valued individual and the current worst-valued individual;
$R_2 = rand(0,1)$;
for $i = 1 : D$
Update the population position through the formula 23;
  end for
for $i = D + 1 : n$
Update the population position through the formula 24;
  end for
for $i = 1 : V$
Update the population position through the formula 26;
  end for
Get the new locations;
Compare population strengths and weaknesses and make substitutions
if $rand > Jr$; $G++$; back to 12;
if $rand < Jr$ **then**
  for $i = 1; i \le N; i++$ **do**
  $r3_i = rand(0,1)$, $r4_i = rand(0,1)$;
    for $j = 1; j \le D; j++$ **do**
      $a_j = \min(oP_{ij})$, $b_j = \max(oP_{ij})$
      $oP_{i,j}^{DO} = oP_{i,j} + \omega r3_i * (r2_i * (a_j + b_j - oP_{i,j}) - oP_{i,j})$;
      Check the boundaries
    end for
  end for
  Select N number of the fittest individuals from $P \cup oP^{DO}$
  $G++$;
  end if
end while

Figure 3: DOLSSA schematic flowchart and pseudocode.

Table 1: Parameter comparison table.

	F1	F3	Mean F4	F7	HF4	AR
$w = 12$, $Jr = 0.2$	2.996*E*+07	7.020*E*+03	1.161*E*+02	2.709*E*-02	8.058*E*+02	12.2
$w = 13$, $Jr = 0.2$	7.651*E*+06	1.430*E*+04	1.114*E*+02	1.232*E*-02	3.908*E*+03	12.0
$w = 14$, $Jr = 0.2$	7.117*E*+06	1.381*E*+04	1.170*E*+02	2.702*E*-02	1.792*E*+03	12.0
$w = 15$, $Jr = 0.2$	7.438*E*+06	1.562*E*+04	7.166*E*+01	1.723*E*-02	9.451*E*+02	7.8
$w = 16$, $Jr = 0.2$	1.265*E*+07	1.054*E*+04	7.718*E*+01	2.704*E*-02	1.718*E*+03	11.2
$w = 17$, $Jr = 0.2$	5.933*E*+06	2.238*E*+04	1.319*E*+02	3.447*E*-02	2.420*E*+03	13.8
$w = 12$, $Jr = 0.3$	5.751*E*+06	2.640*E*+04	1.403*E*+02	1.477*E*-02	1.096*E*+04	13.6
$w = 13$, $Jr = 0.3$	7.557*E*+06	1.285*E*+04	7.934*E*+01	1.234*E*-02	7.672*E*+02	7.6
$w = 14$, $Jr = 0.3$	7.119*E*+06	2.881*E*+04	7.976*E*+01	1.232*E*-02	2.181*E*+03	11.8
$w = 15$, $Jr = 0.3$	5.862*E*+06	1.002*E*+04	6.954*E*+01	1.232*E*-02	2.113*E*+02	**1.8**
$w = 16$, $Jr = 0.3$	6.980*E*+06	1.020*E*+04	1.441*E*+02	1.478*E*-02	4.309*E*+02	9.2
$w = 17$, $Jr = 0.3$	5.928*E*+06	2.622*E*+04	7.716*E*+01	1.233*E*-02	1.408*E*+03	6.8
$w = 12$, $Jr = 0.4$	6.984*E*+06	2.702*E*+04	1.351*E*+02	7.396*E*-02	1.422*E*+03	15.2
$w = 13$, $Jr = 0.4$	1.285*E*+07	2.499*E*+04	8.031*E*+01	2.713*E*-02	1.899*E*+03	15.0
$w = 14$, $Jr = 0.4$	8.202*E*+06	2.802*E*+04	7.672*E*+01	1.477*E*-02	1.670*E*+03	11.6
$w = 15$, $Jr = 0.4$	2.161*E*+07	2.728*E*+04	7.799*E*+01	7.396*E*-02	1.914*E*+03	16.4
$w = 16$, $Jr = 0.4$	7.140*E*+06	7.601*E*+04	8.150*E*+01	9.857*E*-02	1.506*E*+03	15.8
$w = 17$, $Jr = 0.4$	8.057*E*+06	2.514*E*+04	7.614*E*+01	7.396*E*-02	1.172*E*+04	14.8
$w = 12$, $Jr = 0.5$	8.759*E*+06	2.486*E*+04	1.014*E*+02	7.393*E*-02	3.183*E*+03	16.0
$w = 13$, $Jr = 0.5$	6.071*E*+06	4.723*E*+04	1.051*E*+02	9.865*E*-02	2.520*E*+03	17.0
$w = 14$, $Jr = 0.5$	8.767*E*+06	4.368E+04	1.145*E*+02	1.232*E*-02	2.689*E*+04	17.0
$w = 15$, $Jr = 0.5$	5.996*E*+06	2.065E+04	7.877*E*+01	2.216*E*-02	1.977*E*+03	10.0
$w = 16$, $Jr = 0.5$	8.903*E*+06	2.761E+04	1.322*E*+02	1.969*E*-02	1.678*E*+03	16.2
$w = 17$, $Jr = 0.5$	6.493*E*+06	1.098*E*+04	1.205*E*+02	9.765*E*-02	1.508*E*+04	15.2

Table 2: Basic parameters of other algorithms.

Algorithm	Basic parameters
PSO	$c_1 = c_2 = 2.05$; $w = 0.729$
Jaya	—
cfwPSO	$c_1 = c_2 = 2.05$; $w = 0.729$
cfPSO	$c_1 = c_2 = 2.05$; $w = 0.729$
NTLBO	—
ETLBO	—
DE	$F = 0.7$; $CR = 0.5$
SSA	—

find danger signals to nearby individuals, and through the continuous transmission of the signals, the whole population will perform a wide range of migration to avoid being captured by the natural enemies.

Assuming that there are n sparrows in the whole population and the dimension of the problem variable to be optimized is d, the positions of all individual sparrows in the population during foraging can be expressed as follows:

$$X = \begin{bmatrix} X_{1,1} & X_{1,2} & \cdots & X_{1,d} \\ X_{2,1} & X_{2,2} & \cdots & X_{2,d} \\ \vdots & \vdots & \ddots & \vdots \\ X_{n,1} & X_{n,2} & \cdots & X_{n,d} \end{bmatrix}. \tag{15}$$

The formula for discoverer position change in each iteration is as follows:

$$X_{i,j}^{t} = \begin{cases} X_{i,j}^{t-1} \cdot \exp\left(\dfrac{-i}{wG_{\max}}\right), & \text{if } R_2 > \text{ST}, \\ X_{i,j}^{t-1} + QL, & \text{otherwise}, \end{cases} \tag{16}$$

where $X_{i,j}^{t}$ is the position of the i^{th} individual in the population in the j^{th} dimension during the t^{th} iteration, $i = 1, 2, \cdots, n$; $G_{\max}$ is the maximum number of iterations set by

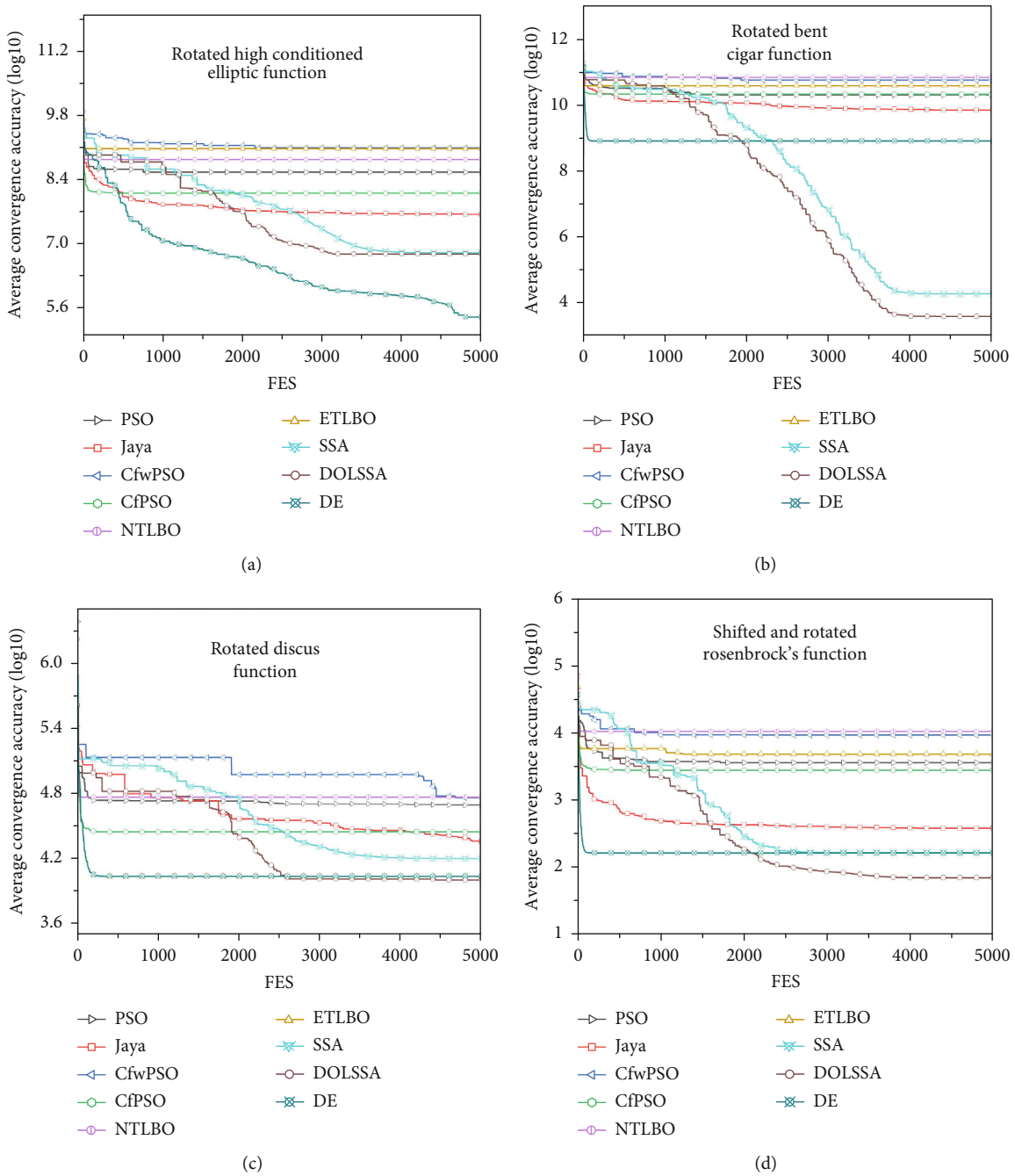


Figure 4: Continued.

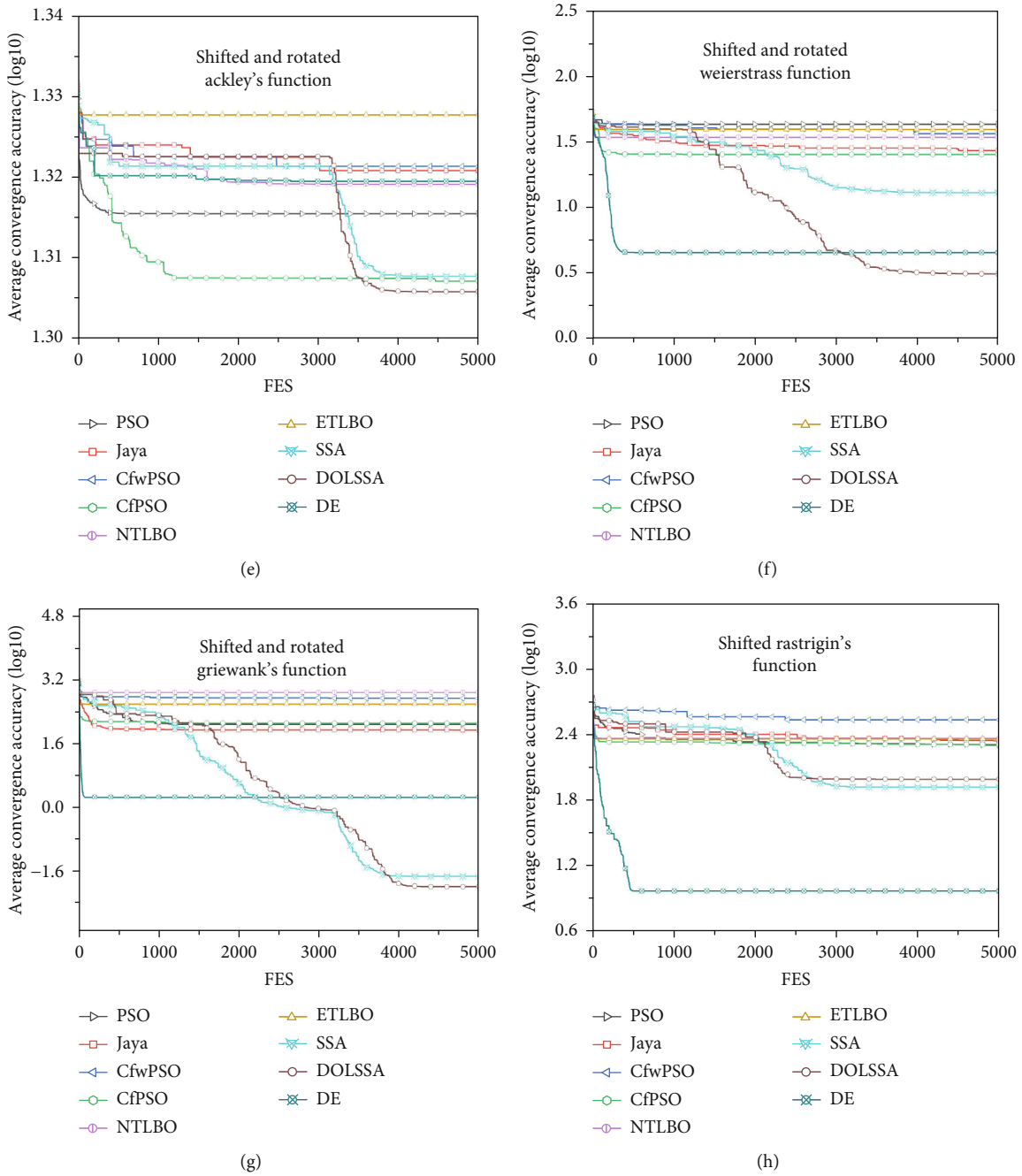


Figure 4: Continued.

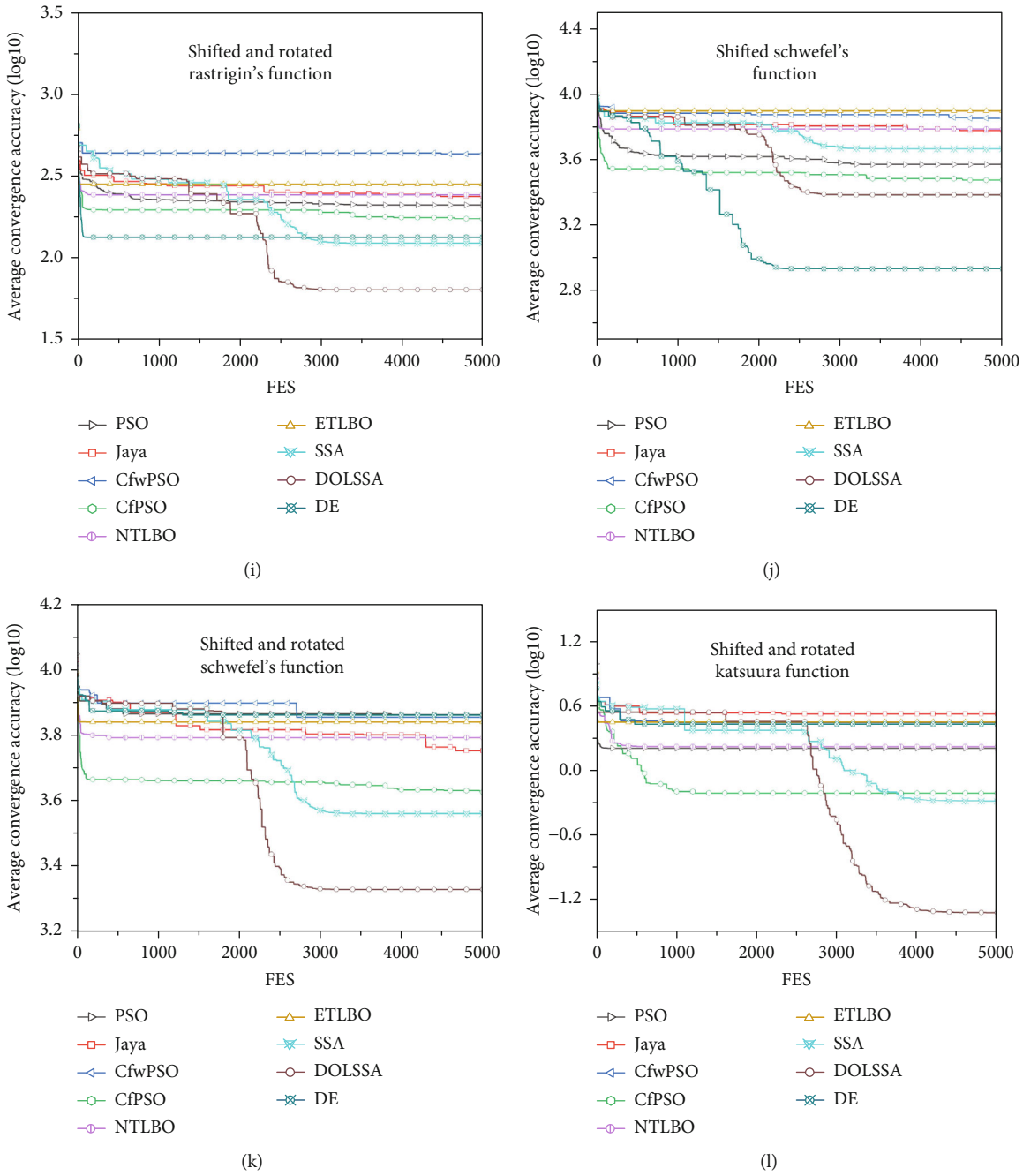


Figure 4: Continued.

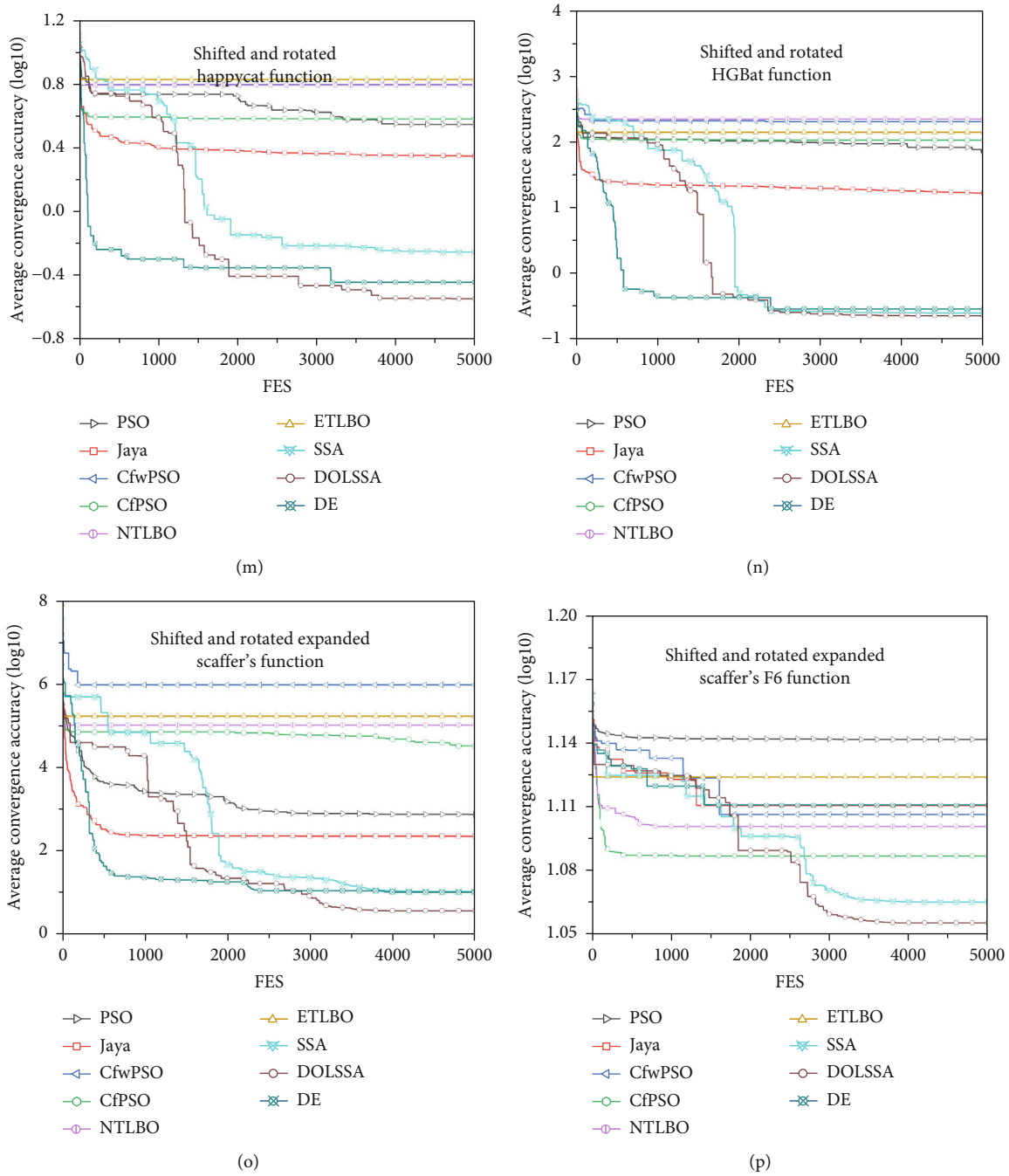


Figure 4: Comparison of unimodal and multimodal function convergence curves.

the SSA in solving the problem to be optimized; w and R_2 are random numbers arbitrarily selected within [0, 1]; Q is a normally distributed random number; L is the row matrix with d columns, and all elements in the matrix are 1; $ST \in [0.5, 1]$; R_2 and ST represent the safety range and warning range of the discoverer in the process of foraging; if the safety range is smaller than the warning range, it means that there is no natural enemy within the foraging environment and the discoverer can forage with ease; on the contrary, the population must migrate to change the location to avoid predation by the natural enemy.

The formula for companion position change in each iteration is as follows:

$$X_{i,j}^{t} = \begin{cases} Q \cdot \exp\left(\dfrac{X_{\text{worst}}^{t-1} - X_{i,j}^{t-1}}{i^2}\right), & \text{if } i > \dfrac{n}{2}, \\ X_{\text{best}}^{t} + \left|X_{i,j}^{t-1} - X_{\text{best}}^{t}\right| A^{+} L, & \text{otherwise}, \end{cases} \tag{17}$$

$$A^{+} = A^{T}\left(AA^{T}\right)^{-1},$$

Table 3: Comparison of unimodal and multimodal function test results.

(a)

Algorithm	F1		F2		F3		F4	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std
PSO	3.569*E*+08	4.396*E*+08	1.949*E*+10	1.681*E*+10	4.898*E*+04	1.428*E*+04	3.663*E*+03	6.132*E*+03
Jaya	4.258*E*+07	2.930*E*+08	6.762*E*+09	1.371*E*+10	2.270*E*+04	3.308*E*+04	3.787*E*+02	4.994*E*+03
cfwPSO	1.207*E*+09	7.593*E*+08	4.549*E*+10	1.116*E*+10	5.763*E*+04	1.606*E*+04	9.464*E*+03	7.650*E*+03
cfPSO	1.261*E*+08	8.990*E*+08	2.042*E*+10	1.425*E*+10	2.770*E*+04	3.584*E*+04	2.834*E*+03	2.335*E*+03
NTLBO	6.757*E*+08	1.766*E*+09	6.676*E*+10	2.402*E*+10	5.849*E*+04	2.532*E*+04	1.066*E*+04	1.402*E*+04
DE	2.523*E*+05	8.103*E*+07	7.923*E*+08	3.761*E*+09	1.085*E*+04	1.834*E*+04	1.632*E*+02	6.992*E*+02
ETLBO	1.161*E*+09	7.492*E*+08	3.718*E*+10	2.679*E*+10	1.085*E*+04	3.115*E*+04	4.899*E*+03	1.102*E*+04
SSA	6.265*E*+06	3.513*E*+07	1.901*E*+04	8.386*E*+03	1.579*E*+04	7.594*E*+03	1.638*E*+02	6.179*E*+01
DOLSSA	5.862*E*+06	2.282*E*+07	4.013*E*+03	1.073*E*+04	1.002*E*+04	2.363*E*+04	6.954*E*+01	3.678*E*+01
Algorithm	F5		F6		F7		F8	
PSO	2.068*E*+01	9.489*E*-02	4.330*E*+01	8.368*E*+00	1.280*E*+02	1.108*E*+02	2.015*E*+02	6.221*E*+01
Jaya	2.093*E*+01	1.291*E*-02	2.721*E*+01	5.687*E*+00	9.230*E*+01	1.580*E*+02	2.206*E*+02	9.992*E*+01
cfwPSO	2.096*E*+01	1.864*E*-02	3.662*E*+01	5.693*E*+00	5.621*E*+02	2.095*E*+02	3.438*E*+02	7.286*E*+01
cfPSO	2.029*E*+01	4.295*E*-02	2.555*E*+01	2.196*E*+00	1.363*E*+02	1.479*E*+02	2.048*E*+02	4.275*E*+01
NTLBO	2.085*E*+01	3.452*E*-02	3.445*E*+01	2.536*E*+00	7.820*E*+02	1.887*E*+02	2.321*E*+02	4.177*E*+01
DE	2.087*E*+01	5.308*E*-02	4.529*E*+00	5.728*E*+00	2.027*E*+00	2.808*E*+01	9.307*E*+00	2.405*E*+01
ETLBO	2.127*E*+01	6.118*E*-02	3.942*E*+01	3.863*E*+00	4.049*E*+02	1.243*E*+02	2.296*E*+02	1.396*E*+01
SSA	2.031*E*+01	9.318*E*-02	1.303*E*+01	1.143*E*+01	2.215*E*-02	2.956*E*-03	8.358*E*+01	5.991*E*+01
DOLSSA	2.023*E*+01	6.536*E*-02	3.123*E*+00	4.501*E*+00	1.232*E*-02	2.025*E*-02	9.850*E*+01	4.728*E*+01
Algorithm	F9		F10		F11		F12	
PSO	2.103*E*+02	4.673*E*+01	3.700*E*+03	1.671*E*+03	7.313*E*+03	1.631*E*+03	1.629*E*+00	3.697*E*-01
Jaya	2.371*E*+02	1.048*E*+02	5.947*E*+03	2.223*E*+03	5.669*E*+03	3.527*E*+02	3.400*E*+00	2.794*E*-01
cfwPSO	4.294*E*+02	9.916*E*+01	7.091*E*+03	9.259*E*+02	7.182*E*+03	2.043*E*+03	2.828*E*+00	1.640*E*-01
cfPSO	1.723*E*+02	9.160*E*+01	2.972*E*+03	8.508*E*+02	4.240*E*+03	1.249*E*+03	6.248*E*-01	2.965*E*-01
NTLBO	2.426*E*+02	3.527*E*+01	6.096*E*+03	3.324*E*+02	6.223*E*+03	7.986*E*+02	1.689*E*+00	5.560*E*-01
DE	1.328*E*+02	5.162*E*+01	8.473*E*+02	8.454*E*+02	7.285*E*+03	6.953*E*+02	2.725*E*+00	4.062*E*-01
ETLBO	2.808*E*+02	4.188*E*+01	7.847*E*+03	2.376*E*+02	6.940*E*+03	1.477*E*+03	2.853*E*+00	5.735*E*-01
SSA	1.224*E*+02	8.539*E*+01	4.611*E*+03	8.057*E*+03	3.633*E*+03	4.021*E*+02	5.227*E*-01	3.618*E*-01
DOLSSA	6.368*E*+01	6.095*E*+01	2.414*E*+03	8.920*E*+03	2.128*E*+03	8.999*E*+02	4.817*E*-02	1.393*E*-01
Algorithm	F13		F14		F15		F16	
PSO	3.542*E*+00	2.512*E*-01	6.841*E*+01	2.879*E*+01	7.475*E*+02	2.396*E*+05	1.385*E*+01	1.871*E*-01
Jaya	2.249*E*+00	2.594*E*-01	1.660*E*+01	3.165*E*+01	2.248*E*+02	2.566*E*+05	1.289*E*+01	2.480*E*-01
cfwPSO	6.285*E*+00	4.005*E*-01	1.994*E*+02	6.370*E*+01	9.961*E*+05	5.946*E*+05	1.277*E*+01	4.721*E*-01
cfPSO	3.842*E*+00	5.193*E*-01	1.053*E*+02	9.696*E*+01	3.456*E*+04	1.546*E*+06	1.221*E*+01	4.298*E*-01
NTLBO	6.240*E*+00	3.224*E*-01	2.191*E*+02	5.332*E*+01	1.086*E*+05	5.407*E*+06	1.260*E*+01	5.113*E*-01
DE	3.650*E*-01	3.445*E*-01	2.860*E*-01	1.048*E*+01	1.034*E*+01	5.398*E*+05	1.290*E*+01	6.986*E*-01
ETLBO	6.760*E*+00	2.829*E*-01	1.387*E*+02	5.835*E*+01	1.779*E*+05	6.232*E*+06	1.330*E*+01	2.920*E*-01
SSA	5.626*E*-01	7.084*E*-02	2.476*E*-01	1.671*E*-02	1.058*E*+01	1.244*E*+01	1.161*E*+01	7.425*E*-01
DOLSSA	2.881*E*-01	6.054*E*-02	2.263*E*-01	3.188*E*-02	3.703*E*+00	1.437*E*+01	1.136*E*+01	5.498*E*-01

(b)

Algorithm	ON	AR
PSO	0	5.75
Jaya	0	5.12
cfwPSO	0	7.88
cfPSO	0	4.56
NTLBO	0	7.06
DE	3	3.25
ETLBO	0	7.44
SSA	0	2.69
DOLSSA	**13**	**1.25**

where X_{best} and X_{worst} are the locations of the best- and worst-adapted individuals, respectively, during the iteration process, to allow the other companions to move closer to the location of the best-adapted individual, while the poorly adapted companions leave their original positions to continue foraging. Like L, A is also a d column row matrix, but A contains both 1 and -1 elements.

The formula for vigilant position change in each iteration is as follows:

$$X_{i,j}^{t} = \begin{cases} X_{\text{best}}^{t-1} + \beta \cdot \left| X_{i,j}^{t-1} - X_{\text{best}}^{t-1} \right|, & \text{if } f_i > f_{\text{best}}, \\ X_{i,j}^{t-1} + K \cdot \left[\dfrac{\left| X_{i,j}^{t-1} - X_{\text{worst}}^{t-1} \right|}{(f_i - f_{\text{worst}}) + \varepsilon} \right], & \text{if } f_i = f_{\text{best}}, \end{cases} \tag{18}$$

where β is a normally distributed random number, K is a random number arbitrarily chosen in the interval (-1, 1) to represent the direction of movement of individuals in the population, ε is a constant that tends infinitely to 0 to avoid the error of dividing by 0, and f is the value of the fitness of the individuals. If $f_i > f_{\text{best}}$, the vigilant is at the edge of the population; if $f_i = f_{\text{best}}$, the vigilant is in a foraging-optimal position and needs to move closer to the edge of the population.

3.2. Dynamic Opposite Learning. Oppositional-based learning(OBL) [50] involves generating inverted particles for random particle positions in the initialization phase of the population, treating all random and inverted individuals as an initialized cluster, and collecting the top half of high-fitness clusters as the initial population.

$$\dot{x} = a + b - x. \tag{19}$$

The above formula is a confirmation formula for the position of an inverse particle in a one-dimensional space, where x is a random particle, $\dot{x}$ is an inverse particle, and a and b are the boundaries of the space. The formula can be generalized to higher dimensions:

$$\dot{x}_j = a_j + b_j - x_j ; j = 1 : d. \tag{20}$$

When the dimension of the space in which the random particle is located is d, $x = (x_1, x_2, \cdots, x_d)$, a_j and b_j are the boundaries of the d-dimensional space, respectively.

Although OBL can perform population initialization optimization, the search area does not change due to the fixed area of the environment in which the population is located, and it is impossible to narrow or move the search area, which easily makes the algorithm's calculation results fall into the local optimum. To solve this problem, Xu et al. introduced a weighting factor in OBL in 2020 and proposed dynamic opposite learning (DOL) [51], which can continuously change the population environment and improve the diversity of population changes in the iterative process. Compared with OBL, it effectively avoids the local optimization problem and improves the performance of the algorithm. The DOL strategy improvement means can be divided into two steps, DOL population initialization and DOL generation jumping [52], and its principle is described as follows.

3.2.1. DOL Population Initialization.

$$oP_{i,j}^{\text{DO}} = oP_{i,j} + r1_i * \left(r2_i * \left(a_j + b_j - oP_{i,j} \right) - oP_{i,j} \right), \tag{21}$$

where $oP_{i,j}$ is a random initial population, $oP_{i,j}^{\text{DO}}$ is a DOL-based initial population, $r1_i$ and $r2_i$ are both random numbers in the range [0, 1], and the population size is set to P. Consistent with OBL, a_j and b_j still denote the boundaries of the space.

$$oP_{i,j}^{\text{DO}} = \text{rand}\left(a_j, b_j \right), \text{if } oP_{i,j}^{\text{DO}} < a_j \left\| oP_{i,j}^{\text{DO}} > b_j. \right. \tag{22}$$

With this step, the initial population with the highest fitness can be formed by selecting the P best individuals from the two populations.

3.2.2. DOL Generation Jumping. DOL provides an implementation strategy for population diversification, but the DOL strategy is not implemented in every population iteration [53]. The decision of whether a DOL strategy is required or not is made regarding the jump rate, and during the j^{th} iteration, if the random number selected is less than

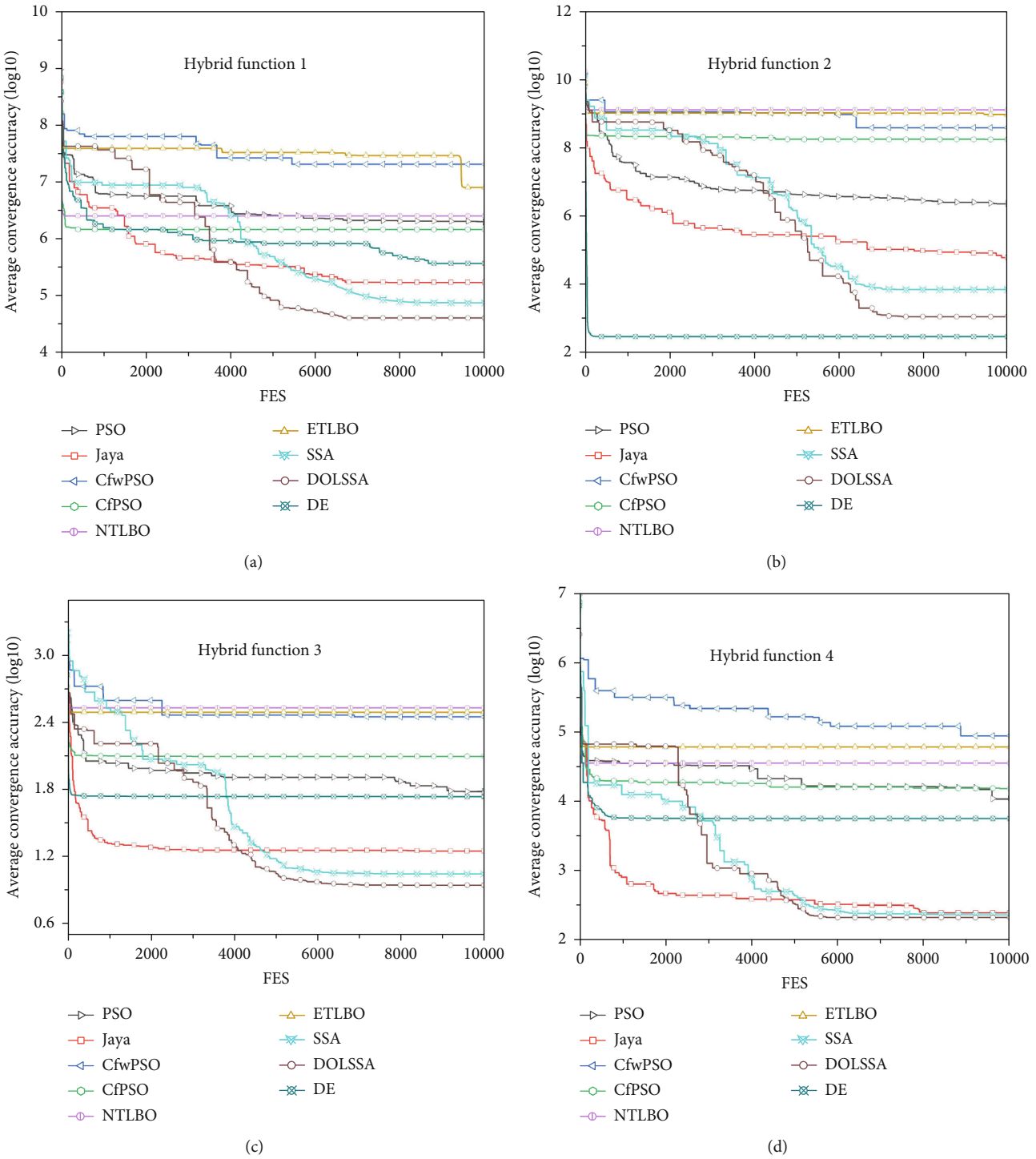


Figure 5: Continued.

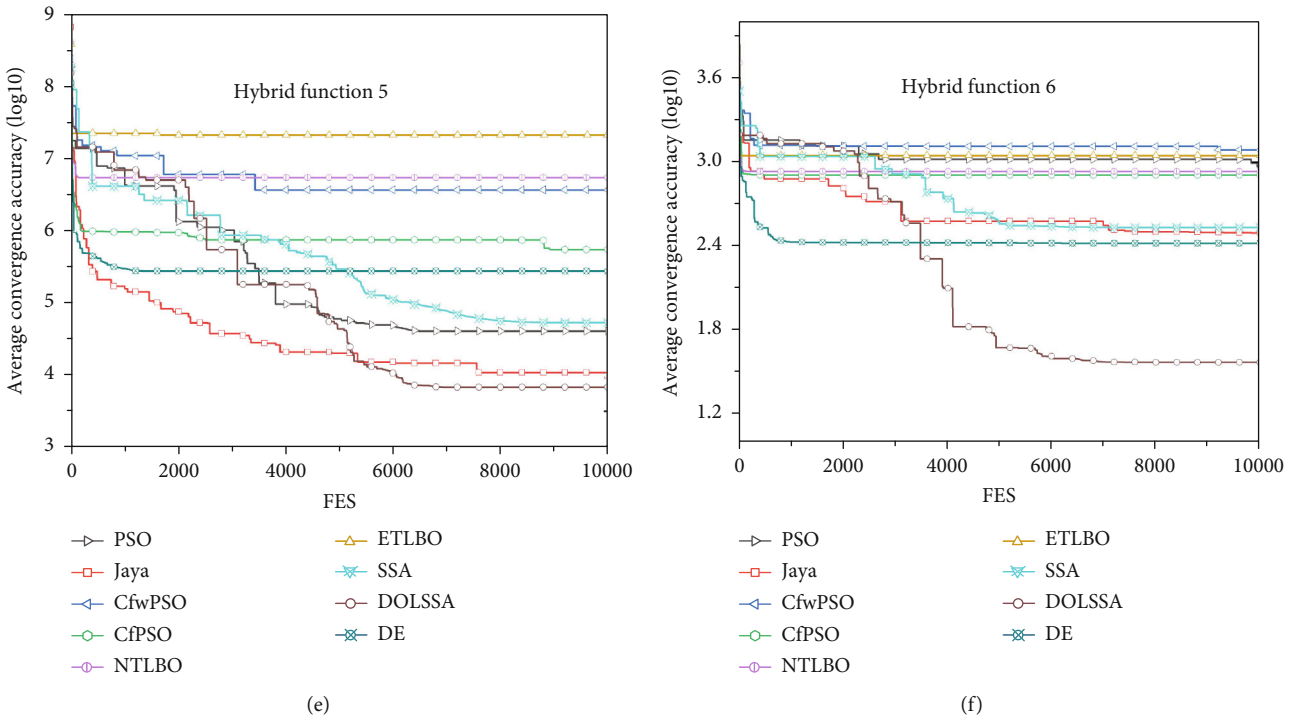


Figure 5: Comparison of hybrid function convergence curves.

Table 4: Comparison of hybrid function test results.

Algorithm	HF1		HF2		HF3		HF4	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std
PSO	1.967*E*+06	4.507*E*+07	2.265*E*+06	2.709*E*+09	6.138*E*+01	4.276*E*+02	1.096*E*+04	6.712*E*+04
Jaya	1.687*E*+05	4.752*E*+07	5.810*E*+04	3.677*E*+09	1.791*E*+01	8.289*E*+01	2.471*E*+02	2.720*E*+04
cfwPSO	2.068*E*+07	1.093*E*+08	3.848*E*+08	3.752*E*+09	2.838*E*+02	4.295*E*+02	8.962*E*+04	9.819*E*+04
cfPSO	1.449*E*+06	1.120*E*+08	1.735*E*+08	4.199*E*+09	1.264*E*+02	7.041*E*+02	1.549*E*+04	5.098*E*+04
NTLBO	2.550*E*+06	3.326*E*+08	1.277*E*+09	6.234*E*+09	3.418*E*+02	1.227*E*+03	3.612*E*+04	2.256*E*+05
DE	3.655*E*+05	1.169*E*+07	2.823*E*+02	1.036*E*+08	5.510*E*+01	2.135*E*+01	5.653*E*+03	1.208*E*+05
ETLBO	8.117*E*+06	2.907*E*+08	9.350*E*+08	6.076*E*+09	3.130*E*+02	1.734*E*+03	6.165*E*+04	2.340*E*+05
SSA	7.432*E*+04	1.830*E*+06	6.768*E*+03	1.773*E*+03	1.125*E*+01	2.679*E*+01	2.324*E*+02	7.350*E*+03
DOLSSA	4.025*E*+04	1.408*E*+06	1.077*E*+03	4.106*E*+02	8.893*E*+00	3.502*E*+01	2.113*E*+02	2.172*E*+04

Algorithm	HF5		HF6		ON	AR
PSO	4.092*E*+04	3.047*E*+07	8.838*E*+02	1.597*E*+03	0	5.17
Jaya	9.749*E*+03	4.165*E*+06	3.119*E*+02	9.115*E*+02	0	3.00
cfwPSO	3.717*E*+06	4.300*E*+07	1.227*E*+03	2.105*E*+03	0	8.00
cfPSO	5.527*E*+05	5.610*E*+07	8.106*E*+02	1.400*E*+03	0	5.67
NTLBO	5.499*E*+06	5.814*E*+07	8.600*E*+02	1.968*E*+04	0	7.67
DE	2.777*E*+05	1.795*E*+06	2.640*E*+02	2.972*E*+02	1	3.33
ETLBO	2.077*E*+07	1.676*E*+08	1.117*E*+03	6.197*E*+04	0	8.17
SSA	5.346*E*+04	5.068*E*+05	3.424*E*+02	1.026*E*+03	0	2.83
DOLSSA	6.813*E*+03	5.182*E*+05	3.735*E*+01	8.039*E*+02	5	1.17

Table 5: Wilcoxon rank sum test results.

Algorithm	DOLSSA											
	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12
PSO	>	>	>	>	>	>	>	>	>	>	>	>
Jaya	>	>	=	>	>	>	>	>	>	>	>	>
cfwPSO	>	>	>	>	>	>	>	>	>	>	>	>
cfPSO	>	>	>	>	>	>	>	>	>	>	>	>
NTLBO	>	>	>	>	>	>	>	>	>	>	>	>
ETLBO	>	>	>	>	>	>	>	>	>	>	>	>
DE	<	>	>	>	>	>	>	<	>	=	>	>
SSA	=	>	>	>	=	>	>	>	>	>	>	>
Algorithm	F13	F14	F15	F16	HF1	HF2	HF3	HF4	HF5	HF6	>/=/<	
PSO	>	>	>	>	>	>	>	>	>	>	22/0/0	
Jaya	>	>	>	>	>	>	>	>	>	>	21/1/0	
cfwPSO	>	>	>	>	>	>	>	>	>	>	22/0/0	
cfPSO	>	>	>	=	>	>	>	>	>	>	21/1/0	
NTLBO	>	>	>	>	>	>	>	>	>	>	22/0/0	
ETLBO	>	>	>	>	>	>	>	>	>	>	22/0/0	
DE	>	>	>	>	>	<	>	>	>	>	18/1/3	
SSA	>	>	>	>	>	>	>	=	>	>	21/3/0	

Table 6: WFLO solution results.

Algorithm	Case 1	Case 2	Case 3	Case 4	Case 5	Case 6	ON	AR
PSO	3.004*E*-05	1.098*E*-06	9.747*E*-06	5.822*E*-07	1.004*E*-05	9.502*E*-06	0	7.33
Jaya	4.244*E*-06	5.292*E*-07	2.196*E*-06	2.088*E*-07	6.300*E*-06	1.807*E*-06	0	4.00
cfwPSO	2.762*E*-05	1.037*E*-06	5.294*E*-06	5.600*E*-07	1.047*E*-05	3.775*E*-06	0	6.33
cfPSO	3.259*E*-05	5.451*E*-07	9.907*E*-06	1.398*E*-07	1.181*E*-05	4.461*E*-06	0	6.50
NTLBO	3.037*E*-06	3.938*E*-07	2.194*E*-06	1.314*E*-07	3.585*E*-06	3.075*E*-07	0	2.50
DE	4.492*E*-06	1.141*E*-06	5.035*E*-06	5.856*E*-07	5.414*E*-07	4.497*E*-06	0	6.00
ETLBO	1.190*E*-05	2.599*E*-06	5.169*E*-03	1.542*E*-07	3.585*E*-06	5.403*E*-05	0	6.83
SSA	3.034*E*-06	5.312*E*-07	2.196*E*-06	2.274*E*-07	5.334*E*-07	3.570*E*-02	0	4.50
DOLSSA	2.945*E*-06	3.449*E*-07	2.180*E*-06	1.228*E*-07	5.334*E*-07	1.616*E*-07	6	1.00

the jump rate, then the iteration is required to follow the DOL strategy:

$$oP_{i,j}^{\mathrm{DO}} = oP_{i,j} + \omega r3_i * \left(r4_i * \left(a_j + b_j - oP_{i,j}\right) - oP_{i,j}\right),$$
$$a_j = \min\left(oP_{i,j}\right), \tag{23}$$
$$b_j = \max\left(oP_{i,j}\right).$$

During the iterative process, the population search area is constantly changing and its spatial boundaries are updated in real time.

3.3. DOL-Based SSA. Compared with other algorithms, the SSA algorithm takes into account the randomness of the position of individuals in the population during the search process in the optimization of the problem and has a strong global search ability, and its search strategy is more flexible and has fast convergence speed and high accuracy that can be better adapted to different types of problems. However, the change of individual position in the iterative process of sparrow search algorithm is deeply affected by the previous foraging strategy, which will lead to its easy to fall into the local optimal solution, so the DOL strategy is introduced, forming a new variant of sparrow search algorithm—DOLSSA.

The improved algorithm DOLSSA has several new steps compared to the original algorithm and its overall flow is shown in Figure 3.

4. Example Analysis

To comprehensively verify the performance of DOLSSA in solving optimization problems, DOLSSA is compared and analyzed with PSO [54], Jaya [55], SSA [25], DE [56], and some improved optimization algorithms (cfwPSO [57],

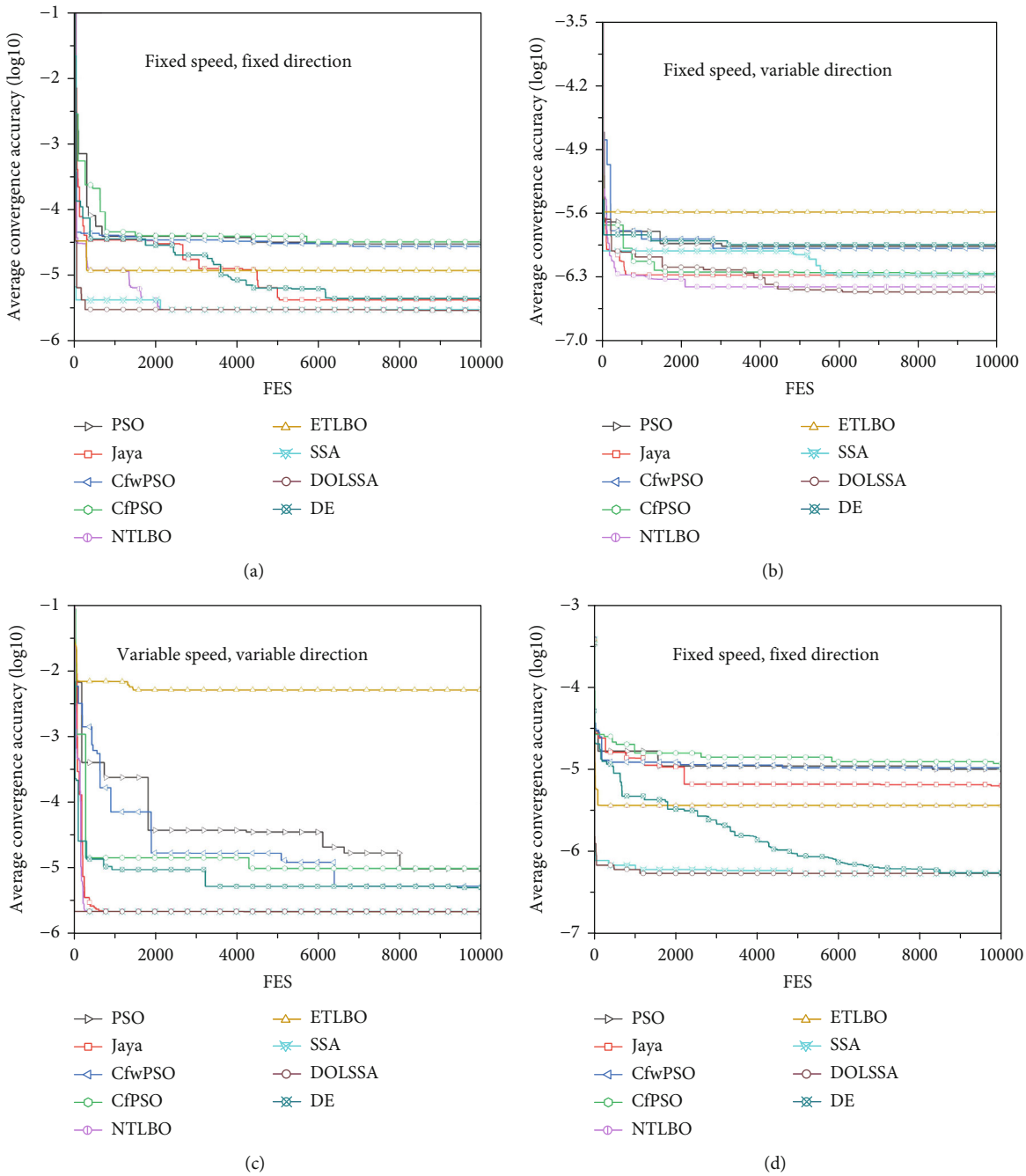


Figure 6: Continued.

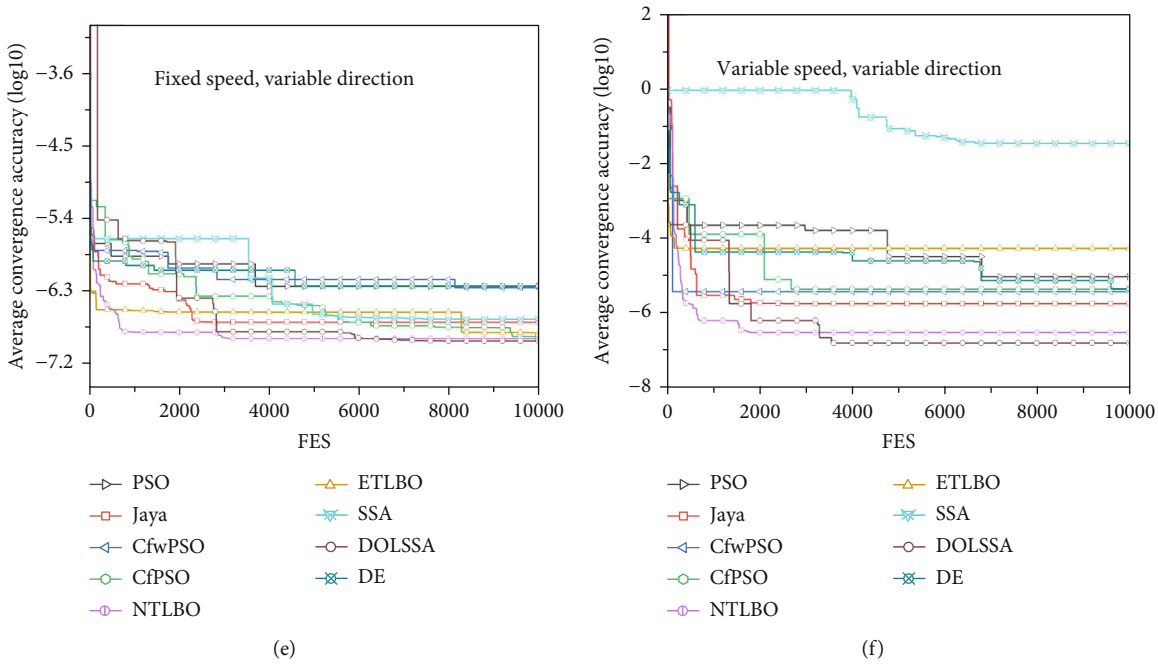


Figure 6: Comparison of WFLO convergence curves.

cfPSO [57], NTLBO [58], and ETLBO [59]) for PSO and TLBO against the experimental results, and the whole experiment is divided into three parts:

(i) For all unimodal functions, multimodal functions, and hybrid functions in the CEC2014 [60] test suite, multiple sets of repeated experiments are performed using DOLSSA and other optimization algorithms, respectively, and the data from multiple sets of experiments are averaged for comparison

(ii) Significant differences were analyzed by the Wilcoxon rank sum test [61] for the results obtained in experiment 1

(iii) The wind farm layout optimization problem is divided into three cases: directional fixed speed, nondirectional fixed speed, and nondirectional variable speed. In the three different cases, the problem is solved to derive the answer and compare the optimal solution

The computer configuration used in this paper is Inter i7-10700F CPU and AMD Radeon R5 430 GPU, and the programming language is MATLAB.

4.1. Benchmark Test. During the experiment, the parameters of DOLSSA algorithm are set as follow: according to the recommendations in [50, 53], the values of w and Jr are 15 and 0.3, respectively. To ensure the rationality of the parameter selection, the authors set the value range of w to 12-17 and the value range of Jr to 0.2-0.5 in this paper. In the evaluation of 24 parameter combinations by five benchmark functions F1, F3, F4, F7, and HF4, select the best combination by comparing the mean of the five calculations of the DOLSSA algorithm. Average ranking is the average rank of the algorithm among all algorithms. The results, exhibited in Table 1, indicate that the algorithm obtains the best performance at $w = 15$ and $Jr = 0.3$. The parameters of each algorithm are set as shown in Table 2. To ensure the fairness of the experiment, set the number of populations for all algorithms to 50.

In this experiment, all unimodal functions, multimodal functions, and hybrid functions in CEC2014 are selected, a total of 22 functions. For all algorithms, the population size is set to 50, and the problem dimension is set to 30. The maximum number of iterations is set to 5000 for unimodal and multimodal functions, and the maximum number of iterations is set to 10000 for hybrid functions. For each algorithm, the mean (Mean) and standard deviation (Std) were calculated for five randomly selected results. According to the experimental results, the optimal number and average ranking of each algorithm in the overall experiment are counted. In all tables of this paper, AR denotes the average ranking and ON denotes the optimal number of times.

The convergence curves of the average computation results of several optimization algorithms for the 16 nonmixed functions are shown in Figure 4, and their optimal solutions are recorded in Table 3. DOLSSA achieves the optimal number of times among the 16 single functions 10 times, which is 9 more than all the other optimization algorithms, and its average ranking is 1.25. Figure 5 and Table 4 show the convergence curves and convergence results of the various optimization algorithms in the six hybrid function tests, with DOLSSA performing best in five of the six tests, with an average rank of 1.17. The average ranking of DOLSSA is ranked first in both sets of benchmark tests, which indicates that DOLSSA performs optimally and has the best performance in this experimental part.

Table 7: Comparison of running rates of different algorithms.

Running time (s)	PSO	Jaya	cfwPSO	cfPSO	NTLBO	DE	ETLBO	SSA	DOLSSA
wt1	24.04	23.714	139.02	46.43	45.32	20.42	45.35	1.61	10.61
wt2	24.92	21.57	162.53	46.17	44.49	18.84	45.76	1.50	11.99
wt3	22.60	21.40	102.92	43.46	44.66	18.59	45.78	2.88	10.58
wt4	22.81	22.98	73.99	45.54	44.29	20.86	48.70	1.47	10.48
wt5	23.88	21.89	57.86	44.47	46.30	19.06	43.17	4.35	11.29
wt6	46.57	41.65	172.57	93.95	87.16	51.84	85.42	7.66	17.96
wt7	22.35	23.41	162.55	45.80	45.03	18.82	44.70	1.62	11.05
wt8	20.93	25.85	143.09	43.66	44.65	18.70	43.13	3.77	10.61
wt9	25.04	22.81	154.29	45.28	47.64	18.67	44.59	1.52	10.59
wt10	22.79	22.27	106.62	46.28	44.73	18.76	45.80	1.72	11.11
wt11	23.21	23.53	90.42	46.48	45.75	19.40	45.32	1.83	11.08
wt12	25.81	25.05	66.07	54.03	51.89	24.21	53.54	2.67	12.17
wt13	25.99	25.01	69.81	55.05	51.52	24.20	51.49	2.61	11.86
wt14	21.88	22.35	142.04	44.75	45.15	17.34	46.38	1.56	10.39
wt15	22.13	21.75	149.16	44.47	44.38	17.76	43.67	1.57	11.01
wt16	21.94	22.18	54.13	46.95	43.86	17.98	44.16	1.60	10.36
HF1	52.41	52.53	287.19	101.99	100.53	40.02	100.25	3.61	24.87
HF2	49.66	49.13	208.75	103.70	101.17	40.96	102.29	4.23	23.81
HF3	60.15	56.73	236.39	120.81	116.65	53.04	115.99	5.83	27.69
HF4	50.47	49.78	167.00	104.30	98.69	38.57	100.45	4.27	23.09
HF5	51.32	49.89	216.66	103.07	100.64	40.71	101.74	3.52	25.67
HF6	52.42	53.72	175.70	110.14	103.65	41.97	104.40	3.50	23.46
Case 1	17359.33	9088.36	17632.02	27698.24	18927.36	23917.22	21054.65	3584.46	2894.15
Case 2	22280.72	12236.27	16910.49	33490.52	25678.75	24227.29	20365.82	4355.44	2799.29
Case 3	10907.01	8077.98	12479.35	20951.03	15711.32	15522.33	18307.37	2657.81	2562.42
Case 4	23422.66	18635.63	48127.99	48025.36	32596.33	40159.25	49128.37	8913.25	7639.36
Case 5	19104.84	18741.12	49839.66	49589.83	36847.64	37018.04	44042.77	6374.57	6201.06
Case 6	26273.12	19359.76	49359.91	50526.07	37786.63	39279.50	46214.07	8175.30	6507.15
Average time	4287.89	3100.52	7053.15	8275.64	6033.79	6454.44	7161.25	1218.78	1033.40

Observing each of the subfigures in Figures 4 and 5, we can clearly see that in all the pictures, the downward trend of the convergence curve of DOLSSA is earlier than that of SSA, which can prove that the DOL strategy expands the initial search range of the population and improves the convergence of the algorithm in the process of early iteration by establishing a reverse population to select the better-adapted individuals to generate the initial population. In the meantime, the convergence accuracy of DOLSSA is higher than the other eight algorithms compared in the vast majority of cases (Figures 4(b)–4(g), 4(i), and 4(k)–4(p) and 5(a) and 5(c)–5(f)). In some benchmark tests (Figures 4(c)–4(f), 4(j), 4(m), and 4(o) and 5(e) and 5(f)), the SSA algorithm is inferior to other algorithms, and the phenomenon that the improved algorithm DOLSSA outperforms other compared algorithms proves that the DOL strategy effectively improves the convergence accuracy of the algorithm through the jump rate.

In summary, the DOLSSA algorithm has better processing potential and computational power in dealing with single-objective optimization problems.

4.2. Wilcoxon Rank Sum Test. To further validate the performance of DOLSSA, statistical analyses were performed using the Wilcoxon rank sum test. In Table 5, > and < indicate that DOLSSA outperforms and underperforms the comparison algorithm on the problem, respectively, and = indicates that there is no significant difference between DOLSSA and the comparison algorithm. The last column of statistics in Tables 5 clearly shows that DOLSSA outperforms the comparison algorithm in the vast majority of function tests, which is due to the assistance of the DOL strategy for selecting the initial population of SSA and dynamically changing the population range.

4.3. WFLO Problem Application. In this section, several optimization algorithms will be used to solve the WFLO problem. In Section 2, this paper describes in detail the physical model of WLFO, the objective function of the solution, and the optimization constraints of the turbine location. Two physical scenarios are set up: locating 10 wind turbines in a $1\,km * 1\,km$ footprint and locating 25 wind turbines in a $2\,km * 2\,km$ footprint. Natural wind is divided into three

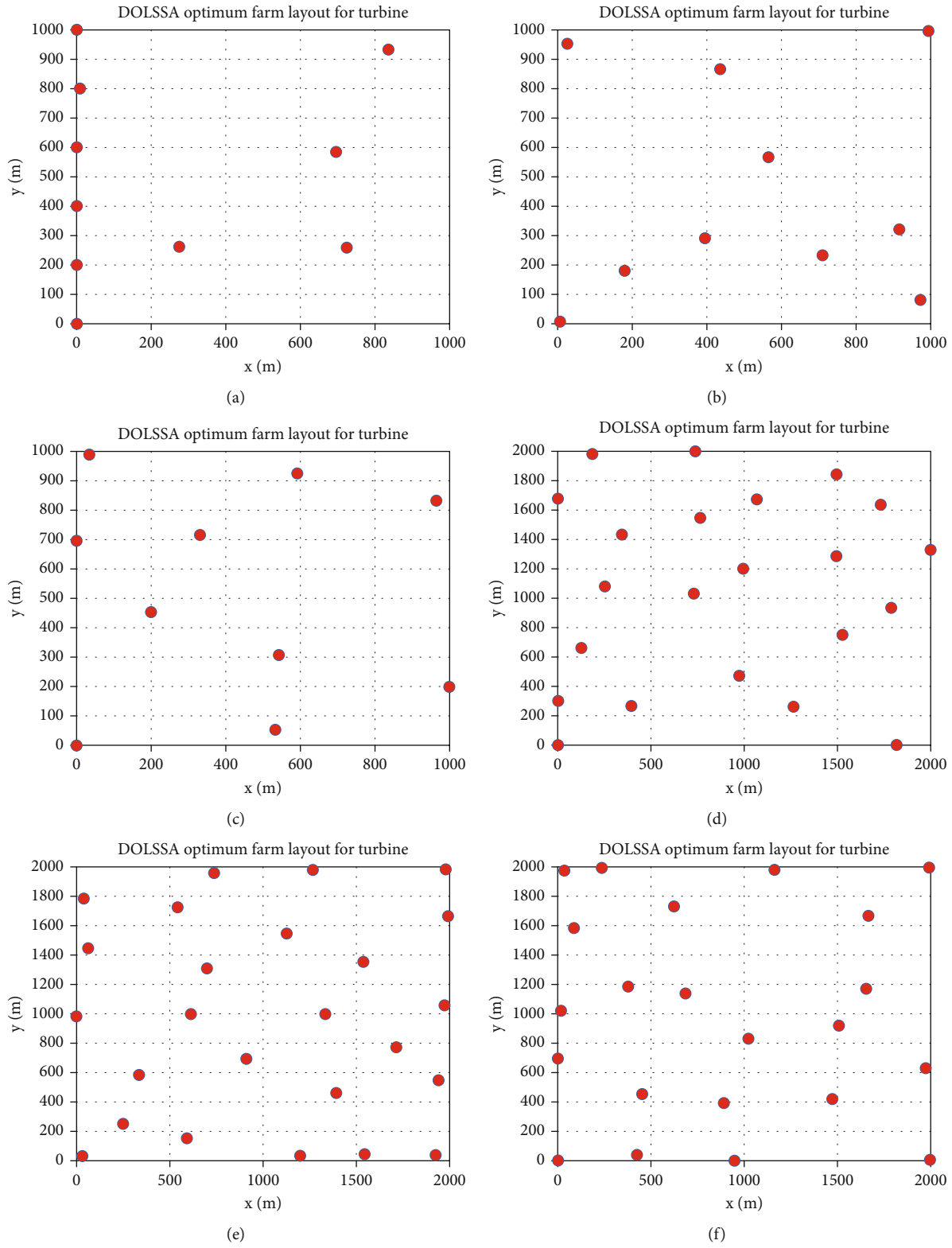


Figure 7: Wind turbine drop point map.

categories: directional fixed speed, nondirectional fixed speed, and nondirectional variable speed. The power of the wind turbine is set to be 1 MW, the hub height to be 60 meters, the blade length to be 40 meters, the thrust coefficient to be 0.88, and the surface roughness to be 0.3. Layout optimization is performed for two physical scenarios under three natural wind conditions: directional fixed speed, nondirectional fixed speed, and nondirectional variable speed.

Table 6 and Figure 6 show that DOLSSA solves optimally in all six physical scenarios. Observing Figures 6(b)–6(d) and 6(f), it can be found that the average ranking of optimization effectiveness of SSA is 5.6, and its optimization-seeking performance is poor. However, the optimization-seeking ability of DOLSSA is optimal in all working conditions by introducing DOL strategy. In six different scenarios, the DOLSSA solution results are improved by an average of 30.73% compared to SSA, respectively. The time comparison data in Table 7 proves that the computational efficiency of the SSA algorithm itself is much higher than that of the other compared algorithms, while the DOLSSA algorithm perfectly combines the advantages of the SSA algorithm and the DOL strategy. The innovative algorithm improves the shortcomings of the original algorithm, which is easy to fall into the local optimum, and retains the original computational efficiency and improves the accuracy of the algorithm, and the computational results also show that DOLSSA has a greater potential in practical engineering applications. The wind turbines in the wind farms in all cases are shown in Figure 7.

Based on this chapter, DOLSSA has excellent performance in function testing and practical physical applications, which is due to the impact of DOL strategy on population diversity and the balance between excellent optimization ability of SSA and dynamic search range changes.

5. Conclusion

In this paper, a new variant of SSA is proposed for solving the WFLO problem. DOLSSA combines the advantages of the DOL strategy and SSA and improves its robustness and convergence compared to the original algorithm. The paper first defines the physical model of the wind farm, the model cost includes the wind turbine construction cost and the annual maintenance cost, the wake effect between the wind turbines is evaluated according to the Jensen wake model, and the physical scenarios are categorized into three cases according to the speed and direction of the natural winds. For the case of variable wind direction, the study utilizes rotated two-dimensional plane coordinates to locate the spatial coordinates of the wind farm. The second part introduces the DOL strategy combined with SSA to improve the population diversity with the help of two steps, DOL population initialization and DOL generation jumping, which helps the original algorithm to better avoid the problem of easy to fall into the local optimization in the computation process as well as to improve the overall convergence of the algorithm. Finally, DOLSSA is compared and analyzed with various other comparative algorithms containing SSA in terms of data and statistics from multiple perspectives through three comparative experiments, and the results demonstrate the sensitivity and robustness of DOLSSA to optimal solutions.

In this paper, we demonstrate that DOLSSA can be an effective tool for solving strongly nonlinear complexity problems. Especially in the field of WFLO, DOLSSA is applicable to different power generation scenarios and effectively improves the power generation efficiency of wind farms. In the future, we will apply the new algorithms proposed in this paper to more practical applications, and the research in this paper can be applied to the study of scheduling strategies for emerging high-percentage new energy microgrid power systems to improve the overall power generation efficiency of microgrid clusters.

Nomenclature

u: Initial wind speed of the incoming wind (m/s)
u_i: Wind speed of wind turbine i (m/s)
a: Axis induction coefficient
ED_T: Distance to the x-axis (m)
R_p: Wake radius (m)
C_t: Thrust coefficient
H: Hub height (m)
Z_0: Surface roughness (μm)
S_ω: Rotating area of rotor affected (m^2)
S_t: Rotating area of the rotor (m^2)
ε: Wind angle degree
O_b: Minimum cost of one kilowatt hour (\$/kWh)
C_T: Construction costs of wind farm (\$)
M_a: Annual maintenance costs of wind farm (\$)
P_t: Annual electricity production of wind farm (kWh)
N: Number of wind turbines
C_m: Annual maintenance costs of single wind turbine (\$)
v: Number of upstream wind turbines
w: Number of downstream wind turbines
P_{wt}: Annual electricity production of upstream wind farm (kWh)
P_{wa}: Annual electricity production of downstream wind farm (kWh)
η: Density ($\mathrm{kg/m}^3$)
ρ: Wind energy conversion efficiency
C_p: Power coefficient
η_m: Mechanical transmission efficiency
η_e: Electrical energy conversion efficiency.

Data Availability

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

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

Yun Zhu and Lidong Zhang conceived and designed the overall study and managed and coordinated the execution of the experiments. Yahui Guo, Tianyu Hu, and Chengke Wu conceived the algorithmic improvements and designed the comparative experiments. Yahui Guo and Tianyu Hu analyzed the data and wrote the manuscript. All authors participated in the revision and approved the final manuscript.

Acknowledgments

This research is financially supported by the Guangxi Special Fund for Innovation-Driven Development (AA19254034) and Shenzhen Science and Technology Program (JSGG20220831105800002).

References

[1] M. Madaleno, E. Dogan, and D. Taskin, "A step forward on sustainability: the nexus of environmental responsibility, green technology, clean energy and green finance," *Energy Economics*, vol. 109, article 105945, 2022.

[2] B. Lin and Z. Li, "Towards world's low carbon development: the role of clean energy," *Applied Energy*, vol. 307, article 118160, 2022.

[3] S. Roga, S. Bardhan, Y. Kumar, and S. K. Dubey, "Recent technology and challenges of wind energy generation: a review," *Sustainable Energy Technologies and Assessments*, vol. 52, article 102239, 2022.

[4] M. Zhou and X. Li, "Influence of green finance and renewable energy resources over the sustainable development goal of clean energy in China," *Resources Policy*, vol. 78, article 102816, 2022.

[5] Y. Wang, Q. Yan, Y. Luo, and Q. Zhang, "Carbon abatement of electricity sector with renewable energy deployment: evidence from China," *Renewable Energy*, vol. 210, pp. 1–11, 2023.

[6] S. M. Masoudi and M. Baneshi, "Layout optimization of a wind farm considering grids of various resolutions, wake effect, and realistic wind speed and wind direction data: a techno-economic assessment," *Energy*, vol. 244, article 123188, Part B, 2022.

[7] R. Li, J. Zhang, and X. Zhao, "Dynamic wind farm wake modeling based on a bilateral convolutional neural network and high-fidelity LES data," *Energy*, vol. 258, article 124845, 2022.

[8] D. Feng, L. K. B. Li, V. Gupta, and M. Wan, "Componentwise influence of upstream turbulence on the far-wake dynamics of wind turbines," *Renewable Energy*, vol. 200, pp. 1081–1091, 2022.

[9] S. R. Reddy, "Wind farm layout optimization (windflo): an advanced framework for fast wind farm analysis and optimization," *Applied Energy*, vol. 269, article 115090, 2020.

[10] S. R. Moreno, J. Pierezan, L. dos Santos Coelho, and V. C. Mariani, "Multi-objective lightning search algorithm applied to wind farm layout optimization," *Energy*, vol. 216, article 119214, 2021.

[11] N. Guo, M. Zhang, B. Li, and C. Yu, "Influence of atmospheric stability on wind farm layout optimization based on an improved Gaussian wake model," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 211, article 104548, 2021.

[12] K. Chen, J. Lin, Y. Qiu, F. Liu, and Y. Song, "Joint optimization of wind farm layout considering optimal control," *Renewable Energy*, vol. 182, pp. 787–796, 2022.

[13] Y. Wang, Y. Yu, S. Cao, X. Zhang, and S. Gao, "A review of applications of artificial intelligent algorithms in wind farms," *Artificial Intelligence Review*, vol. 53, no. 5, pp. 3447–3500, 2020.

[14] H. Long, P. Li, and W. Gu, "A data-driven evolutionary algorithm for wind farm layout optimization," *Energy*, vol. 208, article 118310, 2020.

[15] M. Yeghikian, A. Ahmadi, R. Dashti et al., "Wind farm layout optimization with different hub heights in manjil wind farm using particle swarm optimization," *Applied Sciences*, vol. 11, no. 20, p. 9746, 2021.

[16] F. Bai, X. Ju, S. Wang, W. Zhou, and F. Liu, "Wind farm layout optimization using adaptive evolutionary algorithm with Monte Carlo tree search reinforcement learning," *Energy Conversion and Management*, vol. 252, article 115047, 2022.

[17] Y. Wen, M. Song, and J. Wang, "Wind farm layout optimization with uncertain wind condition," *Energy Conversion and Management*, vol. 256, article 115347, 2022.

[18] Z. Lei, S. Gao, Y. Wang, Y. Yu, and L. Guo, "An adaptive replacement strategy-incorporated particle swarm optimizer for wind farm layout optimization," *Energy Conversion and Management*, vol. 269, article 116174, 2022.

[19] R. M. Rizk-Allah and A. E. Hassanien, "A hybrid equilibrium algorithm and pattern search technique for wind farm layout optimization problem," *ISA Transactions*, vol. 132, pp. 402–418, 2023.

[20] Z. Lei, S. Gao, Z. Zhang, H. Yang, and H. Li, "A chaotic local search-based particle swarm optimizer for large-scale complex wind farm layout optimization," *IEEE/CAA Journal of Automatica Sinica*, vol. 10, no. 5, pp. 1168–1180, 2023.

[21] Y. Yu, T. Zhang, Z. Lei, Y. Wang, H. Yang, and S. Gao, "A chaotic local search-based lshade with enhanced memory storage mechanism for wind farm layout optimization," *Applied Soft Computing*, vol. 141, article 110306, 2023.

[22] H. Yang, S. Gao, Z. Lei, J. Li, Y. Yu, and Y. Wang, "An improved spherical evolution with enhanced exploration capabilities to address wind farm layout optimization problem," *Engineering Applications of Artificial Intelligence*, vol. 123, article 106198, 2023.

[23] L. Zhang, J. Li, X. Xu et al., "High spatial granularity residential heating load forecast based on dendrite net model," *Energy*, vol. 269, article 126787, 2023.

[24] Z. Yang, T. Hu, J. Zhu, W. Shang, Y. Guo, and A. Foley, "Hierarchical high-resolution load forecasting for electric vehicle charging: a deep learning approach," *IEEE Journal of Emerging and Selected Topics in Industrial Electronics*, vol. 4, no. 1, pp. 118–127, 2023.

[25] J. Xue and B. Shen, "A novel swarm intelligence optimization approach: sparrow search algorithm," *Systems Science & Control Engineering*, vol. 8, no. 1, pp. 22–34, 2020.

[26] M. A. Awadallah, M. A. Al-Betar, I. A. Doush, S. N. Makhadmeh, and G. Al-Naymat, "Recent versions and applications of sparrow search algorithm," *Archives of Computational Methods in Engineering*, vol. 30, pp. 2831–2858, 2023.

[27] Y.-H. He, Y. Luo, A.-H. Li, T.-F. Wang, Z. Cheng, and Y.-H. Peng, "Research on protection optimization of distribution network containing distributed power generation based on sparrow algorithm," *In Journal of Physics: Conference Series*, vol. 1820, article 012147, 2021.

[28] A. Fathy, T. M. Alanazi, H. Rezk, and D. Yousri, "Optimal energy management of micro-grid using sparrow search algorithm," *Energy Reports*, vol. 8, pp. 758–773, 2022.

[29] H. M. Li and Y. Zhang, "Study of transformer fault diagnosis based on sparrow optimization algorithm," in *In Proceedings of the 2020 1st International Conference on Control, Robotics and Intelligent System*, pp. 63–66, New York, 2020.

[30] P. Kathiroli, "An efficient clusterbased routing using sparrow search algorithm for heterogeneous nodes in wireless sensor networks," in *2021 International Conference on Communication information and Computing Technology (ICCICT)*, Mumbai, India, 2021.

[31] D. Xu, Y. Li, B. Zhang et al., "Upstream and downstream location of voltage sag source based on sparrow search algorithm and rbf neural network," in *2022 Power System and Green Energy Conference (PSGEC)*, pp. 760–765, Shanghai, China, August 2022.

[32] Z. Zhuang and T. Jin, "Capacity configuration and control strategy of ev charging station with integrated wind power and energy storage based on ssa," in *2021 IEEE 5th Conference on Energy Internet and Energy System Integration (EI2)*, pp. 4316–4322, Taiyuan, China, October 2021.

[33] P. Jeevanantham and R. Revathi, "Sparrow search algorithm based cluster head selection in wsn," *International Journal of Recent Trends in Computer Science and Applications*, vol. 1, no. 1, pp. 1–4, 2022.

[34] F. S. Gharehchopogh, M. Namazi, L. Ebrahimi, and B. Abdollahzadeh, "Advances in sparrow search algorithm: a comprehensive survey," *Archives of Computational Methods in Engineering*, vol. 30, no. 1, pp. 427–455, 2023.

[35] J. Yuan, Z. Zhao, Y. Liu et al., "Dmppt control of photovoltaic microgrid based on improved sparrow search algorithm," *IEEE Access*, vol. 9, pp. 16623–16629, 2021.

[36] P. Wang, Y. Zhang, and H. Yang, "Research on economic optimization of microgrid cluster based on chaos sparrow search algorithm," *Computational Intelligence and Neuroscience*, vol. 2021, Article ID 5556780, 18 pages, 2021.

[37] B. Li, H. Wang, X. Wang, M. Negnevitsky, and C. Li, "Tri-stage optimal scheduling for an islanded microgrid based on a quantum adaptive sparrow search algorithm," *Energy Conversion and Management*, vol. 261, article 115639, 2022.

[38] J. Hou, X. Wang, Y. Su, Y. Yang, and T. Gao, "Parameter identification of lithium battery model based on chaotic quantum sparrow search algorithm," *Applied Sciences*, vol. 12, no. 14, p. 7332, 2022.

[39] L.-L. Li, J.-L. Xiong, M.-L. Tseng, Z. Yan, and M. K. Lim, "Using multi-objective sparrow search algorithm to establish active distribution network dynamic reconfiguration integrated optimization," *Expert Systems with Applications*, vol. 193, article 116445, 2022.

[40] T. Gao, D. Niu, Z. Ji, and L. Sun, "Mid-term electricity demand forecasting using improved variational mode decomposition and extreme learning machine optimized by sparrow search algorithm," *Energy*, vol. 261, article 125328, 2022.

[41] Z. Wang, G. Sun, K. Zhou, and L. Zhu, "A parallel particle swarm optimization and enhanced sparrow search algorithm for unmanned aerial vehicle path planning," *Heliyon*, vol. 9, no. 4, article E14784, 2023.

[42] Z. Tasneem, A. Al Noman, S. K. Das et al., "An analytical review on the evaluation of wind resource and wind turbine for urban application: prospect and challenges," *Developments in the Built Environment*, vol. 4, article 100033, 2020.

[43] K. Yang, G. Kwak, K. Cho, and J. Huh, "Wind farm layout optimization for wake effect uniformity," *Energy*, vol. 183, pp. 983–995, 2019.

[44] Z. Liu, S. Fan, Y. Wang, and J. Peng, "Genetic-algorithm-based layout optimization of an offshore wind farm under real seabed terrain encountering an engineering cost model," *Energy Conversion and Management*, vol. 245, article 114610, 2021.

[45] N. Li, Y. Zhou, Q. Luo, and H. Huang, "Discrete complex-valued code pathfinder algorithm for wind farm layout optimization problem," *Energy Conversion and Management: X*, vol. 16, article 100307, 2022.

[46] J. Qiujun, Z. Haitao, Y. Qingshan, Z. Xuhong, H. Guoqing, and L. Yifeng, "Wind farm layout optimization based on grid-coordinate genetic algorithm," *Acta Energiae Solaris Sinica*, vol. 43, no. 8, pp. 266–272, 2022.

[47] F. Porté-Agel, M. Bastankhah, and S. Shamsoddin, "Wind-turbine and wind-farm flows: a review," *Boundary-Layer Meteorology*, vol. 174, no. 1, pp. 1–59, 2020.

[48] A. P. J. Stanley, J. King, and A. Ning, "Wind farm layout optimization with loads considerations," *Journal of Physics: Conference Series*, vol. 1452, article 012072, 2020.

[49] C. Yang, X. Liu, H. Zhou, Y. Ke, and J. See, "Towards accurate image stitching for dronebased wind turbine blade inspection," *Renewable Energy*, vol. 203, pp. 267–279, 2023.

[50] S. Rahnamayan, H. R. Tizhoosh, and M. M. A. Salama, "Opposition-based differential evolution," *IEEE Transactions on Evolutionary computation*, vol. 12, no. 1, pp. 64–79, 2008.

[51] Y. Xu, Z. Yang, X. Li, H. Kang, and X. Yang, "Dynamic opposite learning enhanced teaching–learning-based optimization," *Knowledge-Based Systems*, vol. 188, article 104966, 2020.

[52] L. Zhang, T. Hu, Z. Yang, D. Yang, and J. Zhang, "Elite and dynamic opposite learning enhanced sine cosine algorithm for application to plat-fin heat exchangers design problem," *Neural Computing and Applications*, vol. 35, no. 17, pp. 12401–12414, 2023.

[53] L. Zhang, T. Hu, L. Zhang et al., "A novel dynamic opposite learning enhanced jaya optimization method for high efficiency plate–fin heat exchanger design optimization," *Engineering Applications of Artificial Intelligence*, vol. 119, article 105778, 2023.

[54] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, pp. 1942–1948, Perth, WA, Australia, 1995.

[55] R. Venkata Rao, "Jaya: a simple and new optimization algorithm for solving constrained and unconstrained optimization problems," *International Journal of Industrial Engineering Computations*, vol. 7, no. 1, pp. 19–34, 2016.

[56] R. Storn and K. Price, "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.

[57] M. Clerc and J. Kennedy, "The particle swarm - explosion, stability, and convergence in a multidimensional complex space," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 1, pp. 58–73, 2002.

[58] A. K. Shukla, P. Singh, and M. Vardhan, "Neighbour teaching learning based optimization for global optimization problems," *Journal of Intelligent & Fuzzy Systems: Applications in Engineering and Technology*, vol. 34, no. 3, pp. 1583–1594, 2018.

[59] R. Kadambur and P. Kotecha, "Multi-level production planning in a petrochemical industry using elitist teaching–learning-basedoptimization," *Expert Systems with Applications*, vol. 42, no. 1, pp. 628–641, 2015.

[60] J. J. Liang, B. Y. Qu, and P. N. Suganthan, *Problem definitions and evaluation criteria for the cec 2014 special session and competition on single objective real-parameter numerical optimization*, vol. 635, no. 2, 2013, Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou China and Technical Report, Nanyang Technological University, Singapore, 2013.

[61] B. Rosner, R. J. Glynn, and M.-L. T. Lee, "Incorporation of clustering effects for the wilcoxon rank sum test: a large-sample approach," *Biometrics*, vol. 59, no. 4, pp. 1089–1098, 2003.