Journal of Network Intelligence
Tawain Ubiquitous Information

Volume **7**, Number **4**, November **2022**

Capacity Optimization of Hybrid Energy Storage System Based on Improved Golden Eagle Optimization

Zhong-Kai Zhang

School of Electronic, Electrical Engineering and Physics
Fujian University of Technology
Fuzhou, 350118, China
1241804613@qq.com

Pei-Qiang Li*

School of Electronic, Electrical Engineering and Physics
Fujian University of Technology
Fuzhou, 350118, China
596905210@qq.com

Jing-Jie Zeng

School of Electronic, Electrical Engineering and Physics
Fujian University of Technology
Fuzhou, 350118, China
384676637@qq.com

Xin-Ke Liu

School of Electronic, Electrical Engineering and Physics
Fujian University of Technology
Fuzhou, 350118, China
243393785@qq.com

*Corresponding author: Pei-Qiang Li

Received May 6, 2022, revised June 15, 2022, accepted August 15, 2022.

Abstract. *To improve the economy of wind-solar hybrid power generation and energy storage system and reduce its operating costs, this paper studies the capacity optimization configuration model of wind-solar hybrid power generation and energy storage system. On the basis of this model, an improved Golden Eagle optimization algorithm is introduced to realize the optimal configuration of hybrid energy storage capacity. This method first introduces the static model of the whole life cycle cost, using batteries and super capacitors as hybrid energy storage devices for wind-solar hybrid systems, taking the minimum life cycle cost of the energy storage device as the goal, and the operating indicators such as the power shortage rate of the system as its constraints, a capacity optimization configuration model of the hybrid energy storage system is established; Secondly, an improved Golden Eagle optimization algorithm is proposed, the improvement strategy consists of a personal example learning strategy, a decentralized foraging strategy, and a random perturbation strategy. personal example learning and random perturbation can enhance the search capability of GEO and prevent the algorithm from falling into local optimal solutions, disperse foraging strategy can enhance the convergence rate and optimization accuracy of GEO; Finally, the model simulation and solution are carried out in Matlab. The experimental results indicate that the IGEO can not only improve the convergence speed and accuracy of the original algorithm, but also further improve the working state of the energy storage system and reduce the whole life cycle cost of the energy storage system.*
Keywords: Golden Eagle Optimization, Personal example learning, Disperse foraging strategy , Random perturbation, Hybrid energy storage system

---

1. **Introduction.** With the concept of carbon neutrality and carbon peaking, Clean energy power generation has developed rapidly in late years. The State has established a large number of wind-solar complementary power generation systems in regions rich in wind and solar resources [1]. However, due to the uncertainty and volatility of clean energy, energy storage system is needed to suppress power fluctuation in wind - solar hybrid system. Commonly used energy storage devices include super capacitors and batteries. The energy of the battery is relatively high, and it can store electric energy for a long time, which can largely increase the energy adjustment range of the entire power generation system. However, the power density of the battery is low, the cycle life is short, and the environment is polluted to a certain extent, the instability and intermittent problems of wind and light will increase the cost of the energy storage part of the system; Super capacitors have high energy storage power density, recharging and discharging rate is very fast, and long service life, which are beneficial to suppressing short-term power fluctuations of the system. To optimize the charge and discharge state of energy storage, reduce the number of charge and discharge of energy storage, and prolong its use time, Batteries and super capacitors can be mixed to achieve complementary advantages and disadvantages of the two types of energy storage, which is called a hybrid energy storage system(HESS)[2][3][4][5] .

To improve the economy of energy storage systems, many researchers studied the economy of energy storage capacity allocation, through the optimization of the algorithm, a more reasonable energy storage capacity configuration is obtained. Due to the long computation time of traditional optimization algorithms and the tendency to fall into local optima, researchers gradually began to improve the meta-heuristic algorithm. The meta-heuristic algorithm is an algorithm proposed by simulating the interactive behavior of living things[6][7]. For instance, particle swarm optimization (PSO)[8][9] simulates the foraging behavior of birds. Grey Wolf Optimizer (GWO)[10] simulates grey Wolf hunting behavior. Sparrow search algorithm (SSA)[11] simulates sparrow predation and anti-predation behavior. Bats algorithm (BA)[12] is based on the echolocation behavior of bats. Flower pollination algorithm (FPA)[13] simulates the flower pollination process of plants. Ant colony optimization (ACO) [14][15] simulates the behavior of ants in finding paths during foraging. Whale Optimization Algorithm (WOA) [16] simulates the social behavior of humpback whales. Salps Swarm algorithm (SSA) [17][18] is an algorithm to simulate the chain foraging behavior of Spirulina. There are also some intelligent algorithms obtained by simulating the survival of the fittest in nature. such as Genetic Algorithm (GA) [19], evolutionary strategy (ES) [20], Differential Evolution (DE) [21], gene expression programming (GEP) [22]. There are also some intelligent algorithms from physical laws or chemical reactions, such as simulated annealing (SA) [23], charged system search (CSS) [24] and gravitational local search (GLSA) [25]. Due to its randomness, the algorithm can avoid local extremum and obtain the optimal solution. According to the no free lunch law (NFL) [26], it can be known that no algorithm can obtain all the optimal solutions. Therefore, Liang et al[27]. added damping constraint processing strategy to enhance the particle swarm optimization algorithm, reducing the possibility of particle swarm optimization falling into local optimum. Chen et al[28]. addressed the problem of parameter estimation for photovoltaic models by combining Cuckoo Search (CS) and Biogeography-Based Optimization (BBO).

To obtain the optimal configuration of energy storage capacity, Domestic and foreign scholars have proposed many optimization algorithms, such as the GEO. The Golden Eagle optimizer imitates the spiral hunting behavior of Golden Eagle. In the search-to-development process, a good transition is achieved by constantly regulation the numerical of attack factors and cruise coefficients. The GEO is resemble to other algorithms, the Golden Eagle optimizer also exists some shortcomings. The main reason is that in the search process, the GEO algorithm dose not achieve a good balance between the search and development stages. To improve the search efficiency of Golden Eagle optimizer and prevent algorithm from falling into local optimum, a Golden Eagle optimizer based on personal example learning, disperse foraging strategy and random perturbation is proposed. On the one hand, adding personal example learning and random perturbation strategies can enable individuals in the population to improve themselves by learning excellent individuals in the sample pool, thereby improving the search efficiency of GEO and the possibility of reducing the local extremum of the algorithm is reduced. On the other hand, the disperse foraging strategy enables some individuals to search for a promising area through an automatically adjusted parameter, so that the algorithm well balanced between development and search, thereby the convergence speed of the algorithm is enhanced. To verify the superiority of the optimization algorithm in this paper, the algorithm is compared under the CEC2013 test set. The experimental results suggest that the performance of the improved algorithm has been greatly improved. Secondly, taking the whole life cycle cost (WLCC) of the HESS as the optimization goal, establish a capacity optimization configuration model of wind-solar hybrid energy storage system, by improving the GEO, the most reasonable number and capacity configuration of batteries and super capacitors are obtained.

2. **Wind-solar hybrid hybrid energy storage model.** In the wind-solar hybrid power generation system, batteries and supercapacitors are mixed as energy storage devices. The main components are wind turbines, photovoltaic arrays, batteries, super capacitors, converters, loads, etc. Figure 1 is the system structure diagram.

2.1. **Full life cycle cost.** The whole life cycle cost, refers to the sum of all the expenses paid in the process of equipment planning, manufacturing, installation, use, maintenance, and disposal during the life cycle of the equipment [29]. In order to estimate the full life cycle of energy storage more reasonably and accurately, this paper establishes the

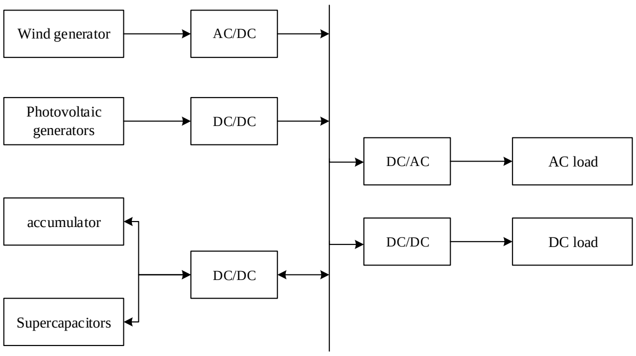


FIGURE 1. Hybrid energy storage structure of solar wind.

decomposition structure of WLCC. According to the operating cost of each stage of the whole life cycle, the whole life cycle cost is decomposed into basic modules, and a cost system arranged in sequence is formed, which is called the cost decomposition structure. According to the decomposition structure of the cost, an estimation model of the WLCC of the energy storage system is established, the cost model that does not consider the time value of money is a static cost model. The WLCC of the energy storage system is:

$$\mathrm{WLCC} = C_1 + C_0 + C_M + C_D \tag{1}$$

In the formula, WLCC is the cost of the whole life cycle; C_1 is the cost of investment equipment; C_0 is the operating cost of the equipment; C_M is the maintenance cost of the equipment; C_D is the disposal cost of the equipment.

2.2. **Objective function.** WLCC of hybrid energy storage system can be divided into four categories, The WLCC model of the HESS is:

$$\min C = C_1 + C_0 + C_M + C_D = (1 + f_{ob} + f_{mb} + f_{db})\, N_b P_b + (1 + f_{oc} f_{dc})\, N_c P_c \tag{2}$$

In the formula, N_b and N_c are the number of batteries and supercapacitors; P is the unit price of the battery super capacitor; f_{ob} and f_{oc} are the maintenance coefficients of batteries and super capacitors, In general, super capacitors are maintenance-free, the maintenance factor of super capacitors is 0; f_{db} and f_{dc} are the processing coefficients of batteries and super capacitors.

2.3. **Reliability index of power supply.** As the volatile operation index of wind-solar hybrid power generation index, The Loss of Power Supply Probability (LPSP) can be expressed by the ratio of power shortage to total power consumption.

$$f_{LPSP} = \sum_{k=1}^{K} E_{lps}(k) \Big/ \sum_{k=1}^{K} E_L(k) \tag{3}$$

In the formula, $E_W(k)$, $E_s(k)$, $E_L(k)$ are the power of wind energy, solar energy and load at time k, respectively; η_c is the power conversion efficiency of the inverter.

Figure 2 shows the calculation process of the LPSP rate. When the wind-solar power generation meets the load demand, the power shortage is 0, and the hybrid device is charged. When the wind-solar hybrid power generation is insufficient, the power shortage of the energy storage device discharge supplementary power is negative.

$$E_{lps} = E_L(k) - (E_W(k) + E_S(k))\, \eta_c \tag{4}$$

The rated capacity of the battery is E_{bn}, The minimum remaining storage power is E_{bmin}.

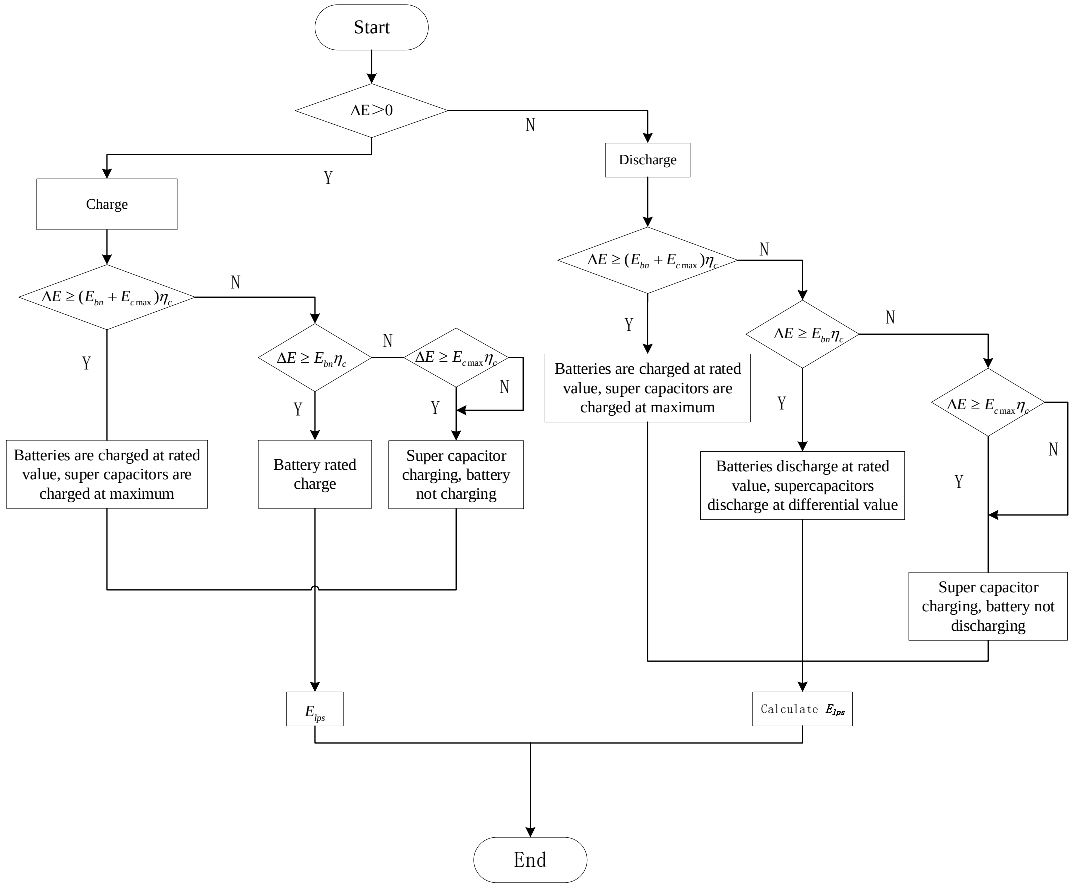


FIGURE 2. Hybrid energy storage structure of solar wind.

$$E_{bn} = N_B C_B U_B / 10^6 \tag{5}$$

$$E_{b\min} = N_B C_B U_B (1 - DOD)/10^6 \tag{6}$$

In the formula, U_b is the rated voltage of the battery; C_b is the rated capacity; DOD is the maximum depth of discharge.

In the actual running of the system, the voltage of the supercapacitor will work within a specified range. and the maximum stored power is:

$$E_{\mathrm{cmax}} = 0.5 \cdot N_c C_c U_{c\max}^2 / 3.6 * 10^9 \tag{7}$$

The minimum value is:

$$E_{cmin} = 0.5 \cdot N_c C_c U_{cmin}^2 / 3.6 * 10^9 \tag{8}$$

In the formula, U_c is the terminal voltage of the super capacitor; C_c is the capacitance value. It can be seen from Figure 2 that the battery is charged and discharged at the rated value, which can reduce the number of times of charging and discharging and the depth of discharge of the battery, and reduces the energy storage loss, increased battery life, and makes the energy storage more economical.

2.4. **Constraints.** (1) The reliability of power supply is characterized by the power shortage rate of the load, and the power shortage rate of the load needs to be within an acceptable range.

$$f_{LPSP} \leq f_{LPSP\max} \tag{9}$$

In the formula, $f_{LPSPmax}$ is the maximum power shortage rate allowed by the load.

(2) The upper and lower limits of the energy stored in the energy storage system:

$$E_{bmin} < E_b(k) < E_{bn} \tag{10}$$

$$E_{cmin} < E_c(k) < E_{cmax} \tag{11}$$

(3) ΔE can be divided into two parts, the basic part and the fluctuation part, the battery is mainly responsible for the basic part, which needs to meet:

$$E_b(k) \leq \mu \cdot \Delta E \tag{12}$$

3. **Golden Eagle Optimizer.** The meta-heuristic algorithm are more often than not partitioned into two stages: search and development. The algorithm expands the search range through the search stage, so that the population resulting from the search agent is as diverse as possible. The role of the development phase is to provide the algorithm with sufficient optimization depth so that the search agent can arrive at the optimal solution. The GEO can also be divided into two phases: search and development. GEO algorithm is an optimization algorithm that imitates the spiral hunting characteristics of golden eagle. Each golden eagle has the attraction of attacking prey and cruising to get better food.

3.1. **Prey selection.** Each iteration of the prey selection, Each golden eagle individual needs to select a target prey to perform cruising and attacking behaviors. In GEO, the target prey will be defined as the best solution that the current golden eagle group can find. Each golden eagle can remember the best solution found. In each iteration calculation, each search vector can select the target prey from the whole group memory. Then calculate the assaulting and patrolling vector of every golden eagle relative to the selected prey. Update memory if the calculated current position is better than the previous one. Prey selection strategy is very important in geographic information systems. When choosing a basic way to calculate, each golden eagle can only choose prey in his memory. However, in order to improve the search efficiency, it is stipulated that each golden eagle individual can randomly choose prey from the memory of other golden eagle groups. But to improve search efficiency, every prey in group memory is distributed one by one to the Golden Eagle. Then calculate the attack and cruise vector of each golden eagle to the selected prey.

3.2. **Aggressive behavior.** GEO's development phase is mainly guided by attack vectors, Equation (13) can calculate the attack vector:

$$A_i = X_p^* - X_i \tag{13}$$

In the formula, A_i is the attack vector of the i-th golden eagle. X_p^* is the current optimal position. X_i is the current position of the Golden Eagle.

3.3. **Cruise behavior.** Golden Eagle uses the cruise vector to complete the search process of the search space, and then find a more sufficient area for food. The tangent vector of the circle is cruise vector, vertical with the attack vector. Cruising can also be expressed as the linear velocity of the golden eagle relative to its prey. The n-dimensional patrol vector on a hyperplane tangent to the circle. Therefore, the equation of tangent hyperplane should be calculated first. Equation (14) is a scalar expression of an n-dimensional hyperplane.

$$\sum_{j=1}^{n} s_j x_j = d \tag{14}$$

Equation (15) is used to calculate the hyperplane where the cruise vector lies.

$$\sum_{j=1}^{n} b_j x_j = \sum_{j=1}^{n} b_j^t x_j^* \tag{15}$$

From the n variables, if the attack vector is 0, this element cannot be chose. GEO randomly picks one variable from the variables as a fixed variable and assigns the other free vectors to random values. Fixed variables calculated by equation (16).

$$C_y = \frac{d - \sum_{j,j\neq y} b_j}{b_y} \tag{16}$$

After the fixed variable of cruise hyperplane is determined, the points on the cruise hyperplane can be expressed by Equation (17)

$$C_i = \left(c_1 = \text{ random }, c_2 = \text{ random }, \ldots C_y = \frac{d - \sum_{j,j\neq y} b_j}{b_y}, \ldots c_n = \text{ random } \right) \tag{17}$$

3.4. **Location Update.** The movement of the golden eagle includes both attacking and cruising. Define Equation (18) as the iteration step size of Golden Eagle

$$\Delta x_i^t = r_1 p_a^t \frac{A_i}{\|A_i\|} + r_2 p_C^t \frac{C_i}{\|C_i\|} \tag{18}$$

$$p_a^t = p_a^0 + \frac{t}{T}\left|p_a^T - p_a^0\right| \tag{19}$$

$$p_c^t = p_c^0 + \frac{t}{T}\left|p_c^T - p_c^0\right| \tag{20}$$

In equations (19) and (20), t represents the current number of iterations, and T Represents the maximum number of iterations. Equation (19) represents the aggression vector coefficient, Equation (20) represents the cruise vector coefficient. These two coefficients control the effect of the attack vector and cruise vector on the step vector. Equation (19) $= [0.5, 2]$, Equation (20)$= [1, 0.5]$, Parameter reference [30]. $r_1 = [0, 1]$, $r_2 = [0, 2]$. $\|A_i\|$, $\|C_i\|$ is the Euclidean norm of the attack vector and the patrol vector.

$$\|A_i\| = \sqrt{\sum_{j=1}^{n} b_j^2} \tag{21}$$

$$\|C_i\| = \sqrt{\sum_{j=1}^{n} C_j^2} \tag{22}$$

The iterative position t is added to the iterative step vector to generate a new iterative individual position.

$$\|A_i\| = \sqrt{\sum_{j=1}^{n} b_j^2} \tag{23}$$

If the iterative position is better than the memory position, the memory position will be replaced by the new position, otherwise it will remain unchanged. Pseudo-code:

Algorithm 1 The framework of the GEO

Set the population size n, current number of iterations t, and the maximum number ofiterations T;
Initialize population individuals;
Calculate fitness function;
Initialize population memory;
Initialize p_a and p_c
$t = 1$
while $t < T$ **do**
  update p_a and p_c by Eqs.(19) and (20)
  for i **do**=1:n
    Randomly select a prey from the population's memory
    Calculate C by Eq.(16)
    Calculate ΔX by Eq.(18)
    Update *new* position $xnew_i$; by Eq.(23)
    Calculate fitness function of $xnew_i$;
    if f **then**itness of $xnew$; is better than the fitness of the position in eagle i's memory
      Update the memory of eagle i
    end if
  end for
  $t = t + 1$
end while

4. **Improved Golden Eagle Optimizer (IGEO).** The GEO algorithm unable well balance the relationship between the search stage and the development stage. To enhance the performance of GEO, this paper will improve the performance of the algorithm by introducing dividual example learning, dispersion foraging strategy and random perturbation strategy.

4.1. **Personal example learning.** The attack vector of search agent i represents the distance vector from the current location to the personal best location of search agent p in GEO. The personal best position of search agent p is chosen randomly from the memory of the population. If the best personal position of search agent p is poor, the attack vector A_i will develop in a poor direction. This larger randomness may increase the GEO computation time. In this paper, applying case learning to GEO can make search develop in a better direction, and it can improved search capability for GEO and prevent GEO from falling into local optimum. Firstly, the fitness of individuals in the population needs to be arranged in ascending order, and X_i can learn from its example pool. Figure 3 Sample pool for personal example learning. The individual $X_i's$ example pool is composed of the individual before X_i and himself. The m sample value is randomly selected from all the best positions of individuals, and the attack vector will be updated with this sample value. Change Equation (13) to Equation (26).

$$\overline{X_p^*} = X_p^*(m) \tag{24}$$

$$\mathrm{m} = ceil(\mathrm{i} * \text{ rand }) \tag{25}$$

$$A_i = \overline{X_p^*} - X_i \tag{26}$$

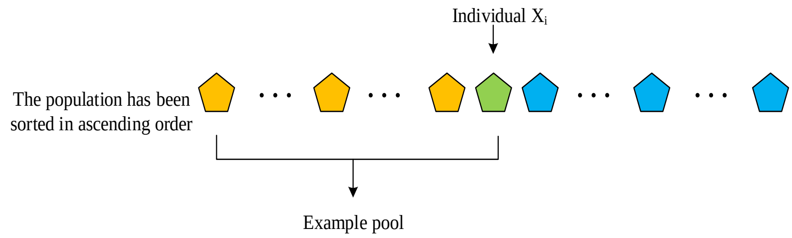


FIGURE 3. Example pool of personal example learning.

In the formula, $X_p^*(m)$is the sample value of the m-th individual in the sample pool consisting of individual best positions; *ceil* represents rounding; X_i represents the position of the ith individual.

4.2. **Dispersion foraging strategy.** Food shortage may genesis in areas where a group of golden eagles live. This event may have forced some golden eagle individuals to disperse into a new area to forage. This dispersal behavior facilitates the migration of golden eagles to food-rich areas and improves their survivability in harsh environments. The dispersed foraging strategy is proposed by mathematically modeling the dispersal behavior. During the dispersive foraging phase, partial search agents can be relocated to a more promising area based on dispersion (DR), and the updated location formula is as equation (27).

$$X_{i,d}^{t+1} = X_{i,d}^{t} + \rho \cdot \Delta_{i,d}^{t} \cdot B_{i,d}^{t} \tag{27}$$

$$\Delta_{i,d}^{t} = \left(X_{m,d}^{t} - X_{n,d}^{t}\right) \tag{28}$$

Where ρ denotes the scaling factor of the transference distance of the seek agent in the search stage, $\rho \sim \mathrm{N}\left(0.5, 0.1^2\right)$. For the install of ρ parameters, please refer to [31]. Δ^t expresses the random transference distance of the seek agent. B can be used to determine whether the seek agent is scattered and is a logical value, which can be expressed as:

$$B_{i,d}^{t} = \begin{cases} 1, & \text{rand } > DR \\ 0, & \text{otherwise} \end{cases} \tag{29}$$

Where DR represents the manner decreasing parameter in the iterative process, which can be expressed by equation (30):

$$\mathrm{DR} = \mathrm{D}R_{\max} - \left(\mathrm{D}R_{\max} - \mathrm{D}R_{\min}\right) \cdot t/L \tag{30}$$

It follows from equations (27) and (30) that the scaling factor ρ in the search agent improves the randomness throughout the optimization process. And through the adaptive adjustment of DR, the scale of search agents scattered to new areas can be controlled. In the initial stage of iteration, the small-scale golden eagle has a larger DR value in the scattered foraging stage, which can improve the convergence speed of the algorithm. Secondly, the DR value decreases with the iteration times increases, and the search agent is more likely to participate in the decentralized restle process, preventing the algorithm from obtaining a local optimal solution.

Pseudo-code:

4.3. **Random perturbation strategy.** In the iterative process of complex problems, the algorithm may fall into a local optimum. To make up for the deficiency of the algorithm, this paper adds a random disturbance, and continues to run the algorithm on this

Algorithm 2 Disperse foraging algorithm

1: **Input:** The search agent population X^t, parameter DR
2: **Output:** Updated search agents population X^{t+1}, parameter DR
3: Generate a logic matrix B^t by Eq.(29)
4: Generate the dispersed distance matrix by Δ^t Eq.(28)
5: Update the position of search agent by Eq.(27)

basis. The formula looks like this:

$$x^{i,\text{ iter }+1} = \begin{cases} x^{i,\text{ iter}} + \text{ rand }, \text{ if } r \geq 0.2 \\ x^{i,\text{ iter}} \times \text{Gaussian}(\mu, \delta), \text{ otherwise} \end{cases} \tag{31}$$

Where rand is randomly selected in [0,1] and r_j is the probability of choosing to perform a random perturbation. The Gaussian variation distribution function is as follows:

$$\text{Gaussian}(\mu, \delta) = (1/\sqrt{2\pi\delta}) \exp\left(-(x-\mu)^2/2\delta^2\right) \tag{32}$$

Where μ is the mean and δ^2 is the variance.

4.4. **Numerical experiments on benchmark functions.** To validate the optimal capability of the proposed algorithm in this paper, the performance of IGEO, GEO, PSO, WOA and GWO were compared under the CEC2013 test suite. There are three types of benchmark functions in the CEC2013 test suite, and four benchmark functions are selected for each type of benchmark function in this paper. The function expressions and related information are shown in Table 1. The dimension of the benchmark function is set to 50. The initial solution is in the range [-100, 100]. This paper carries out 30 independent tests on each algorithm to ensure the reliability of the experiment. The maximum number of values of functions (NFES) is 100,000. Take the fitness error $f = f_i - f^*$ as the objective function. If the f value is smaller, the optimization result of the algorithm is better. All experiments are carried out in the computing environment of MATLAB 2018b.

According to the results in Tables 2 and 3, the IGEO algorithm is outstrip to other algorithms. This shows that the improvement of the algorithm in this paper is effective. The main reason is that the individual sample learning and random perturbation strategy can improve the search ability of GEO by learning the excellent individuals in the sample pool and prevent GEO from falling into local optimum. The disperse foraging strategy enables some individuals to search a promising area through an automatically adjusted parameter, so that the algorithm can achieve a good balance between development and search, thereby improving the velocity of convergence of the algorithm.

It can be seen from the function convergence curve that the convergence accuracy of IGEO on F4 and F10 is not as good as that of PSO and GWO, but the accuracy of convergence optimization of IGEO is better compared with other algorithms on other benchmark functions. In terms of convergence rate, the convergence speed of IGEO algorithm has obvious advantages over GEO. However, other algorithms are easy to fall into local extremum in the iterative process, which leads to the rapid convergence of the algorithm, so the convergence rate on some benchmark functions is not as fast as miscellaneous arithmetic, but the final optimization accuracy is also better than other algorithms. This is due to the introduced personal example learning strategy increases the population multiplicity of the algorithm, while the disperse foraging strategy can enhance the convergence rate and optimization accuracy of GEO, and the random perturbation strategy can further recede the chance of the algorithm falling into the local extremum and increase the search performance of the algorithm.

Table 1. Benchmark functions

	No.	Function Name	Dim	Space	$f_{min} = f_i - f^*$
Unimodal Functions	1	Sphere Function	50	[−100,100]	0
	2	Rotated Bent Cigar Function	50	[−100,100]	0
	3	Rotated Discus Function	50	[−100,100]	0
	4	Different Power Function	50	[−100,100]	0
Basic Multimodal Functions	5	Rotated Schaffers F7 Function	50	[−100,100]	0
	6	Rastrigin's Function	50	[−100,100]	0
	7	Rotated Rastrigin's function	50	[−100,100]	0
	8	Lunacek Bi_Rastrigin Function	50	[−100,100]	0
Composition Functions	9	Composition Function 4 (n=3,Rotated)	50	[−100,100]	0
	10	Composition Function 5 (n=3,Rotated)	50	[−100,100]	0
	11	Composition Function 6 (n=5,Rotated)	50	[−100,100]	0
	12	Composition Function 6 (n=5,Rotated)	50	[−100,100]	0

Figure 4 is the convergence curve of the five algorithms. As can be seen from the convergence curve, IGEO has better convergence rate and accuracy. Combining with Tables 2 and 3, it can be seen that in the CEC2013 test function, IGEO performs better than other algorithms.

5. Application of improved algorithm in energy storage capacity optimization.

5.1. **System parameter settings.** Taking a wind-solar hybrid power generation system as an example, Figure 5 shows the annual power generation of the wind turbine, Figure 6 shows the solar power generation, and Figure 7 shows the annual power consumption of the load. The LPSPmax of the inverter is set to 0.95, and the power supply shortage rate of the system is 0.05. The parameters of batteries and super capacitors are shown in the Table 4.

5.2. **Simulation results and analysis of examples.** According to the previously determined objective function, constraints and various parameters, use IGEO, GEO, PSO, WOA, GWO algorithms to solve, simulate in matlab. The population size of the algorithm is 20, and the maximum number of iterations is 100. The change curve of the obtained optimal individual fitness value is shown in Figure 8. The full life cycle cost of the energy storage system after optimization is shown in Table 5 .

Table 2. Performance Comparison of Algorithms

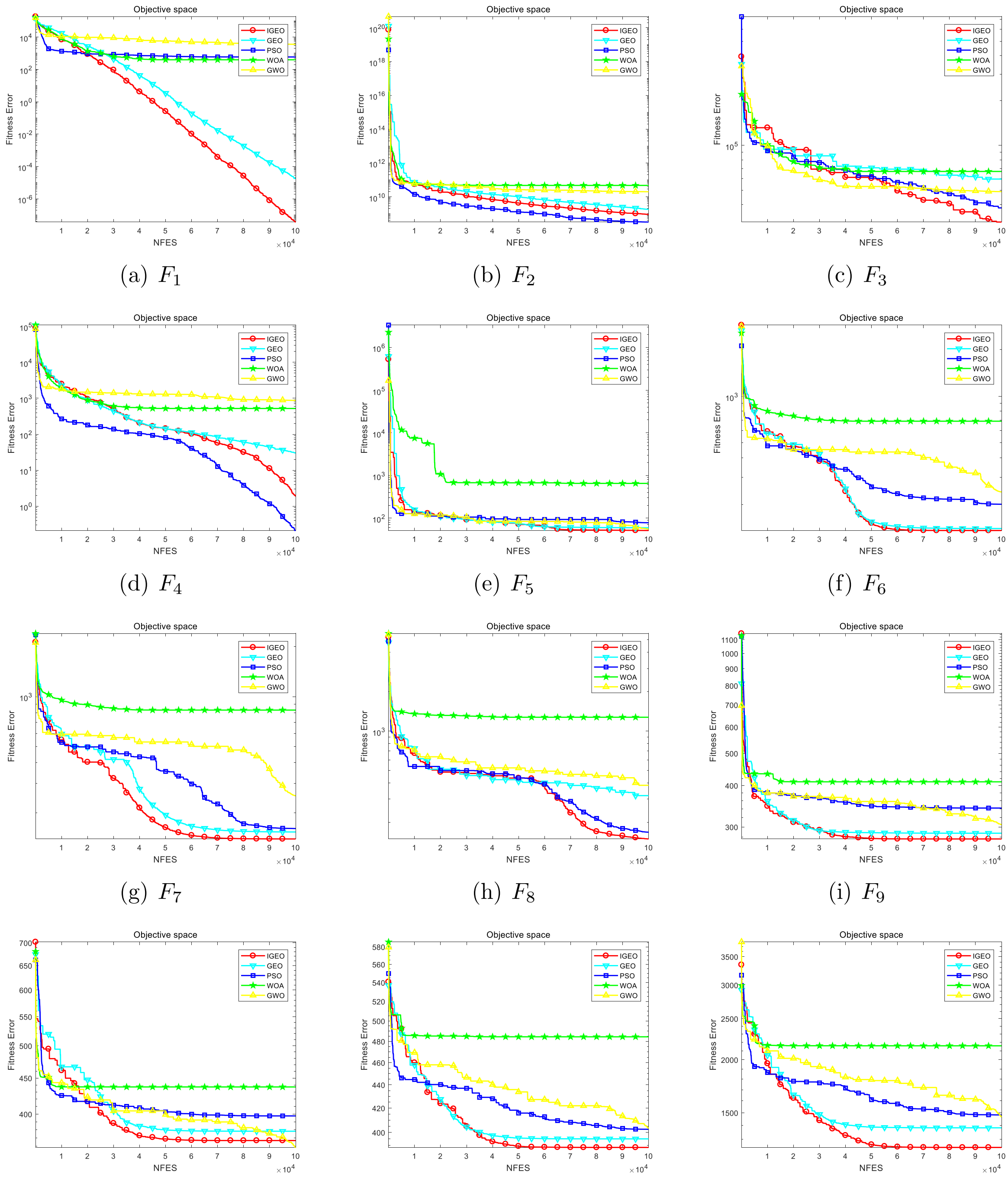
50D	GEO			PSO			IGEO		
	Best	Ave	Std	Best	Ave	Std	Best	Ave	Std
F1	8.46E-06	2.01E-05	7.03E-06	3.74E-04	1.07E+02	2.52E+02	**6.57E-09**	**3.08E-08**	**1.98E-08**
F2	4.64E+08	1.29E+09	6.26E+08	2.57E+08	3.27E+09	3.31E+09	**1.14E+08**	**8.91E+08**	**4.76E+08**
F3	3.92E+04	5.24E+04	6.37E+03	1.64E+04	2.73E+04	5.94E+03	**1.51E+04**	**2.45E+04**	**5.07E+03**
F4	2.38E+01	3.62E+01	6.80E+00	**7.66E-02**	6.10E+00	2.33E+01	7.60E-01	**5.95E+00**	**4.37E+00**
F5	4.49E+01	5.55E+01	6.08E+00	5.34E+01	9.69E+01	2.48E+01	**3.30E+01**	**4.59E+01**	**6.96E+00**
F6	1.05E+02	1.39E+02	1.73E+01	1.53E+02	2.36E+02	3.74E+01	**6.77E+01**	**1.13E+02**	**1.62E+01**
F7	9.42E+01	1.28E+02	**1.64E+01**	1.61E+02	2.59E+02	5.66E+01	**7.57E+01**	**1.11E+02**	2.81E+01
F8	1.52E+02	2.45E+02	7.26E+01	1.17E+02	1.71E+02	2.95E+01	**1.06E+02**	**1.47E+02**	**2.35E+01**
F9	2.73E+02	2.88E+02	**6.95E+00**	2.94E+02	3.25E+02	1.61E+01	**2.63E+02**	**2.79E+02**	1.13E+01
F10	3.49E+02	3.67E+02	**9.06E+00**	3.35E+02	3.84E+02	1.76E+01	**3.24E+02**	3.51E+02	1.36E+01
F11	2.02E+02	3.80E+02	4.93E+01	3.67E+02	4.08E+02	1.59E+01	**2.01E+02**	**3.32E+02**	8.07E+01
F12	1.08E+03	1.28E+03	**9.39E+01**	1.24E+03	1.51E+03	1.27E+02	**8.81E+02**	**1.19E+03**	1.09E+02
Win	12	12	8	11	12	11	-	-	-
Draw	0	0	0	0	0	0	-	-	-
Lose	0	0	4	1	0	1	-	-	-

Table 3. Performance Comparison of Algorithms

50D	WOA			GWO			IGEO		
	Best	Ave	Std	Best	Ave	Std	Best	Ave	Std
F1	1.23E+02	3.69E+02	1.58E+02	1.01E+03	3.00E+03	1.16E+03	**6.57E-09**	**3.08E-08**	**1.98E-08**
F2	2.40E+10	5.72E+10	1.83E+10	6.94E+09	1.56E+10	5.65E+09	**1.14E+08**	**8.91E+08**	**4.76E+08**
F3	5.37E+04	7.57E+04	1.37E+04	3.09E+04	5.16E+04	9.96E+03	**1.51E+04**	**2.45E+04**	**5.07E+03**
F4	2.83E+02	4.86E+02	1.28E+02	3.80E+02	9.15E+02	3.44E+02	7.60E-01	**5.95E+00**	**4.37E+00**
F5	1.76E+02	8.10E+02	9.84E+02	3.69E+01	6.69E+01	1.57E+01	**3.30E+01**	**4.59E+01**	**6.96E+00**
F6	6.28E+02	7.91E+02	1.02E+02	1.28E+02	2.16E+02	5.13E+01	**6.77E+01**	**1.13E+02**	**2.76E+01**
F7	6.75E+02	9.20E+02	1.17E+02	1.76E+02	2.81E+02	8.98E+01	**7.57E+01**	**1.11E+02**	2.81E+01
F8	1.00E+03	1.18E+03	8.68E+01	2.06E+02	3.43E+02	8.43E+01	**1.06E+02**	**1.47E+02**	**2.35E+01**
F9	3.94E+02	4.17E+02	1.42E+01	2.88E+02	3.06E+02	1.14E+01	**2.63E+02**	**2.79E+02**	1.13E+01
F10	4.09E+02	4.34E+02	1.42E+01	3.28E+02	**3.48E+02**	1.17E+01	**3.24E+02**	3.51E+02	1.36E+01
F11	2.05E+02	4.78E+02	5.22E+01	3.75E+02	4.01E+02	**9.42E+00**	**2.01E+02**	**3.32E+02**	8.07E+01
F12	2.03E+03	2.26E+03	1.12E+02	1.12E+03	1.35E+03	1.11E+02	**8.81E+02**	**1.19E+03**	1.09E+02
Win	12	12	12	12	11	10	-	-	-
Draw	0	0	0	0	0	0	-	-	-
Lose	0	0	0	0	1	2	-	-	-

Table 5. Energy storage parameters

Optimization parameters	WOA	GWO	PSO	GEO	IGEO
battery/pc	31474	38835	43975	49060,	49060,
Super capacitor/pc	5758741	5670286	5607062	5545390	5545369
LPSP	0.0365	0.0365	0.0365	0.0365	0.0365
cost /yuan	157064	156911	156766	156646	156545



(a) F_1 (b) F_2 (c) F_3

(d) F_4 (e) F_5 (f) F_6

(g) F_7 (h) F_8 (i) F_9

(j) F_{10} (k) F_{11} (l) F_{12}

Figure 4. Convergence curve comparison of all algorithms.

It can be seen from Figure 8 that the IGEO algorithm has the fastest convergence speed compared with other algorithms. According to Table 5, when the IGEO algorithm is used, 49,060 batteries and 5,545,369 super capacitors are required, the full life cycle cost at this time is 156,545 yuan, and the load power shortage rate is 0.0365. Compared with other algorithms, the IGEO algorithm has the lowest life-cycle cost and the lowest number of super capacitors.

6. **Conclusions.** This paper proposes an improved strategy of the Golden Eagle optimizer. Improved strategies include individual example learning, decentralized foraging

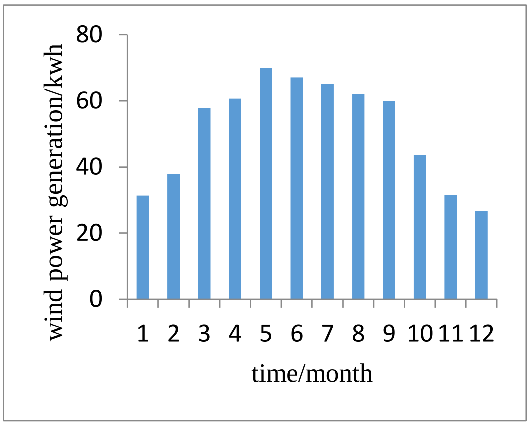


Figure 5. Wind power generation.

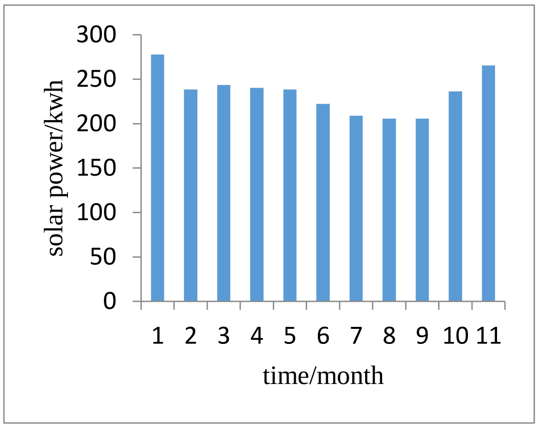


Figure 6. Solar power generation.

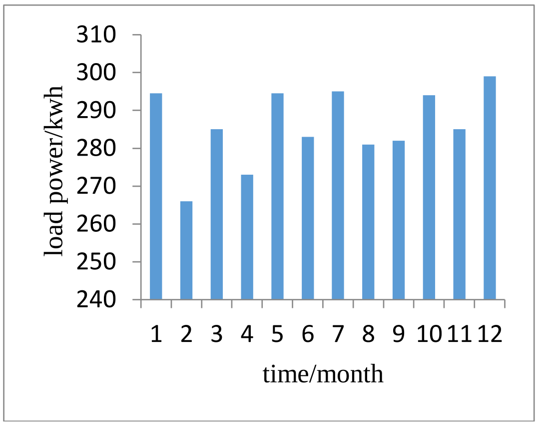


Figure 7. Load consumption.

strategies, and random perturbations. Personal example learning can increase the search acapabilities of GEO and decrease the likelihood that GEO will fall into local optimality. The decentralized foraging strategy can enhance the optimization precision of GEO and make GEO have faster convergence speed. The improved algorithm will be tested under the CEC2013 test suite. The simulation results show that for most benchmark functions, this algorithm is superior to other algorithms in terms of convergence speed and accuracy.

TABLE 4. Energy storage parameters

	battery	capacitor
Rated voltage	12	2.7
Rated Capacity	100	–
charging power	0.75	0.98
Discharge efficiency	0.85	0.98
depth of discharge	0.4	–
Operating parameters	0.1	0.01
maintenance factor	0.02	–
Cycle life/time	1500	500000
Unit price/yuan	400	350
Processing factor	0.08	0.04

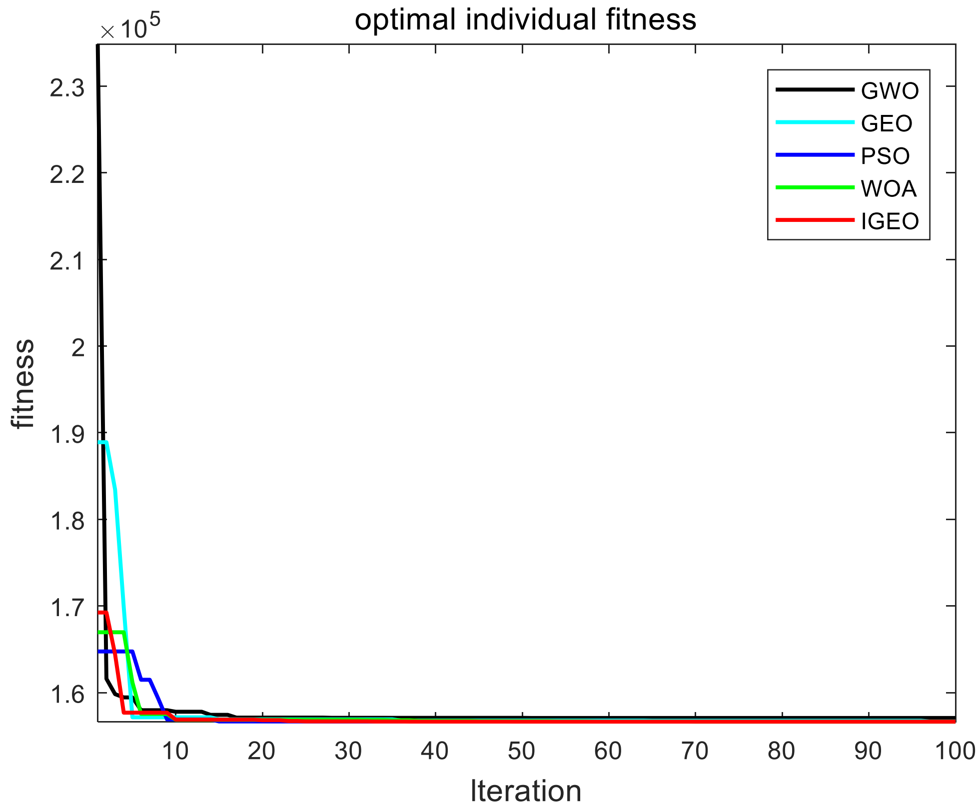


FIGURE 8. Load consumption.

Moreover, IGEO has high population diversity, which makes the algorithm abtain a good balance in the search and development process.

Apply IGEO to wind-solar hybrid hybrid energy storage model. Taking the whole life cycle cost of the system as the minimum goal and operating indicators such as power shortage rate as the corresponding constraints, the capacity allocation of energy storage is optimized. The simulation results indicate that the proposed IGEO algorithm can obtain the optimal capacity allocation of energy storage, the system has the least life cycle cost, and the convergence rate of the IGEO algorithm is better than that of the GEO, PSO, GWO, and WOA algorithms.

REFERENCES

[1] M.-T. Hagh, A. Lafzi, and A.-R. Milani, “Dynamic and stability improvement of a wind farm connected to grid using UPFC,” in *2008 IEEE International Conference on Industrial Technology.* IEEE, 2008, pp. 1-5.

[2] L. Xu, X.-B. Ruan, C.-X. Mao, B.-H. Zhang, and Y. Luo, “An improved optimal sizing method for wind-solar-battery hybrid power system,” *IEEE Transactions on Sustainable Energy*, vol. 4, no. 3, pp. 774–785, 2013.

[3] S. Golshannavaz, F. Aminifar, and D. Nazarpour, “Application of UPFC to enhancing oscillatory response of series-compensated wind farm integrations,” *IEEE Transactions on Smart Grid*, vol. 5, no. 4, pp. 1961–1968, 2014.

[4] W. Gu, and X.-Y. Li, and W. Wei, “Simulation study on wind farm stability with UPFC,” *Power System Protection and Control*, vol. 38, no. 11, 2010.

[5] C.-M. Chen, L.-L. Chen, Y.-Y. Huang, S. Kumar, and J.-M. Wu, “Lightweight authentication protocol in edge-based smart grid environment,” *EURASIP Journal on Wireless Communications and Networking*, vol. 2021, no. 1, pp. 1–18, 2021.

[6] P.-C. Song, S.-C. Chu, J.-S. Pan, and T.-Y. Wu, “An adaptive stochastic central force optimisation algorithm for node localisation in wireless sensor networks,” *Inderscience Publishers (IEL)*, vol. 39, no. 1-2, pp. 1–19, 2022.

[7] L.-L. Kang, R.-S. Chen, Y.-C. Chen, C.-C. Wang, X.-G. Li, and T.-Y. Wu, “Using cache optimization method to reduce network traffic in communication systems based on cloud computing,” *IEEE Access*, vol. 7, pp. 124397–124409, 2019.

[8] L.-L. Kang, R.-S. Chen, N.-X. Xiong, Y.-C. Chen, Y.-X. Hu, and C.-M. Chen, “Selecting hyper-parameters of Gaussian process regression based on non-inertial particle swarm optimization in Internet of Things,” *ACM Transactions on Internet Technology (TOIT)*, vol. 21, no. 2, pp. 1–26, 2021.

[9] J. Kennedy, and R. Eberhart, “Particle swarm optimization,” in *Proceedings of ICNN’95-international Conference on Neural Networks.* IEEE, 1995, pp. 1942–1948.

[10] K. Hussain, M.-N.-M. Salleh, S. Cheng, and Y.-H. Shi, “On the exploration and exploitation in popular swarm-based metaheuristic algorithms,” *Neural Computing and Applications*, vol. 31, no. 11, pp. 7665–7683, 2019.

[11] J.-K. 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.

[12] T.-T. Nguyen, J.-S. Pan, and T.-K. Dao, “A compact bat algorithm for unequal clustering in wireless sensor networks,” *Applied Sciences*, vol. 9, no. 10, pp. 1973, 2019.

[13] J.-S. Pan, T.-K. Dao, T.-S. Pan, T.-T. Nguyen, S.-C. Chu, and J.-F. Roddick, “An Improvement of Flower Pollination Algorithm for Node Localization Optimization in WSN,” *Journal of Information Hiding and Multimedia Signal Processing*, vol. 8, no. 2, pp. 486–499, 2017.

[14] J.-M.-T. Wu, G. Srivastava, J.-C.-W. Lin, and Q. Teng, “A Multi-Threshold Ant Colony System-based Sanitization Model in Shared Medical Environments,” *IEEE Access*, vol. 7, pp. 59504–59513, 2019.

[15] M. Dorigo, M. Birattari, and T. Stutzle, “Ant colony optimization,” *IEEE Computational Intelligence Magazine*, vol. 1, no. 4, pp. 28–39, 2006.

[16] S. Mirjalili, and A. Lewis, “The whale optimization algorithm,” *Advances in Engineering Software*, vol. 95, pp. 51–67, 2016.

[17] J.-S. Pan, J. Shan, S.-G. Zheng, S.-C. Chu, and C.-K. Chang, “Wind power prediction based on neural network with optimization of adaptive multi-group salp swarm algorithm,” *Cluster Computing*, vol. 24, no. 3, pp. 2083–2098, 2021.

[18] F.-Q. Zhang, T.-Y. Wu, Y.-O. Wang, R. Xiong, G.-Y. Ding, P. Mei, and L.-Y. Liu, “Application of quantum genetic optimization of LVQ neural network in smart city traffic network prediction,” *IEEE Access*, vol. 8, pp. 104555–104564, 2020.

[19] J.-H. Holland, “Genetic algorithms,” *Scientific American*, vol. 267, no. 1, pp. 66–73, 1992.

[20] Z.-Y. Meng, and J.-S. Pan, “QUasi-Affine TRansformation Evolution with External ARchive (QUATRE-EAR): an enhanced structure for differential evolution,” *Knowledge-Based Systems*, vol. 155, pp. 35–53, 2018.

[21] 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.

[22] Y.-Y. Zhang, "Backtracking search algorithm with specular reflection learning for global optimization," *Knowledge-Based Systems*, vol. 212, 106546, 2021.

[23] S. Mirjalili, "SCA: a sine cosine algorithm for solving optimization problems," *Knowledge-based Systems*, vol. 96, pp. 120–133, 2016.

[24] L. Abualigah, A. Diabat, S. Mirjalili, A.-E. Mohamed, and A.-H. Gandomi, "The arithmetic optimization algorithm," *Computer methods in Applied Mechanics and Engineering*, vol. 376, 113609, 2021.

[25] X.-P. Wang, J.-S. Pan, and S.-C. Chu, "A parallel multi-verse optimizer for application in multilevel image segmentation," *IEEE Access*, vol. 8, pp. 32018–32030, 2020.

[26] D.-H. Wolpert, and W.-G. Macready, "No free lunch theorems for optimization," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 67–82, 1997.

[27] J. Liang, S.-L. Ge, B.-Y. Qu, K.-J. Yu, F.-J. Liu, H.-T. Yang, P.-P. Wei, and Z.-M. Li, "Classified perturbation mutation based particle swarm optimization algorithm for parameters extraction of photovoltaic models," *Energy Conversion and Management*, vol. 203, 112138, 2020.

[28] X. Chen, and K.-J. Yu, "Hybridizing cuckoo search algorithm with biogeography-based optimization for estimating photovoltaic model parameters," *Solar Energy*, vol. 180, pp. 192–206, 2019.

[29] M.-T. Hagh, A. Lafzi, and A.-R. Milani, "Dynamic and stability improvement of a wind farm connected to grid using UPFC," in *2008 IEEE International Conference on Industrial Technology.* IEEE, 2008, pp. 1–5.

[30] M.-B. Abdolkarim, M.-D. Nayeri, A. Azar, T.-Y. Mohammadreza, "Golden eagle optimizer: A nature-inspired metaheuristic algorithm," *Computers & Industrial Engineering*, vol. 152, 107050, 2021.

[31] M. Zhao, J.-S. Pan, and S.-T. Chen, "Entropy-based audio watermarking via the point of view on the compact particle swarm optimization," *Internet Technol 2015*, vol. 16, no. 3, pp. 485–493, 2015.