electronics

Article

Research on Microgrid Optimal Scheduling Based on an Improved Honey Badger Algorithm

Zheng Wang, Zhenhai Dou *, Yuchen Liu, Jiaming Guo, Jingwei Zhao and Wenliang Yin

School of Electrical and Electronic Engineering, Shandong University of Technology, Zibo 255000, China; 22504040047@stumail.sdut.edu.cn (Z.W.); 22504040039@stumail.sdut.edu.cn (Y.L.); 22504040025@stumail.sdut.edu.cn (J.G.); 22804050008@stumail.sdut.edu.cn (J.Z.); yinwenliang@sdut.edu.cn (W.Y.)
* Correspondence: douzhenhai@sdut.edu.cn

Abstract: As global energy demands continue to grow and environmental protection pressures increase, microgrids have garnered widespread attention due to their ability to effectively integrate distributed energy sources, improve energy utilization efficiency, and enhance grid stability. Due to the complexity of internal structure, variety of energy sources, and uncertainty of load demand, the optimal scheduling problem of microgrids becomes extremely complicated. Traditional optimization methods often perform poorly in complex and dynamic microgrid environments, and it is assumed that the complexity is low or that more simplification is needed, which leads to poor convergence and local optimality when dealing with uncertainty and nonlinear problems, making intelligent optimization algorithms a crucial solution to this problem. To address the shortcomings of the traditional honey badger algorithm, such as the slow convergence speed and a tendency to fall into local optima in complex microgrid optimal scheduling problems, this paper proposes a multi-strategy improved honey badger algorithm. During the population initialization phase, a combined opposition-based learning strategy is introduced to enhance the algorithm's exploration and exploitation capabilities. Additionally, the introduction of variable spiral factors and a linearly decreasing strategy for parameters improves the overall efficiency of the algorithm and reduces the risk of local optima. To further enhance population diversity, a hunger search strategy is employed, providing stronger adaptability and global search capabilities in varying environments. The improved honey badger algorithm is then applied to solve the multi-objective optimal scheduling problem in grid-connected microgrid modes. The simulation results indicate that the improved honey badger algorithm effectively enhances the economic and environmental benefits of microgrid operations, improving system operational stability.

Keywords: microgrid; optimization scheduling; honey badger algorithm; linearly decreasing strategy; combined opposition-based learning; hunger search strategy; spiral factors

Citation: Wang, Z.; Dou, Z.; Liu, Y.; Guo, J.; Zhao, J.; Yin, W. Research on Microgrid Optimal Scheduling Based on an Improved Honey Badger Algorithm. *Electronics* **2024**, *13*, 4491. https://doi.org/10.3390/electronics13224491

Academic Editor: Shoji Nishikata

Received: 9 October 2024
Revised: 5 November 2024
Accepted: 14 November 2024
Published: 15 November 2024

1. Introduction

Currently, China's electric power industry is rapidly developing, aiming to enhance energy efficiency, minimize operational costs, and achieve environmental protection goals. Lowering electricity costs for users and accelerating the development of comprehensive energy systems, as well as improving the economic operation of microgrid systems, are current research hotspots. Therefore, optimizing the scheduling of microgrids is of significant importance [1–3].

In recent years, with the large-scale introduction of renewable energy sources and the continuous growth of electricity consumption demand, the complexity of energy networks has significantly increased. As the core of the energy system, the economic dispatch problem (EDP) plays a crucial role in ensuring the efficient and stable operation of the power system [4]. The goal of EDP is to allocate the output power of generators rationally to meet the load demand with the lowest production cost while adhering to various operational

constraints. This problem not only affects the pricing mechanism of the power market but also plays an important role in promoting the effective integration of new energy sources and improving energy utilization efficiency [5]. Therefore, researching how to efficiently solve the economic dispatch problem in complex energy networks is crucial for achieving efficient, reliable, and sustainable energy supply. The core objective of the economic dispatching problem is to optimize the distribution of power generation resources according to the demand of the power load so as to minimize the cost of power generation [6]. In an environment with multiple generators, varying fuel prices, emission limits, and complex load fluctuations meeting electricity demand while minimizing operating costs and carbon emissions is a highly challenging task.

Solving the economic dispatching problem is not only important for power companies to optimize resource allocation and improve operational efficiency, but it also has a profound impact on the security of the energy supply and environmental sustainability of the whole society [7]. Therefore, in-depth research on how to efficiently solve the economic dispatch problem in the dynamic energy network will not only help to minimize the cost of the power system but also effectively promote the application of green energy and promote energy transformation on a global scale.

For tackling the nonlinear problem of optimizing microgrid scheduling, most scholars establish objective functions and employ intelligent optimization algorithms to solve them under various constraints [8]. Traditional optimization algorithms struggle to achieve the desired results, whereas intelligent optimization algorithms offer effective solutions [9]. Swarm intelligence optimization algorithms involve iteratively sharing information to conduct subsequent searches, differing mainly in their global and local search capabilities [10–12]. The honey badger algorithm (HBA) [13] has been proven to be advantageous due to its simplicity, ease of implementation, and fast convergence speed compared to algorithms like the whale optimization algorithm (WOA) [14], particle swarm optimization (PSO) [15], and grey wolf optimization (GWO) [16]. However, it faces challenges such as slow convergence and susceptibility to local optima in solving high-dimensional complex problems. Addressing these issues involve optimizing the algorithm with additional strategies.

For instance, Yu et al. [17] proposed an improved IJAYA optimization algorithm that incorporates adaptive weighting to avoid local optima and ensures faster convergence in different search stages. Learning strategies to maintain population diversity and enhance search capabilities. Deng et al. [18] introduced multi-population strategies, collaborative evolution mechanisms, and information pheromone update strategies to balance the convergence speed and solution diversity, thereby improving the overall efficiency. Ahmed et al. [19] proposed an improved tabu search algorithm, improving the initialization stage and using a new reinforcement mechanism, ultimately improving the algorithm's computational efficiency. Deng et al. [20] applied chaotic mapping to the initialization stage, then used an antagonistic learning strategy to update the original pattern, improving the convergence speed of the whale optimization algorithm (WOA). They also proposed a new adaptive coefficient, leading to significant improvements in all aspects of the improved WOA. Shishavan et al. [21] proposed an improved cuckoo search optimization (CSO) algorithm, combining genetic algorithm (GA) for dynamic adjustment, improving the speed and accuracy of CSO. Yıldız et al. [22] proposed a hybrid method to improve the arithmetic optimization algorithm (AOA), enhancing its stability and quality and making it robust when dealing with complex real-world problems. Wu et al. [23] proposed an improved adaptive ant colony optimization algorithm, introducing a new heuristic mechanism and a new state transition probability rule to improve the algorithm's convergence speed and practicality. Wei et al. [24] proposed an improved sine–cosine algorithm, introducing the Haleton sequence in the initialization stage and a hybrid mutation strategy to improve the algorithm's convergence speed and accuracy. He [25] proposed an improved chaotic sparrow algorithm, introducing a nonlinear dynamic weight silver and optimizing the sparrow algorithm with the sine–cosine algorithm, improving the algorithm's performance

and convergence accuracy. Jia et al. [26] proposed a guided learning strategy (GLS) that can improve the performance of most algorithms.

In the improvement of the honey badger algorithm, Hu et al. [27] introduced the Bernoulli displacement mapping strategy and horizontal crossover strategy into the honey badger algorithm, enhancing the population diversity and improving the overall performance of the original algorithm. Lei et al. [28] introduced a spiral exploration mechanism and a density update silver based on the cosine law, improving the convergence speed of the algorithm and enabling the algorithm to jump out of local optimal values. Nassef et al. [29] introduced a strategy called dimensional learning hunting (DLH) into the honey badger algorithm, greatly improving the effectiveness and stability of the honey badger algorithm in solving global optimization problems. Han et al. [30] applied logical mapping to the initialization stage of the honey badger algorithm, obtaining a more reasonable population distribution. Deng et al. [31] proposed combining the Lévy flight strategy with HBA, improving the algorithm's optimization ability for complex engineering problems but lacking stability in dealing with complex environments. Dong et al. [32] introduced the cross-mutation principle into the exploration stage of the algorithm, obtaining a larger search range and faster convergence speed. Chai et al. [33] introduced a dual-population optimization mechanism that combines the slime mold algorithm and the honey badger algorithm, thereby improving the search efficiency and optimization performance of the entire algorithm. Xiang et al. [34] designed a restricted reverse learning mechanism and introduced an adaptive weight factor, accelerating the convergence speed of the algorithm. Compared with the original honey badger algorithm, the above-improved algorithms have made some progress in certain aspects, but there are still shortcomings: some improved algorithms only use a single optimization strategy, only improving the initialization stage of the algorithm, and not making much improvement to the exploration stage. The existing optimization strategies do not delve into the reasons for falling into local optima, only mentioning surface problems, leading to no improvement in the ability to escape local optima.

The research content of this paper is as follows:

1. Establish a Microgrid Model: An optimization model of the microgrid is established based on the total operating cost.
2. Propose a Multi-Strategy Improved Honey Badger Algorithm (MIHBA): Update of Dynamic Density Factor in Exploration Phase: The dynamic density factor is updated in the exploration phase to facilitate the smooth transition of the search phase. Introduce the Spiral Factor: A spiral factor is introduced to enhance the searching ability of the algorithm. Introduce a Hunger Search Strategy: A hunger search strategy is introduced to explore the solution space in a larger range and reduce the possibility of falling into local optima.
3. Benchmark Testing: Simulation and comparison tests are conducted using several benchmark functions to evaluate the performance of the algorithms.
4. Solve the Microgrid Optimization Problem: The improved honey badger algorithm is used to solve the optimization scheduling model of the microgrid, verifying the effectiveness and superiority of the algorithm in solving the optimization scheduling problem.

2. Microgrid Optimal Scheduling Model

The microgrid model established in this paper mainly includes photovoltaic cells (PV), wind turbines (WT), fuel cells (FC), microturbines (MT), and storage batteries (BT).

2.1. Photovoltaic Power Generation Model

Photovoltaic power generation utilizes solar photovoltaic cells to convert solar energy into electrical energy. This process involves the photovoltaic effect, where sunlight irradiates the photovoltaic panels, generating electricity. In this study, the output power of the

solar photovoltaic model is represented by solar irradiance. The mathematical model is as follows [35]:

$$P_{PV}(t) = \delta_{PV}(t) \times \frac{P_{PV,st}(1 + k_{PV}(T_C - T_R))M_t}{M_{st}}. \quad (1)$$

In Equation (1), $P_{PV}(t)$ is the output power of the photovoltaic model at time period t, $P_{PV,st}$ is the maximum output power of the photovoltaic cells under standard test conditions, k_{PV} is the power coefficient of the photovoltaic panels, T_C is the surface temperature of the cells at time period t, T_R is the surface temperature of the cells under standard conditions, M_t is the solar irradiance at time t, and M_{st} is the solar irradiance under standard conditions. $\delta_{PV}(t)$ is the binary decision variable of a photovoltaic generator at time t.

2.2. Wind Power Generation System

The wind turbine (WT) is a device that converts wind energy into electrical energy. The output power of a wind turbine is related to its rated capacity and wind speed [36]. The mathematical model is given by [37]:

$$P_{WT}(t) = \begin{cases} 0, & 0 \le v(t) < v_{\text{in}}, \\ \delta_{WT}(t)(v^3(t) + bv^2(t) + cv(t) + d), & v_{\text{in}} \le v(t) < v_N, \\ \delta_{WT}(t)P_N, & v_N \le v(t) < v_{\text{out}}, \\ 0, & v(t) \ge v_{\text{out}}. \end{cases} \quad (2)$$

In Equation (2), $P_{WT}(t)$ represents the output power of the wind turbine at time t, P_N is the rated power of the generator, $v(t)$ is the actual measured wind speed at time t, v_{in} is the cut-in wind speed, v_{out} is the cut-out wind speed, and v_N is the rated wind speed. In addition, a, b, c, and d are specific coefficients and are related to the fan model. $\delta_{WT}(t)$ is the binary decision variable of the wind power plant at time t.

2.3. Micro Gas Turbine Model

A micro gas turbine (MT) is a type of thermal power generator that converts mechanical energy into electrical energy by injecting consumed natural gas. The generated waste heat can be recovered through a recovery device, providing thermal energy to heat or cool loads to achieve efficient energy utilization. The mathematical models for electrical efficiency and thermal efficiency are as follows:

$$P_{MT}(t) = \delta_{WT}(t)P_g\eta_E, \quad (3)$$

$$V_{MT}(t) = \frac{P_g}{L_g}\Delta T. \quad (4)$$

In Equations (3) and (4), P_{MT} is the output electrical power of the micro gas turbine, η_E is the generating efficiency, P_g is the power consumption of natural gas, $V_{MT}(t)$ is natural gas consumption, L_g is the lower heating value of natural gas, typically taken as $9.78(\text{kW} \cdot \text{h})/\text{m}^3$, and ΔT is the unit running time [38]. $\delta_{WT}(t)$ is the binary decision variable of the micro gas turbine at time t.

2.4. Fuel Cell Model

The fuel cell (FC) is a device that converts the chemical energy in natural gas into electrical energy. Fuel cells have minimal environmental pollution during power generation and relatively high energy conversion rates. The mathematical model for the consumption of natural gas and the output electrical power is as follows [39]:

$$P_{FC}(t) = \delta_{FC}(t) \times \frac{V_F(t)\eta_{FC}}{\Delta t}. \quad (5)$$

In Equation (5), $P_{FC}(t)$ is the electrical power output of the fuel cell at time period t, $V_F(t)$ is the amount of natural gas consumed, and η_{FC} is the efficiency of the fuel cell. $\delta_{FC}(t)$ is the binary decision variable of the fuel cell at time t.

2.5. Battery Model

Batteries can store electrical energy when the load demand is low and release it when the load is high to flatten peaks and fill valleys. The mathematical model is as follows [40]:

$$E_{BT}(t+1) = E_{BT}(t) - \eta^b_{\text{in}} P_{BT_{\text{in}}}(t)\Delta t, \tag{6}$$

$$E_{BT}(t+1) = E_{BT}(t) - P_{BT_{\text{out}}}(t)\Delta t / \eta^b_{\text{out}}, \tag{7}$$

$$SOC(t+1) = \begin{cases} SOC(t) + \frac{\eta^b_{in} P_{BT_{\text{in}}}(t)\Delta t \delta_{\text{in}}(t)}{E_{BT\max}}, & \text{if charge,} \\ SOC(t) - \frac{P_{BT_{\text{out}}}(t)\Delta t \delta_{\text{out}}(t)}{\eta^b_{\text{out}} E_{BT\max}}, & \text{if discharge.} \end{cases} \tag{8}$$

In Equations (6)–(8), $E_{BT}(t+1)$ and $E_{BT}(t)$ are the energy storage levels of the battery at times $t+1$ and t, respectively, $E_{BT_{\max}}$ is the maximum energy storage capacity of the battery, $P_{BT_{\text{in}}}$ is the battery charging power, and $P_{BT_{\text{out}}}$ is the battery discharging power. $SOC(t)$ is the charged state of the battery at time t, and $SOC(t+1)$ is the charged state of the battery at time $t+1$. As for η^b_{in} and η^b_{out}, these are the charging and discharging efficiencies of the battery, respectively, and $\eta^b_{\text{in}} = \eta^b_{\text{out}} = 0.9$. $\delta_{\text{in}}(t)$ and $\delta_{\text{out}}(t)$ are the decision variables of charging and discharging at time t, respectively.

2.6. Objective Function for Daily Operating Cost

This paper establishes a real-time economic dispatch model for microgrids with consideration given to the photovoltaic power generation units, wind turbine power generation units, fuel cell power generation units, micropower generation units, and battery energy storage. The model optimizes the output of controllable generation equipment to minimize the system's economic operating cost, environmental cost, and maintenance cost.

$$\min C = \sum_{i=1}^{T} (C_{NG} + C_{ma} + C_{ex} + C_{en}). \tag{9}$$

In Equation (9), C_{NG} is the fuel cost for the system operation, C_{ma} is the maintenance cost of the system's equipment, C_{ex} is the cost of the grid interaction, and C_{en} is the cost of the pollutant treatment generated by the equipment.

The various cost expressions are as follows:

(1) The fuel cost for system operation

$$C_{NG} = \sum_{i=1}^{N} C_i(P_{it}) = \sum_{i=1}^{N} C_{ng} \times \frac{1}{LHV_{ng}} \times \frac{P_{it}}{\eta_{it}}. \tag{10}$$

In Equation (10), C_i is the fuel cost, N is the type of power generation unit, P_{it} is the output power of the type of power generation unit i at time t, C_{ng} is the price of natural gas, LHV_{ng} is the low calorific value of natural gas, with a value of 9.7 kW · h/m^3, and η_{it} is the efficiency of the corresponding unit.

(2) The maintenance cost of the system's equipment

$$C_{ma} = \sum_{i=1}^{N} DP_i(P_{it}) = \sum_{i=1}^{N} \frac{ADCC_i}{P_{N,i} \times 8760 \times cf_i}, \tag{11}$$

$$ADCC_i = InsCost_i \times CFR_i, \tag{12}$$

$$CFR_i = \frac{d_i \times (1+d_i)^{L_i}}{(1+d_i)^{L_i} - 1}. \tag{13}$$

In Equations (11)–(13), DP_{i} is the depreciated maintenance cost, N is the generation unit type (WT, PV, MT, FC), $ADCC_i$ is the average annual depreciation capital, $P_{N,i}$ is the maximum output power, cf_i is the capacity factor, $InsCost_i$ is the installation cost per unit capacity, CFR_i is the capital recovery factor, d_i is the annual depreciation rate, and L_i is the depreciation life of the generation unit type.

(3) The cost of grid interaction

$$C_{ex} = CP_t \times CGP_t - CS_t \times CSP_t. \tag{14}$$

In Equation (14), CP_t is the electricity price from the microgrid to the distribution network at time t, CGP_t is the electricity price from the distribution network at time t, CS_t is the electricity price from the microgrid to the distribution network at time t, and CSP_t is the electricity sold from the microgrid to the distribution network at time t.

(4) The cost of pollutant treatment generated by the equipment

$$C_{en} = \sum_{i=1}^{N} C_{eni}(P_{it}) = \sum_{i=1}^{N} \left(\sum_{k=1}^{M} \alpha_{ik} \times \lambda_{ik} \times P_{it} \right). \tag{15}$$

In Equation (15), CP_t is the electricity price from the microgrid to the distribution network at time t, CGP_t is the electricity price from the distribution, C_{eni} is the pollutant emission treatment cost of power generation unit type i, M is the emission type, α_{ik} is the unit sewage treatment cost of power generation unit type i when the emission type k is discharged, and λ_{ik} is the emission coefficient of power generation unit type i when the emission type k.

2.7. Constraints

In this microgrid scheduling model, constraints are imposed on various distributed power sources and equipment to ensure power balance and to meet the upper and lower limits on the output of each generation unit.

2.7.1. Power Balance Constraint

$$P_{\mathrm{grid}}(t) + P_{PV}(t) + P_{WT}(t) + P_{MT}(t) + P_{FC}(t) + P_{BT_{\mathrm{out}}}(t) - P_{BT_{\mathrm{in}}}(t) = P_L(t). \tag{16}$$

In Equation (16), $P_{\mathrm{grid}}(t)$, $P_{PV}(t)$, $P_{WT}(t)$, $P_{MT}(t)$, and $P_{FC}(t)$ are the power exchange with the grid, the power generation of the photovoltaic and wind turbines, and the power generation of the micro gas turbine and fuel cell at time t, respectively. $P_{BT_{\mathrm{out}}}(t)$ is the discharge power of the battery at time t, and $P_{BT_{\mathrm{in}}}(t)$ is the charging power of the battery at time t. $P_L(t)$ is the total load in time period t. The power values are expressed in kW.

2.7.2. Constraints on Power Purchase and Sale with the Grid

$$\begin{cases} 0 \le P_{\mathrm{grid},g} \le P_{\mathrm{grid},g_{\max}}, \\ 0 \le P_{\mathrm{grid},s} \le P_{\mathrm{grid},s_{\max}}. \end{cases} \tag{17}$$

In Equation (17), $P_{\mathrm{grid},g_{\max}}$ and $P_{\mathrm{grid},s_{\max}}$ are the maximum power for purchasing and selling with the grid, respectively. The power values are expressed in kW.

2.7.3. Constraints on Controllable Generation Unit Output

$$P_{x_{\min}} \le P_x \le P_{x_{\max}}. \tag{18}$$

In Equation (18), $P_{x_{\min}}$ and $P_{x_{\max}}$ are the lower and upper limits of the output power for each generation unit, respectively. The power values are expressed in kW.

2.7.4. Micro Gas Turbine Ramp Constraint

$$R_{MT_{\text{down}}} \leq P_{MT}(t) - P_{MT}(t-1) \leq R_{MT_{\text{up}}}. \tag{19}$$

In Equation (19), $R_{MT_{\text{up}}}$ and $R_{MT_{\text{down}}}$ are the upper and lower limits of the ramp power for the micro gas turbine unit, respectively.

2.7.5. Energy Storage Battery Constraints

$$\begin{cases} E_{BT}^{\min}(t) \leq E_{BT}(t) \leq E_{BT}^{\max}(t), \\ SOC_{\min}(t) \leq SOC(t) \leq SOC_{\max}(t), \\ E_{BT}^{T} = E_{BT}^{0}, \\ \delta_{\text{in}}(t) + \delta_{\text{out}}(t) \leq 1. \end{cases} \tag{20}$$

In Equation (20), $E_{BT}^{\max}(t)$, $E_{BT}^{\min}(t)$ are the upper and lower limits of the output power of the battery at time t. Considering the charge and discharge loss of the battery, we set $SOC_{\max}(t)$ at 95% and $SOC_{\min}(t)$ at 5%. As for E_{BT}^{T} and E_{BT}^{0}, they are the energy storage levels at the final and initial states, respectively. As for $\delta_{\text{in}}(t)$ and $\delta_{\text{out}}(t)$, they are the decision variables of charging and discharging at time t, respectively.

3. The Principle and Improvement Measures of Honey Badger Algorithm

This chapter introduces the basic principle of the honey badger algorithm and expounds its improvement measures, and explains and analyzes the operation process of the improved honey badger algorithm.

3.1. Honey Badger Algorithm

The honey badger algorithm (HBA) is a novel intelligent optimization algorithm with strong exploration and exploitation capabilities. It is based on the foraging behavior of honey badgers and has two foraging modes: one is autonomous excavation, where the honey badger uses its own sense of smell to locate prey and excavates the beehive along a heart-shaped trajectory, and the other is the honey collection, where the honey badger relies on a guiding bird to collect honey. By simulating the flexible foraging behavior of honey badgers, the algorithm can balance global exploration and local exploitation, efficiently escape local optima, find global optimal solutions, and demonstrate high adaptability and robustness [13].

3.1.1. Population Initialization

$$x_i = lb_i + r_1(ub_i - lb_i). \tag{21}$$

In Equation (21), x_i is the position of the i-th honey badger, with $i = 1, 2, \ldots, n, n$ representing the total number of honey badger individuals. As for ub_i and lb_i, they are the upper and lower bounds of the search space, and r_1 is a random number in the range $(0, 1)$.

3.1.2. Defining Odor Intensity

The odor intensity I determines the foraging speed of the honey badger. The larger I is, the faster the honey badger's foraging speed. The definition of odor intensity is as follows [41]:

$$I = r_2 \frac{S}{4\pi d_i^2}, \tag{22}$$

$$S = (x_i - x_{i+1})^2, \tag{23}$$

$$d_i = x_{\text{prey}} - x_i. \tag{24}$$

In Equations (22)–(24), S is the concentration intensity, d_i is the distance between the prey and the i-th honey badger, and r_2 is a random number between 0 and 1.

3.1.3. Updating Density Factor

The density factor α ensures a smooth transition from exploration to exploitation. It decreases with the number of iterations, reducing the randomness over time. The definition is as follows:

$$\alpha = C \times \exp\left(\frac{-t}{t_{\max}}\right). \tag{25}$$

In Equation (25), $t_{\max}$ is the maximum number of iterations, and $C \geq 1$, typically set to 2.

3.1.4. Excavation Stage

During the excavation stage, the honey badger updates its individual position in a motion similar to a cardioid, as shown in the following equation:

$$x_{\text{new}} = x_{\text{prey}} + F \times \beta \times I \times x_{\text{prey}} + F \times r_3 \times \alpha \times d_i \times |\cos(2\pi r_4) \times [1 - \cos(2\pi r_5)]|. \tag{26}$$

In Equation (26), x_{new} is the new position of the honey badger, x_{prey} is the current global best position, $\beta(\geq 1)$ is the honey badger's ability to find food, d_i is the distance between the prey and the i-th honey badger, r_3, r_4, and r_5 are three different random numbers between 0 and 1. F is a flag that changes the search direction, calculated as:

$$F = \begin{cases} 1, & \text{if } r_6 \leq 0.5, \\ -1, & \text{else.} \end{cases} \tag{27}$$

In Equation (27), r_6 is a random number in the range (0, 1).

3.1.5. Honey Harvesting Phase

In the honey collection phase, the honey badger updates its individual position following the honeyguide bird, as shown in the following equation:

$$x_{\text{new}} = x_{\text{prey}} + F \times r_7 \times \alpha \times d_i. \tag{28}$$

In Equation (28), x_{new} represents the new position of the honey badger, x_{prey} represents the current global best position, F and α are determined by the equation, and r_7 is a random number in the range (0, 1). In this stage, the search is influenced by the search behavior α, which changes over time.

3.2. Improved Honey Badger Algorithm

3.2.1. Population Initialization Based on Cooperative–Competitive Learning

The honey badger algorithm typically uses a random method to generate the initial population of individuals. However, the initial population generated by random methods cannot guarantee their diversity, nor can it effectively extract useful information from the search space, which may affect the search efficiency of the algorithm to some extent. Combined opposition-based learning (COBL) is an enhancement strategy for optimization algorithms that consider both the current candidate solution and its opposite solution to improve the diversity and convergence speed of the population [42]. Therefore, this paper proposes to improve the honey badger algorithm using the COBL strategy. For a given candidate solution x_i within a search space defined by lower and upper bounds, the opposite solution x_{pop} is defined as follows:

$$x_{\text{pop}} = lb_i + ub_i - x_i. \tag{29}$$

In Equation (29), x_{pop} represents the opposite solution, and lb_i and ub_i are vectors representing the lower and upper bounds for each dimension. Calculate the fitness values of the original solution and the opposite solution, and retain the solution with the higher fitness value for the next step. This approach introduces more potentially good solutions

during the population initialization stage, improving the diversity and overall quality of the population.

3.2.2. Fixed Parameter Linear Decrease Strategy

In the original algorithm, C is a fixed constant greater than or equal to 1. However, a fixed value of C cannot balance the exploration of simple and complex environments throughout the entire solving process to find the optimal solution. It lacks adaptability to different environments and can weaken the algorithm's exploitation ability [43]. In the early stages of algorithm optimization, a sufficiently large C is needed to search more space. In later iterations, the algorithm needs to be more precise for exploitation and requires a smaller value. Therefore, a linear decrease strategy for the parameter is proposed.

$$C = (C_{\min} - C_{\max}) \times \frac{\mathrm{t}}{T} + C_{\max}. \tag{30}$$

In Equation (30), $C_{\min}$ and $C_{\max}$ are, respectively, the maximum and minimum values of the parameter. By plugging the improved C into the density factor formula shown in Equation (25), according to its characteristics, there is a large enough density factor to find more space in the early exploration phase of optimization. In later iterations, the algorithm needs to be more precise to reduce the density factor.

3.2.3. Introducing Spiral Factor

In the process of finding the optimal solution, the search range of the algorithm is larger, but the number of overshoots also increases, so it is necessary to add a variable spiral factor to search within a certain space.

$$H = \alpha_k \times \cos(k \times l \times \pi), \tag{31}$$

$$a_k = \begin{cases} 1, & t < \frac{M}{2}, \\ e^{5l}, & \text{otherwise.} \end{cases} \tag{32}$$

$$l = 1 - 2 \times t/M. \tag{33}$$

In Equations (31)–(33), H represents the variable spiral factor [43], α_k is a parameter used to control the spiral, with a value close to 1 in the early iterations and gradually decreasing later on, k is $M/10$, and l is a parameter that linearly decreases from 1 to -1 as the number of iterations increases. M is the maximum number of iterations. After introducing the spiral factor, the position update formula for the exploitation mode is shown in (34):

$$x_{\text{new}} = x_{\text{prey}} + F \times \beta \times I \times x_{\text{prey}} + F \times r_3 \times \alpha \times d_i \times |\cos(2\pi r_4) \times [1 - \cos(2\pi r_5)]| \times H. \tag{34}$$

In the process of searching for the optimal solution, the honey badger algorithm has a wide search range but also increases the number of ineffective searches. To address this, a variable spiral factor is introduced to fully utilize the entire search space and conduct effective searches within the space, thereby avoiding the attraction of locally optimal solutions and improving effectiveness. As shown in Figure 1, in the early stages, the algorithm can conduct a wide range of spiral searches in the entire space, enhancing the algorithm's global search capability. In the later stages, the range gradually decreases, enhancing the algorithm's localization and development capabilities. In the honey badger algorithm, individuals in the exploitation mode play a dominant role. Therefore, this paper uses the spiral factor for the position update in the exploitation mode, allowing other individuals to also improve their search capabilities.

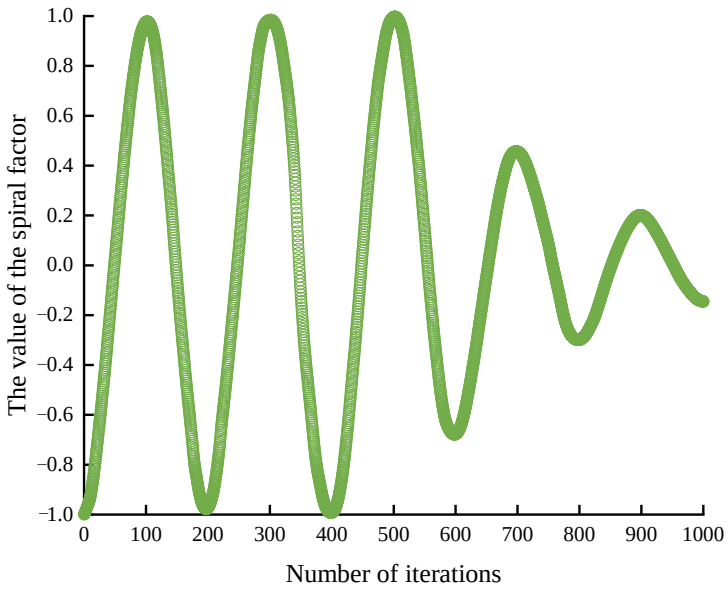

Figure 1. Diagram of the spiral factor principle.

3.2.4. Starvation Search Strategy

In the honey collection phase of the honey badger algorithm, individuals tend to prematurely converge near the prey's location under the guidance of the guide bird. To avoid the algorithm becoming stuck in local optima, a new starvation search strategy is introduced. After updating the position in the honey collection phase, if an individual exhibits low energy and poor fitness, it is defined as being in a state of starvation. By immediately changing the search path of the individual in a starvation state, the algorithm's search capability is enhanced, helping the individual to overcome starvation and allowing the algorithm to escape from local optima.

$$x_{\text{news}} = x_{\text{prey}} + F \times r_7 \times \alpha \times d_i \times \exp(x_{\text{worse}} - x_i). \tag{35}$$

In Equation (35), x_{worse} represents the position of the globally worst individual, and x_i represents the position of the current individual in the population. From the formula, it can be seen that the starvation search strategy is based on the improvement of the difference between the globally worst position and the current position. After improvement, the honey badger algorithm has a wider search range in the exploration phase, which allows it to better reach the global optimum.

3.3. Implementation Process of the MIHBA Algorithm

The improved honey badger algorithm's computation process is as follows: randomly generate an initial population and initialize the original data of the microgrid system and various parameters in the honey badger algorithm. Then, set the calculation value of the objective function as the fitness value of the honey badger individual. Calculate the fitness value of the honey badger individual, record the best solution, and derive candidate solutions based on combined opposition-based learning. Calculate the fitness values of the original solution and the opposite solution and retain the solution with the higher fitness value. Update the density factor according to the linear decrease strategy, then conduct global search iterations and update the positions of the honey badger individuals. Introduce the spiral factor in the exploration process of the exploitation mode. After updating the positions and fitness values of the honey badger individuals in the exploration and honey collection modes, update the optimal position for individuals with a significant fitness difference through the starvation search strategy. Record the individual with the best fitness value in the current population and check if the termination condition of the algorithm is met. If the termination condition is met, output the individual with the best fitness value in

the current population as the optimal solution. If not, return to continue the iteration. The flowchart is shown in Figure 2.

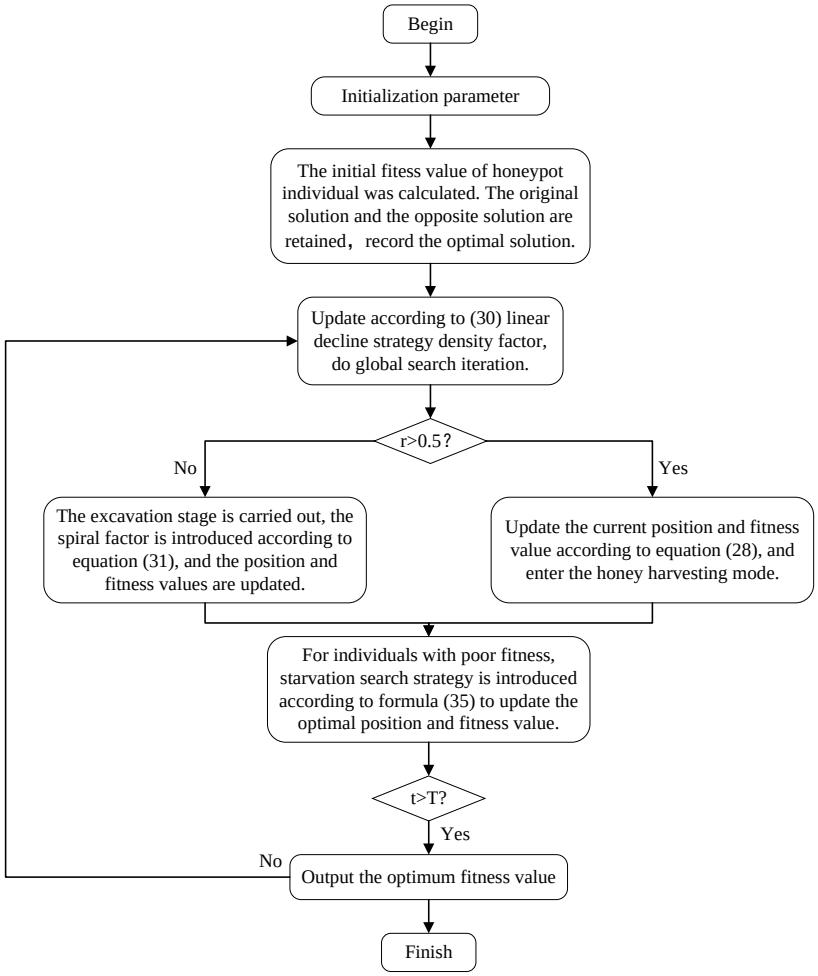

Figure 2. MIHBA algorithm flowchart.

4. Algorithm Performance Testing and Comparison

This chapter illustrates the testing process of the improved honey badger algorithm and other common optimization algorithms by comparing different unimodal and bimodal test functions. The experimental results based on actual data are included to show whether the improved honey badger algorithm performs better than the traditional algorithm in different test scenarios. In general, this chapter demonstrates the performance differences of different algorithms through empirical comparison and verifies the effectiveness of the improved honey badger algorithm.

4.1. Selection of Test Functions

To verify the optimization effectiveness and stability of MIHBA, we compared it with the genetic algorithm (GA), particle swarm optimization (PSO), whale optimization algorithm (WOA), grey wolf optimization (GWO), sparrow search algorithm (SSA), as well as the original honey badger algorithm and the improved honey badger algorithm. We used eight unimodal and multimodal benchmark test functions to measure the algorithm's performance based on three evaluation criteria: Mean, STDEV.P (population standard deviation), and best. A comparative analysis was then conducted.

4.2. Parameter Settings and Test Environment

This paper conducted simulation experiments using MATLAB R2021b. The processor was an AMD Ryzen 7 6800H with Radeon Graphics 3.20 GHz, with 4 GB of RAM, and running on Windows 10. The population size was set to 50, the maximum number of iterations was 1000, and the algorithm dimensions were 10, 30, and 100, respectively. The algorithm parameters for GA, PSO, WOA, GWO, SSA, HBA, and MIHBA are shown in Table 1.

Table 1. Algorithm simulation experiment parameters.

Optimization Algorithm	Parameters
GA	$P_c = 0.8, P_m = 0.2$
PSO	$W = 0.9, W_{\text{damp}} = 0.9, c_1 = 2, c_2 = 2$
WOA	$B = 1$
GWO	$\omega_1 = \omega_2 = \omega_3 = \frac{1}{3}$
NGO	$P = 0.2$
HBA	$\beta = 6, c = 2$
MIHBA	$\beta = 6$

4.3. Analysis of Test Function Results

In this paper, multiple experiments were conducted using MATLAB simulation software to evaluate the performance of different algorithms on various test functions. The resulting metrics for each algorithm are shown in Table 2. Functions F1 to F5 are unimodal test functions, while F6 to F8 are multimodal test functions [44]. Unimodal test functions are used to simulate problems with globally optimal solutions. It has different peaks or extremes throughout the search space, allowing the optimization algorithm to test its performance in simple cases. The bimodal test function is used to simulate problems with multiple locally optimal solutions—but only one global optimal solution—and it can test the ability of the optimization algorithm to avoid the trap of locally optimal solutions in complex environments.

From Table 2 and Figure 3, the optimal adaptive values are obtained by different algorithms in the iterative process of different test functions. For example, the test function in Figure 3 focuses on accuracy evaluation, and some algorithms quickly reach the minimum adaptive value, indicating that the MIHBA algorithm can quickly find the optimal solution. Figure d shows the performance of each algorithm under high precision requirements. MIHBA and HBA have obvious advantages, and the adaptive value drops to an extremely low level. When comparing overall performance, MIHBA showed fast and stable convergence in almost all tests. It can be concluded that when the dimension is constant and under the same test function, MIHBA shows a faster convergence speed and higher convergence accuracy compared to GA, PSO, WOA, GWO, NGO, and HBA, demonstrating better performance. In the unimodal benchmark test function F1, both HBA and MIHBA can find the theoretical optimal value, but HBA is significantly weaker than MIHBA in terms of convergence accuracy. In the unimodal benchmark test functions F2 to F4, HBA's various metrics are generally superior to other algorithms by several orders of magnitude, and MIHBA can consistently find the optimal value, highlighting the superiority of the improved algorithm. For the unimodal benchmark test function F5, although MIHBA did not find the theoretical optimal solution, its three evaluation metrics are better than those of other algorithms. For the multimodal benchmark test function F7, compared to other algorithms, HBA and MIHBA can stably find the optimal value across different dimensions, demonstrating the inherent superiority of the honey badger algorithm.

Table 2. The metrics for different algorithms on various test functions.

Function	Dimension	Metric	GA	PSO	WOA	GWO	NGO	HBA	MIHBA
F1	10	Best	7.65×10^{-5}	2.09×10^{-4}	2.32×10^{-193}	4.01×10^{-148}	1.67×10^{-206}	0.00	0.00
		Mean	2.93×10^{-4}	5.71×10^{-4}	1.34×10^{-175}	8.84×10^{-146}	1.65×10^{-204}	0.00	0.00
		Std	1.38×10^{-4}	3.08×10^{-4}	0.00	1.93×10^{-145}	2.82×10^{-204}	0.00	0.00
	30	Best	5.23×10^{-1}	1.14×10^{-1}	4.53×10^{-183}	1.88×10^{-72}	1.09×10^{-183}	1.83×10^{-305}	0.00
		Mean	8.23×10^{-1}	1.85×10^{-1}	5.33×10^{-173}	8.70×10^{-70}	7.61×10^{-182}	3.13×10^{-301}	0.00
		Std	3.64×10^{-1}	6.51×10^{-2}	0.00	1.82×10^{-69}	1.17×10^{-181}	4.3×10^{-301}	0.00
	100	Best	1.30×10^{4}	1.32×10	1.35×10^{-178}	2.75×10^{-35}	4.51×10^{-174}	5.62×10^{-276}	0.00
		Mean	1.73×10^{4}	1.51×10	5.49×10^{-171}	2.77×10^{-34}	6.74×10^{-172}	1.07×10^{-269}	0.00
		Std	3.17×10^{3}	1.29	0.00	3.34×10^{-34}	1.44×10^{-171}	0.00	0.00
F2	10	Best	9.55×10^{-4}	6.55×10^{-2}	3.71×10^{-121}	1.08×10-82	3.54×10^{-106}	4.75×10^{-197}	0.00
		Mean	2.14×10^{-3}	9.61×10^{-2}	1.18×10^{-116}	2.23×10^{-80}	1.99×10^{-104}	6.46×10^{-195}	0.00
		Std	8.18×10^{-4}	2.83×10^{-2}	1.76×10^{-116}	4.41×10^{-80}	3.11×10^{-104}	1.07×10^{-194}	0.00
	30	Best	1.18×10^{-1}	1.66	3.77×10^{-116}	1.50×10^{-41}	1.34×10^{-93}	9.83×10^{-163}	0.00
		Mean	1.90×10^{-1}	2.51	1.50×10^{-109}	5.84×10^{-41}	6.64×10^{-93}	1.25×10^{-156}	0.00
		Std	8.98×10^{-2}	8.22×10^{-1}	2.69×10^{-109}	6.67×10^{-41}	4.20×10^{-93}	2.23×10^{-156}	0.00
	100	Best	4.61 × 10	3.52 × 10	1.17×10^{-114}	5.52×10^{-21}	2.88×10^{-89}	6.06×10^{-145}	2.20×10^{-239}
		Mean	5.75 × 10	4.36 × 10	1.38×10^{-107}	1.07×10^{-20}	4.10×10^{-89}	2.90×10^{-142}	1.45×10^{-235}
		Std	7.47	1.02 × 10	2.75×10^{-107}	6.11×10^{-21}	7.37×10^{-90}	6.07×10^{-142}	1.36×10^{-234}
F3	10	Best	1.06 × 10	3.44×10^{-3}	1.51×10^{-3}	8.34×10^{-73}	5.21×10^{-99}	7.00×10^{-308}	0.00
		Mean	1.12×10^{2}	7.02×10^{-3}	2.79	7.01 × 10-67	1.09×10^{-96}	3.41×10^{-294}	0.00
		Std	1.31×10^{2}	4.04×10^{-3}	6.06	1.57 × 10-66	1.28×10^{-96}	7.63×10^{-294}	0.00
	30	Best	1.22×10^{4}	1.20 × 10	4.88×10^{3}	1.42×10^{-23}	9.75×10^{-54}	6.39×10^{-232}	0.00
		Mean	1.49×10^{4}	1.40 × 10	1.22×10^{4}	2.35×10^{-19}	1.86×10^{-48}	1.67×10^{-227}	0.00
		Std	2.69×10^{3}	2.35	7.14×10^{3}	3.13×10^{-19}	3.06×10^{-48}	3.30×10^{-227}	0.00
	100	Best	1.58×10^{5}	3.56×10^{3}	6.97×10^{5}	1.71×10^{-3}	2.02×10^{-30}	3.76×10^{-190}	0.00
		Mean	2.08×10^{5}	4.82×10^{3}	7.69×10^{5}	5.38×10^{-1}	2.41×10^{-22}	2.87×10^{-180}	0.00
		Std	3.35×10^{4}	8.91×10^{2}	6.68×10^{4}	7.36×10^{-1}	5.20×10^{-22}	0.00	0.00
F4	10	Best	4.25×10^{-2}	1.84×10^{-2}	1.02×10^{-7}	3.92×10^{-47}	1.55×10^{-92}	2.72×10^{-166}	0.00
		Mean	1.04×10^{-1}	2.94×10^{-2}	5.83×10^{-3}	7.89×10^{-45}	3.76×10^{-91}	5.56×10^{-162}	0.00
		Std	5.44×10^{-2}	1.40×10^{-2}	1.28×10^{-2}	1.57×10^{-44}	5.49×10^{-91}	1.24×10^{-161}	0.00
	30	Best	1.37 × 10	5.36×10^{-1}	7.61×10^{-1}	8.02×10^{-19}	2.00×10^{-77}	4.95×10^{-134}	0.00
		Mean	1.81 × 10	1.13	4.16 × 10	2.42×10^{-17}	7.41×10^{-77}	3.00×10^{-127}	0.00
		Std	3.31	6.11×10^{-1}	3.09 × 10	3.27×10^{-17}	6.00×10^{-77}	6.22×10^{-127}	0.00
	100	Best	6.64 × 10	7.67	5.80	1.30×10^{-5}	3.92×10^{-70}	1.27×10^{-93}	0.00
		Mean	7.49 × 10	9.28	6.33 × 10	5.67×10^{-5}	8.97×10^{-70}	1.13×10^{-88}	0.00
		Std	6.48	1.27	3.82 × 10	6.40×10^{-5}	6.46×10^{-70}	2.42×10^{-88}	0.00
F5	10	Best	3.16×10^{-3}	8.92×10^{-2}	1.81×10^{-5}	4.46×10^{-5}	4.089×10^{-5}	1.46×10^{-5}	7.80×10^{-6}
		Mean	4.22×10^{-3}	2.83×10^{-1}	1.90×10^{-3}	1.45×10^{-4}	1.93×10^{-5}	1.15×10^{-4}	2.19×10^{-5}
		Std	1.14×10^{-3}	1.27×10^{-1}	1.72×10^{-3}	1.15×10^{-4}	1.23×10^{-5}	1.05×10^{-4}	1.32×10^{-5}
	30	Best	1.45×10^{-2}	1.14×10^{-1}	9.46×10^{-5}	6.45×10^{-5}	8.19×10^{-5}	1.38×10^{-5}	2.45×10^{-6}
		Mean	2.03×10^{-2}	4.68×10^{-1}	1.76×10^{-3}	3.82×10^{-4}	2.02×10^{-4}	1.20×10^{-4}	7.82×10^{-6}
		Std	4.97×10^{-3}	2.86×10^{-1}	3.17×10^{-3}	2.69×10^{-4}	1.06×10^{-4}	1.22×10^{-4}	4.47×10^{-6}
	100	Best	1.43 × 10	1.59×10^{-1}	1.86×10^{-4}	1.20×10^{-3}	2.45×10^{-4}	2.98×10^{-5}	3.31×10^{-7}
		Mean	2.38 × 10	2.38 × 10	9.95×10^{-4}	1.73×10^{-3}	3.42×10^{-4}	1.71×10^{-4}	3.08×10^{-5}
		Std	1.07 × 10	4.18 × 10	7.18×10^{-4}	4.98×10^{-4}	8.59×10^{-5}	1.30×10^{-4}	2.74×10^{-5}
F6	10	Best	2.82×10^{-5}	4.06	0.00	0.00	0.00	0.00	0.00
		Mean	2.46×10^{-4}	5.12	0.00	0.00	0.00	0.00	0.00
		Std	2.45×10^{-4}	1.25	0.00	0.00	0.00	0.00	0.00
	30	Best	8.20	5.53 × 10	0.00	0.00	0.00	0.00	0.00
		Mean	8.66	6.65 × 10	0.00	2.47	0.00	0.00	0.00
		Std	5.20×10^{-1}	1.20 × 10	0.00	5.53	0.00	0.00	0.00
	100	Best	3.45×10^{2}	3.67×10^{2}	0.00	2.27×10^{-13}	0.00	0.00	0.00
		Mean	3.91×10^{2}	5.06×10^{2}	0.00	2.96×10^{-13}	0.00	0.00	0.00
		Std	2.89 × 10	9.54 × 10	0.00	1.02×10^{-13}	0.00	0.00	0.00
F7	10	Best	3.73×10^{-3}	2.72×10^{-2}	4.44×10^{-16}	4.00×10^{-15}	4.00×10^{-15}	4.44×10^{-16}	4.44×10^{-16}
		Mean	8.26×10^{-3}	4.28×10^{-2}	3.29×10^{-15}	4.71×10^{-15}	4.00×10^{-15}	4.44×10^{-16}	4.44×10^{-16}
		Std	6.20×10^{-3}	1.81×10^{-2}	1.59×10^{-15}	1.59×10^{-15}	0.00	0.00	0.00
	30	Best	2.36×10^{-1}	2.23	4.44×10^{-16}	1.11×10^{-14}	4.00×10^{-15}	4.44×10^{-16}	4.44×10^{-16}
		Mean	4.66×10^{-1}	2.47	3.29×10^{-15}	1.32×10^{-14}	4.71×10^{-15}	4.44×10^{-16}	4.44×10^{-16}
		Std	2.38×10^{-1}	2.99×10^{-1}	2.97×10^{-15}	1.95×10^{-15}	1.59×10^{-15}	0.00	0.00
	100	Best	1.35 × 10	4.92	4.00×10^{-15}	6.44×10^{-14}	4.00×10^{-15}	4.44×10^{-16}	4.44×10^{-16}
		Mean	1.50 × 10	5.34	4.71×10^{-15}	6.94×10^{-14}	6.84×10^{-15}	4.44×10^{-16}	4.44×10^{-16}
		Std	1.29	4.86×10^{-1}	1.59×10^{-15}	5.39×10^{-15}	1.59×10^{-15}	0.00	0.00
F8	10	Best	3.50×10^{-2}	5.67×10^{-2}	0.00	0.00	0.00	0.00	0.00
		Mean	6.83×10^{-2}	1.65×10^{-1}	8.00×10^{-2}	2.91×10^{-2}	0.00	0.00	0.00
		Std	2.08×10^{-2}	1.14×10^{-1}	9.24×10^{-2}	2.55×10^{-2}	0.00	0.00	0.00
	30	Best	7.02×10^{-1}	2.74×10^{-2}	0.00	0.00	0.00	0.00	0.00
		Mean	8.74×10^{-1}	4.21×10^{-2}	0.00	0.00	0.00	0.00	0.00
		Std	1.31×10^{-1}	2.35×10^{-2}	0.00	0.00	0.00	0.00	0.00
	100	Best	1.76×10^{2}	4.27×10^{-1}	0.00	0.00	0.00	0.00	0.00
		Mean	2.05×10^{2}	4.79×10^{-1}	0.00	3.62×10^{-3}	0.00	0.00	0.00
		Std	2.63 × 10	3.09×10^{-2}	0.00	8.09×10^{-3}	0.00	0.00	0.00

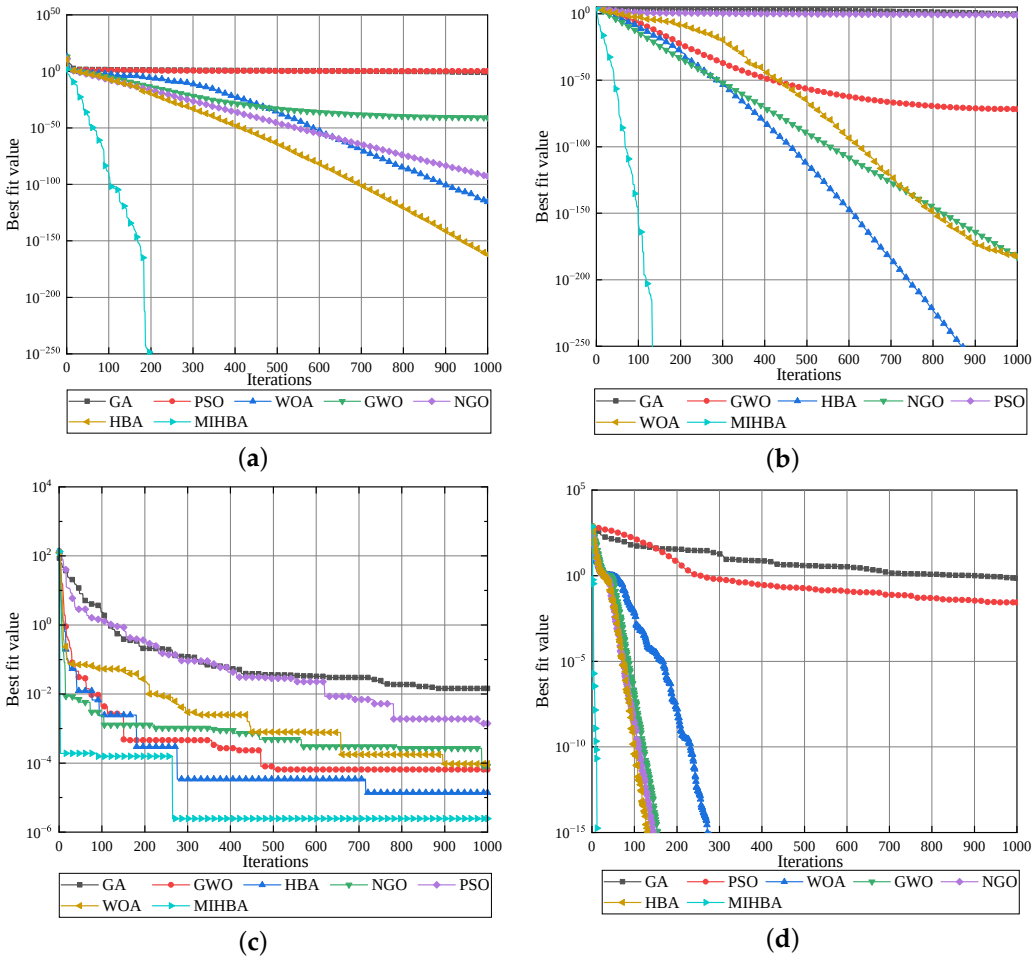

Figure 3. Convergence curves of selected test functions with 30 dimensions. (**a**) Convergence curve of test function F1 in dimension 30; (**b**) convergence curve of test function F2 in dimension 30; (**c**) convergence curve of test function F5 in dimension 30; (**d**) convergence curve of test function F8 in dimension 30.

In order to further verify the solving efficiency of MIHBA in different test functions, the solution times of different algorithms in the F1–F5 function in the 30 dimension is compared, as shown in Table 3.

Table 3. Comparison of the average running time of MIHBA algorithm and other algorithms on test functions (s).

Function	MIHBA	HBA	NGO	GWO	WOA	PSO	GA
F1	0.226	0.232	0.241	0.282	0.392	0.298	0.248
F2	0.176	0.212	0.240	0.218	0.239	0.396	0.256
F3	0.525	0.558	0.678	0.589	0.445	0.631	0.465
F4	0.409	0.445	0.609	0.576	0.450	0.619	0.509
F5	0.411	0.508	0.465	0.377	0.431	0.418	0.340

As shown in Table 3, the MIHBA algorithm consistently delivers the shortest average runtime across multiple test functions, particularly excelling in F1, F2, and F4 with times of 0.226 s, 0.176 s, and 0.409 s, respectively. Even in F3 and F5, although its performance is slightly inferior to that of a few individual algorithms, MIHBA still significantly outperforms the majority of others in terms of speed. Overall, MIHBA demonstrates superior efficiency and adaptability, making it a strong candidate for solving optimization problems.

5. Microgrid Model Simulation Analysis

Through simulation analysis of the microgrid, this chapter mainly shows the excellent performance of the improved honey badger algorithm under the cost optimization objective and proves its superiority by comparing it with other algorithms.

5.1. Basic Parameter Settings of Microgrid

To address the nonlinear issues in the microgrid model and verify the effectiveness of the microgrid model and the improved algorithm, the MIHBA algorithm is used to solve the microgrid optimization scheduling model. The relevant parameters for each power source are shown in Table 4, the time-of-use electricity price parameters are shown in Table 5 [45], and the pollutant emission parameters are shown in Table 6.

Table 4. Distributed power parameters.

Type of Power Supply	WT	PV	MT	FC	BT
Power Upper Limit/kW	20	40	20	50	20
Power Lower Limit/kW	0	0	0	0	−20
Operating Cost/(¥·kWh^{-1})	0.314	0.014	0.032	0.085	0.0016

Table 5. Microgrid time-of-use electricity prices.

Periods	Purchase Price (¥)	Sale Price (¥)
Off-Peak Hours (23:00–7:00)	0.52	0.32
Flat Hours (8:00–9:00, 15:00–19:00)	0.83	0.63
Peak Hours (10:00–14:00, 20:00–22:00)	1.13	0.88

Table 6. Emission amount and treatment cost of pollutants.

Pollutant Types	Treatment Costs/ (¥·kg^{-1})	Pollutant Emission Amounts (g·kWh^{-1})				
		WT	PT	MT	FC	Grid
SO2	19.034	0	0	0.041	0.004	1.841
NOX	65.249	0	0	0.32	0.022	1.626
CO	11.842	0	0	0.053	0	0.044

5.2. Simulation Results and Analysis

Figures 4–6, respectively, show the power curves for wind and photovoltaic loads, the power balance of distributed energy sources, and the output power of distributed energy sources in the microgrid. From the data in these figures, the following can be observed.

Figure 4 shows the power changes for the PV, wind, and microgrid loads. The photovoltaic curve peaks during the day, the output is zero at night, and the photovoltaic power generation is highest at noon. The effect of sunlight on the photovoltaic system can be observed through the curve of the day. The wind power generation curve shows relatively irregular fluctuations, indicating the volatility of wind energy, which is affected by wind speed and climate conditions. By looking at the load curve of the microgrid, it can be concluded that the load is lower in the morning and night (non-working hours) and higher in the day.

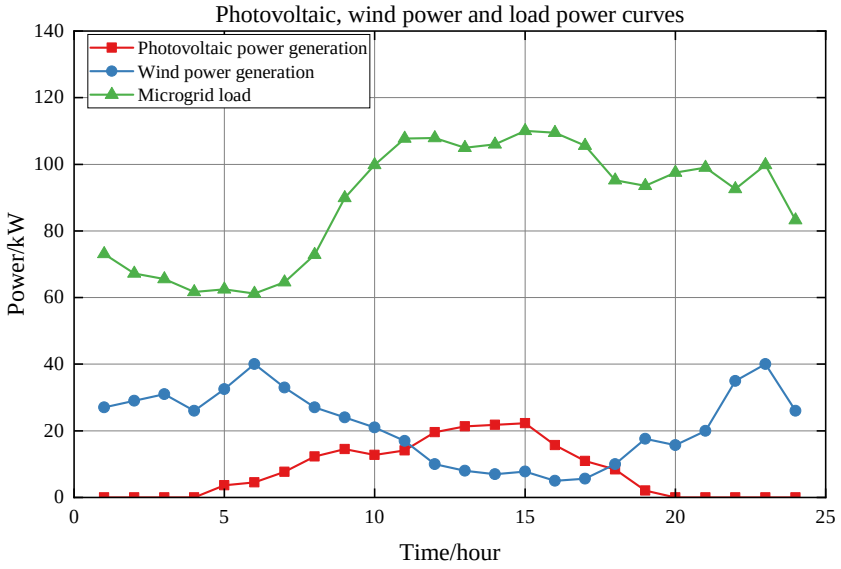

Figure 4. Photovoltaic, wind power, and microgrid load power curves.

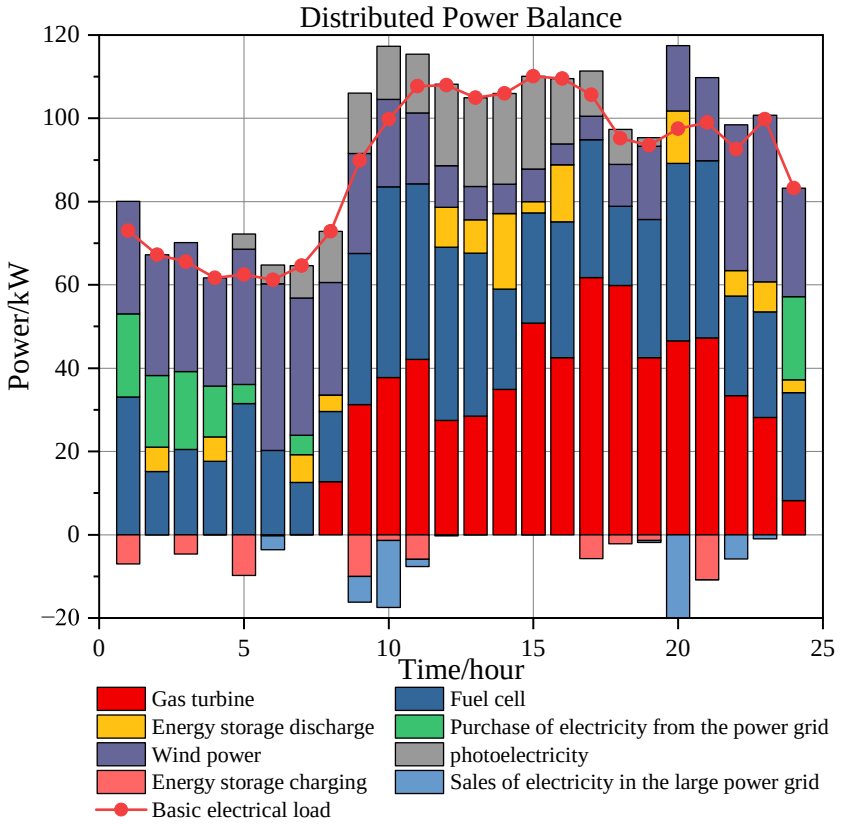

Figure 5. Distributed power source power balance.

Figure 5 shows in detail the dynamic balance of power supply and demand throughout the day, as well as the scheduling strategies for different types of power. The electricity demand fluctuates throughout the day, is low at night and in the morning, rises during the day, and peaks in the evening. Natural gas turbines and fuel cells provide stable power support at all times of demand. They are the main power source for the entire system, while with photovoltaic and wind power, photovoltaic output peaks in the middle of the day, indicating that the light is strongest at this time. Wind power generation is relatively irregular and is greatly affected by wind conditions. Regarding the purchase of electricity

from the grid when demand exceeds self-produced electricity (such as in the morning and at dusk); when there is a surplus of power supply (such as the peak of photovoltaic), electricity is sold to the grid to maximize economic benefits. It can be seen from this figure that the power management system achieves an all-weather power balance through the flexible combination of a variety of energy sources and the interaction with the grid, which not only meets the basic power demand but also realizes the improvement of economic benefits through electricity sales and optimization strategies.

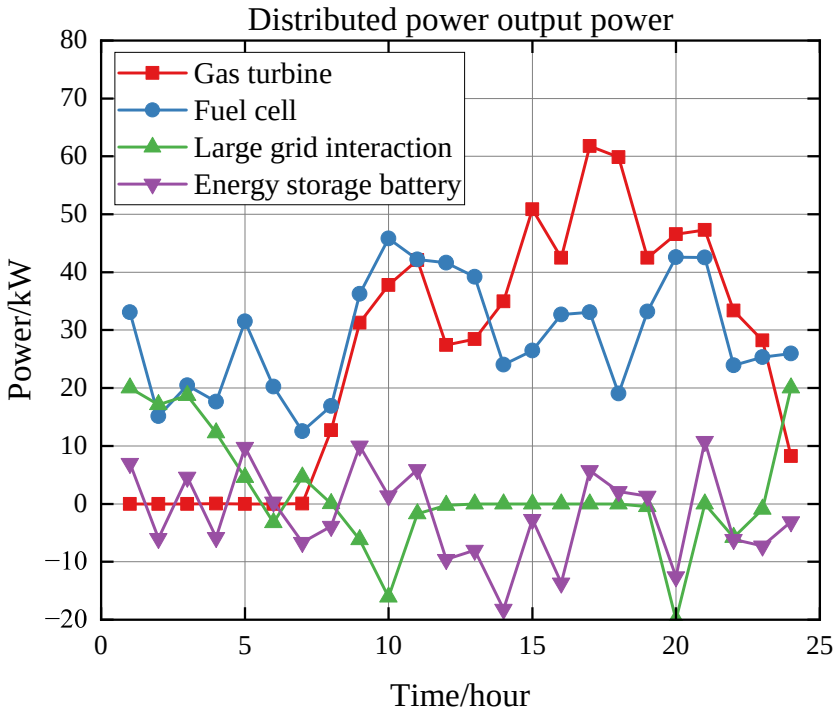

Figure 6. Distributed power source output power.

Figure 6 shows how the distributed power output changes over the course of a day, including the output power of four power sources. During times of high demand, natural gas turbines and fuel cells provide the main power, while batteries discharge to meet additional demand. In times of low demand, the system may purchase power from the grid and charge it at a low load.

By analyzing Figures 4–6, we can draw the following conclusions:

During the low-demand period from 23:00 to 7:00, due to low load and sufficient wind power, wind turbines continuously output power, and the pollution coefficient of the fuel cell is relatively low. Therefore, the fuel cell and wind power are used together. The battery continuously charges and discharges to meet the load demand. During this period, electricity prices are low, so only a small amount of electricity is purchased from the main grid to alleviate peak electricity costs.

During the medium-demand periods from 8:00 to 9:00 and 15:00 to 19:00, the load demand exceeds the total output of wind, photovoltaic, and fuel cells. Because of sufficient sunlight during these times, the priority is to use photovoltaic and wind turbines together, then use the fuel cell and gas turbine for auxiliary power generation. In this case, the cost of power generation from the fuel cell and battery is higher than the price of grid electricity, so the system purchases electricity from the grid to make up the difference.

During the high-demand periods from 10:00 to 14:00 and 20:00 to 22:00, to enhance the operational stability of the microgrid, wind and photovoltaic power generation are prioritized to meet the minimum generation requirements. Additionally, the output of fuel cells, gas turbines, and batteries increases, with fuel cells taking precedence over gas turbines and operating near full capacity. If the output limits of all distributed energy

sources are reached, additional power is purchased from the main grid to meet the overall load demand.

To verify the solving accuracy and stability of MIHBA, an initial population size of 50 and a maximum of 1000 iterations were set. The proposed microgrid optimization scheduling model was applied to MIHBA, HBA, GA, PSO, WOA, GWO, and NGO, with each algorithm solving the problem 30 times. The operational cost curves are shown in Figure 7, and the specific mean, standard deviation, maximum, and minimum values of each algorithm are listed in Table 7.

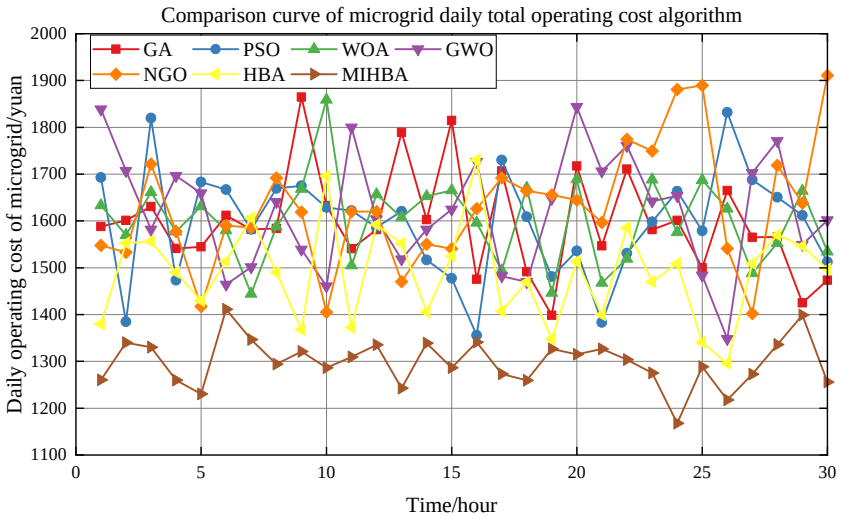

Figure 7. Comparison curve of daily operating costs algorithm for microgrid.

From the algorithm comparison curves shown in Figure 7, it can be observed that MIHBA achieved the lowest economic cost scheduling scheme in solving the microgrid optimization scheduling model. The optimal fitness values are uniformly distributed, and the curve fluctuations are significantly smaller than those of other algorithms.

Table 7. Comparison of algorithmic results.

Algorithm	Daily Operation Cost of Microgrid (Yuan)			
	Maximum Value	Minimum Value	Average Value	Standard Deviation
MIHBA	1411.138	1167.602	1298.381	51.399
HBA	1730.782	1295.395	1490.700	102.740
NGO	1910.764	1401.868	1629.087	129.897
GWO	1843.409	1347.808	1620.149	123.078
WOA	1858.906	1444.331	1599.982	89.967
PSO	1831.749	1355.912	1595.317	115.612
GA	1864.723	1398.646	1597.750	107.783

By analyzing the data in Table 7, the performance of several algorithms in the optimization of the daily operating cost of microgrids is given. By comparing the maximum value, minimum value, average value, and standard deviation, we can fully understand the efficiency and stability of these algorithms in solving microgrid optimization problems. The average values of MIHBA is 12.90%, 20.30%, 19.86%, 18.85%, 18.61%, and 18.74% lower than that of the HBA, NGO, GWO, WOA, PSO, and GA, respectively. The standard deviation of MIHBA was 49.97%, 60.43%, 58.24%, 42.87%, 55.54%, and 52.31% lower than that of the HBA, NGO, GWO, WOA, PSO, and GA, respectively. Specifically, the average cost of MIHBA is 1298.381 yuan, which is the lowest among all algorithms and 12.90∼20.38% lower than other algorithms. This significant cost reduction is a testament to MIHBA's efficiency. At the same time, the standard deviation of MIHBA is 51.399, which is also

significantly lower than other algorithms. This means that WHBA can maintain more consistent and stable results over multiple runs, further demonstrating its reliability. In contrast, the average costs of the HBA, NGO, GWO, WOA, PSO, and GA algorithms are generally higher, which are 1490.700 yuan, 1629.087 yuan, 1602.149 yuan, 1599.982 yuan, 1595.317 yuan, and 1599.750 yuan, respectively. In addition, the standard deviations of these algorithms are large; for example, the standard deviations of the NGO and the HBA are 129.897 and 102.740, respectively, showing the volatility and instability of the results. These results show that although these algorithms can achieve certain optimization effects when dealing with microgrid operation cost optimization, they are relatively insufficient in terms of stability and cost-effectiveness. Comprehensive analysis shows that MIHBA is superior to other algorithms in terms of cost-effectiveness and result stability and is suitable for microgrid systems with high stability requirements and low-cost requirements. The MIHBA algorithm not only reduces the operating costs, but also provides a more stable solution, which proves its theoretical superiority in practice.

In summary, MIHBA demonstrates superior solving accuracy and stability compared to other algorithms, confirming the advantages of the improved algorithm.

6. Conclusions

To address the shortcomings of the traditional honey badger algorithm in complex microgrid optimization and scheduling problems, such as slow convergence and susceptibility to local optima, this paper proposes an improved honey badger algorithm for microgrid optimization and scheduling. By introducing joint opposite learning strategies, variable spiral factors, parameter linear decreasing strategies, and starvation search strategies, the algorithm's global search capability and local development capability are significantly improved.

Through comparisons with classical intelligent optimization algorithms, new intelligent optimization algorithms, and traditional honey badger algorithms, the improved algorithm demonstrates significant enhancements in stability, convergence speed, and convergence accuracy in experiments involving multiple benchmark test functions and different dimensions. In specific applications of microgrid optimization and scheduling, the improved honey badger algorithm exhibits high solution accuracy and efficiency. In this paper, we compare the performance of various algorithms in the optimization of the daily operating costs of microgrids and identify the significant advantages of the MIHBA algorithm. MIHBA achieved the lowest average operating cost of only 1298.381 yuan, 12.90% to 20.38% lower than other algorithms, while maintaining the lowest result volatility with a standard deviation of 51.399, showing extremely high stability. Although other algorithms have some performance in optimization, they are relatively insufficient in terms of cost-effectiveness and stability. Therefore, the MIHBA algorithm provides the optimal solution for reducing operation costs and improving stability, indicating its practical application potential and value in microgrid optimization. The simulation results show that the algorithm has higher adaptability and robustness in microgrid optimization and scheduling, effectively handling complex optimization problems and improving the overall operational efficiency of microgrids.

Future studies can further explore the application of the algorithm in different types of microgrid environments and optimize its parameter settings to adapt to a wider range of practical application scenarios. As technologies such as artificial intelligence, machine learning, and the Internet of Things continue to advance, future research can also consider combining these emerging technologies with algorithms to further improve their efficiency. In general, future research should start with improving the performance of the algorithm and by expanding its application range so that it can become an important tool for responding to diverse energy needs and challenges in the evolving energy landscape.

Author Contributions: Conceptualization, Z.W. and Z.D.; methodology, Z.W.; validation, Z.W., Y.L., and J.G.; formal analysis, Z.W.; investigation, Z.W.; resources, Z.W.; data curation, Z.W.; writing—original draft preparation, Z.W.; writing—review and editing, Z.W.; visualization, J.Z.; supervision,

Z.D.; project administration, Z.D.; funding acquisition, W.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China under grant 52005306 and the Natural Science Foundation of Shandong Province under grant ZR2020QE220.

Data Availability Statement: Dataset available on request from the authors.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Gao, K.; Wang, T.; Han, C.; Xie, J.; Ma, Y.; Peng, R. A review of optimization of microgrid operation. *Energies* **2021**, *14*, 2842. [CrossRef]
2. Lei, B.; Ren, Y.; Luan, H.; Dong, R.; Wang, X.; Liao, J.; Fang, S.; Gao, K. A Review of Optimization for System Reliability of Microgrid. *Mathematics* **2023**, *11*, 822. [CrossRef]
3. Shezan, S.A.; Kamwa, I.; Ishraque, M.F.; Muyeen, S.; Hasan, K.N.; Saidur, R.; Rizvi, S.M.; Shafiullah, M.; Al-Sulaiman, F.A. Evaluation of different optimization techniques and control strategies of hybrid microgrid: A review. *Energies* **2023**, *16*, 1792. [CrossRef]
4. Chaudhary, V.; Pandit, M.; Dubey, H.M. Renewable Energy Integrated Economic Dispatch Using Intelligent Techniques: An Overview. In Proceedings of the International Conference on Communication and Computational Technologies, Jaipur, India, 26–27 February 2022; pp. 493–506.
5. Wei, H.; Wang, W.S.; Kao, X.X. A novel approach to hybrid dynamic environmental-economic dispatch of multi-energy complementary virtual power plant considering renewable energy generation uncertainty and demand response. *Renew. Energy* **2023**, *219*, 119406.
6. Azeem, M.; Malik, T.N.; Muqeet, H.A.; Hussain, M.M.; Ali, A.; Khan, B.; Rehman, A.U. Combined economic emission dispatch in presence of renewable energy resources using CISSA in a smart grid environment. *Electronics* **2023**, *12*, 715. [CrossRef]
7. Yan, N.; Ma, G.; Li, X.; Guerrero, J.M. Low-carbon economic dispatch method for integrated energy system considering seasonal carbon flow dynamic balance. *IEEE Trans. Sustain. Energy* **2022**, *14*, 576–586. [CrossRef]
8. Abd El-Sattar, H.; Hassan, M.H.; Vera, D.; Jurado, F.; Kamel, S. Maximizing hybrid microgrid system performance: A comparative analysis and optimization using a gradient pelican algorithm. *Renew. Energy* **2024**, *227*, 120480. [CrossRef]
9. Roslan, M.; Hannan, M.; Ker, P.J.; Begum, R.; Mahlia, T.I.; Dong, Z. Scheduling controller for microgrids energy management system using optimization algorithm in achieving cost saving and emission reduction. *Appl. Energy* **2021**, *292*, 116883. [CrossRef]
10. Tang, J.; Liu, G.; Pan, Q. A review on representative swarm intelligence algorithms for solving optimization problems: Applications and trends. *IEEE/CAA J. Autom. Sin.* **2021**, *8*, 1627–1643.
11. Li, W.; Wang, G.G.; Gandomi, A.H. A survey of learning-based intelligent optimization algorithms. *Arch. Comput. Methods Eng.* **2021**, *28*, 3781–3799.
12. Kumar, A.; Nadeem, M.; Banka, H. Nature inspired optimization algorithms: A comprehensive overview. *Evol. Syst.* **2023**, *14*, 141–156.
13. Hashim, F.A.; Houssein, E.H.; Hussain, K.; Mabrouk, M.S.; Al-Atabany, W. Honey Badger Algorithm: New metaheuristic algorithm for solving optimization problems. *Math. Comput. Simul.* **2022**, *192*, 84–110.
14. Mirjalili, S.; Lewis, A. The whale optimization algorithm. *Adv. Eng. Softw.* **2016**, *95*, 51–67.
15. Kennedy, J.; Eberhart, R. Particle swarm optimization. In Proceedings of the ICNN'95-International Conference on Neural Networks, Perth, WA, Australia, 27 November–1 December 1995; Volume 4, pp. 1942–1948.
16. Mirjalili, S.; Mirjalili, S.M.; Lewis, A. Grey wolf optimizer. *Adv. Eng. Softw.* **2014**, *69*, 46–61.
17. Yu, K.; Liang, J.; Qu, B.; Chen, X.; Wang, H. Parameters identification of photovoltaic models using an improved JAYA optimization algorithm. *Energy Convers. Manag.* **2017**, *150*, 742–753.
18. Deng, W.; Xu, J.; Zhao, H. An improved ant colony optimization algorithm based on hybrid strategies for scheduling problem. *IEEE Access* **2019**, *7*, 20281–20292.
19. Ahmed, Z.H.; Yousefikhoshbakht, M. An improved tabu search algorithm for solving heterogeneous fixed fleet open vehicle routing problem with time windows. *Alex. Eng. J.* **2023**, *64*, 349–363.
20. Deng, H.; Liu, L.; Fang, J.; Qu, B.; Huang, Q. A novel improved whale optimization algorithm for optimization problems with multi-strategy and hybrid algorithm. *Math. Comput. Simul.* **2023**, *205*, 794–817.
21. Shishavan, S.T.; Gharehchopogh, F.S. An improved cuckoo search optimization algorithm with genetic algorithm for community detection in complex networks. *Multimed. Tools Appl.* **2022**, *81*, 25205–25231.
22. Yıldız, B.S.; Kumar, S.; Panagant, N.; Mehta, P.; Sait, S.M.; Yildiz, A.R.; Pholdee, N.; Bureerat, S.; Mirjalili, S. A novel hybrid arithmetic optimization algorithm for solving constrained optimization problems. *Knowl.-Based Syst.* **2023**, *271*, 110554.
23. Wu, L.; Huang, X.; Cui, J.; Liu, C.; Xiao, W. Modified adaptive ant colony optimization algorithm and its application for solving path planning of mobile robot. *Expert Syst. Appl.* **2023**, *215*, 119410. [CrossRef]
24. Wei, F.; Zhang, Y.; Li, J. Multi-strategy-based adaptive sine cosine algorithm for engineering optimization problems. *Expert Syst. Appl.* **2024**, *248*, 123444. [CrossRef]

25. He, Y.; Wang, M. An improved chaos sparrow search algorithm for UAV path planning. *Sci. Rep.* **2024**, *14*, 366. [CrossRef]
26. Jia, H.; Lu, C. Guided learning strategy: A novel update mechanism for metaheuristic algorithms design and improvement. *Knowl.-Based Syst.* **2024**, *286*, 111402. [CrossRef]
27. Hu, G.; Zhong, J.; Wei, G. SaCHBA_PDN: Modified honey badger algorithm with multi-strategy for UAV path planning. *Expert Syst. Appl.* **2023**, *223*, 119941. [CrossRef]
28. Lei, W.; He, Q.; Yang, L.; Jiao, H. Solar photovoltaic cell parameter identification based on improved honey badger algorithm. *Sustainability* **2022**, *14*, 8897. [CrossRef]
29. Nassef, A.M.; Houssein, E.H.; Helmy, B.E.d.; Rezk, H. Modified honey badger algorithm based global MPPT for triple-junction solar photovoltaic system under partial shading condition and global optimization. *Energy* **2022**, *254*, 124363. [CrossRef]
30. Han, E.; Ghadimi, N. Model identification of proton-exchange membrane fuel cells based on a hybrid convolutional neural network and extreme learning machine optimized by improved honey badger algorithm. *Sustain. Energy Technol. Assess.* **2022**, *52*, 102005. [CrossRef]
31. Deng, B. An improved honey badger algorithm by genetic algorithm and levy flight distribution for solving airline crew rostering problem. *IEEE Access* **2022**, *10*, 108075–108088. [CrossRef]
32. Dong, H.; Lin, G. Robust optimization design of fresh closed-loop supply chain network based on improved honey badger algorithm. *Appl. Res. Comput.* **2022**, *39*, 3020–3025.
33. Chai, Y.; Wang, R.; Ren, S. Improved honey badger algorithm for dual population collaborative evolution. *Appl. Res. Comput.* **2024**, *41*, 736–745+771.
34. Xiang, H.; Li, H.; Fu, X.; Su, X. Improved Honey Badger Algorithm Based on Multi-Strategy and Its Applications. *Comput. Eng.* **2023**, *49*, 78–87.
35. Hemeida, A.M.; Omer, A.S.; Bahaa-Eldin, A.M.; Alkhalaf, S.; Ahmed, M.; Senjyu, T.; El-Saady, G. Multi-objective multi-verse optimization of renewable energy sources-based micro-grid system: Real case. *Ain Shams Eng. J.* **2022**, *13*, 101543. [CrossRef]
36. Wang, H. Study on Optimal Dispatch of Microgrid Considering the Uncertainty of Source and Load Side. Ph.D. Thesis, Anhui University of Science and Technology, Huainan, China, 2023.
37. He, S.; Liu, T. Research on optimized scheduling of combined cooling heating and power microgrid based on improved butterfly algorithm. *Electr. Eng.* **2021**, *22*, 14–19+68.
38. Raghav, L.P.; Kumar, R.S.; Raju, D.K.; Singh, A.R. Analytic hierarchy process (AHP)–swarm intelligence based flexible demand response management of grid-connected microgrid. *Appl. Energy* **2022**, *306*, 118058. [CrossRef]
39. Li, Z.; Zhang, F.; Liang, J.; Yun, Z.; Zhang, X. Dynamic Scheduling of CCHP Type of Microgrid Considering Additional Opportunity Income. *Autom. Electr. Power Syst.* **2015**, *39*, 8–15.
40. Wang, Y.; Tang, L.; Yang, Y.; Sun, W.; Zhao, H. A stochastic-robust coordinated optimization model for CCHP micro-grid considering multi-energy operation and power trading with electricity markets under uncertainties. *Energy* **2020**, *198*, 117273. [CrossRef]
41. Kapner, D.J.; Cook, T.S.; Adelberger, E.G.; Gundlach, J.H.; Heckel, B.R.; Hoyle, C.; Swanson, H.E. Tests of the gravitational inverse-square law below the dark-energy length scale. *Phys. Rev. Lett.* **2007**, *98*, 021101. [CrossRef]
42. Goldanloo, M.J.; Gharehchopogh, F.S. A hybrid OBL-based firefly algorithm with symbiotic organisms search algorithm for solving continuous optimization problems. *J. Supercomput.* **2022**, *78*, 3998–4031. [CrossRef]
43. Tang, A.; Zhou, H.; Han, T.; Xie, L. A modified manta ray foraging optimization for global optimization problems. *IEEE Access* **2021**, *9*, 128702–128721.
44. Suganthan, P.N.; Hansen, N.; Liang, J.J.; Deb, K.; Chen, Y.P.; Auger, A.; Tiwari, S. Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization. *Nat. Comput.* **2005**, 341–357. Available online: https://www.researchgate.net/publication/235710019_Problem_Definitions_and_Evaluation_Criteria_for_the_CEC_2005_Special_Session_on_Real-Parameter_Optimization (accessed on 8 October 2024).
45. Shuai, H.; Fang, J.; Ai, X.; Tang, Y.; Wen, J.; He, H. Stochastic optimization of economic dispatch for microgrid based on approximate dynamic programming. *IEEE Trans. Smart Grid* **2018**, *10*, 2440–2452. [CrossRef]