applied sciences

MDPI

Article

A Hybrid Grey Wolf Optimization Algorithm Using Robust Learning Mechanism for Large Scale Economic Load Dispatch with Vale-Point Effect

Tzu-Ching Tai [1], Chen-Cheng Lee [2] and Cheng-Chien Kuo [2,*]

[1] Graduate Institute of Energy and Sustainability Technology, National Taiwan University of Science and Technology, Taipei 10607, Taiwan
[2] Department of Electrical Engineering, National Taiwan University of Science and Technology, Taipei 10607, Taiwan
* Correspondence: cckuo@mail.ntust.edu.tw

Abstract: This paper proposes a new hybrid algorithm for grey wolf optimization (GWO) integrated with a robust learning mechanism to solve the large-scale economic load dispatch (ELD) problem. The robust learning grey wolf optimization (RLGWO) algorithm imitates the hunting behavior and social hierarchy of grey wolves in nature and is reinforced by robust tolerance-based adjust searching direction and opposite-based learning. This technique could effectively prevent search agents from being trapped in local optima and also generate potential candidates to obtain a feasible solution. Several constraints of power generators, such as generation limits, local demand, valve-point loading effect, and transmission losses, are considered in practical operation. Five test systems are used to evaluate the effectiveness and robustness of the proposed algorithm in solving the ELD problem. The simulation results clearly reveal the superiority and feasibility of RLGWO to find better solutions in terms of fuel cost and computational efficiency when compared with the previous literature.

Keywords: hybrid algorithm; optimization; economic load dispatch; grey wolf optimization

Citation: Tai, T.-C.; Lee, C.-C.; Kuo, C.-C. A Hybrid Grey Wolf Optimization Algorithm Using Robust Learning Mechanism for Large Scale Economic Load Dispatch with Vale-Point Effect. *Appl. Sci.* **2023**, *13*, 2727. https://doi.org/10.3390/app13042727

Academic Editor: Giovanni Petrone

Received: 16 December 2022
Revised: 10 February 2023
Accepted: 14 February 2023
Published: 20 February 2023

1. Introduction

Economic load dispatch (ELD) is considered the most elementary and computationally heavy optimization problem in electricity industry studies. ELD enables the power system to analyze the committed generating units to allocate the optimum level of active power output to achieve the total power demands at the cheapest operating cost while satisfying several physical and operational constraints [1]. Over the last few years, various mathematical programming techniques and optimization methods have been applied to solve ELD problems, such as the Newton method [2], the gradient method [3], the base point and participation factors method, and the Lambda-iteration method [2]. Nevertheless, none of the mentioned techniques perform well in addressing practical problems with nonlinear and discontinuous characteristics since these methods required the incremental cost of the generators to be increase in a monotonic or piecewise linear fashion [2]. Therefore, nonlinear programming [4], dynamic programming [5], and some of their modified applications have been employed to deal with ELD issues. Unfortunately, the drawback of these methods is their computational time, which when applied to modern power systems with a huge number of generating units is too long.

To overcome the downside of traditional mathematical programming, researchers around the world have introduced metaheuristic optimization algorithms, such as simulated annealing (SA) [6], genetic algorithms (GAs) [7], tabu searches (TS) [8], and artificial neural networks (ANNs) [9]. Yalcinoz [10] successfully addressed complex optimization issues. However, these probabilistic heuristic algorithms do not always guarantee the global search property. Recently, different kinds of global optimization algorithms have been proposed, such as particle swarm optimization (PSO) [11], modified particle swarm optimization [12], biogeography-based optimization (BBO) [13], modified genetic algorithm [14], bacteria foraging optimization [15], differential evolution (DE) [16], and ant colony optimization [17]. The application of these techniques in ELD has delivered some promising solutions in terms of minimizing total generation cost and improving convergence rate.

However, recent research has recognized a few deficiencies in stochastic algorithms like GA. This degradation in efficiency and limited search capability may be noticeable when applied to multimodal objective functions. Additionally, TS requires a suitable selection of control parameters to attain an optimal solution. Slow convergence can be considered a key disadvantage of GA, SA, TS, and ANN, making them unable to appropriately address real-time issues. Even though PSO has attracted research interest due to its rapid convergence characteristic and its flexibility, PSO is still restricted when applied to large-scale real-time ELD, since it is not always guaranteed that the total generation cost is the global best solution.

The modification of the presented algorithm can be a solution for achieving the global optimal solution and upgrading the convergence rate. Existing modified metaheuristic methods include chaotic differential evolution and sequential quadratic programming (DEC_SQP) [18], improved coordinated aggregation-based PSO (ICA_PSO) [19], quantum-inspired particle swarm optimization (QPSO) [20], modified shuffled frog leaping algorithm with GA crossover (MSFLA and GA) [21], a different version of PSO [19], shuffled differential evolution (SDE) [21], hybrid biogeography-based optimization with differential evolution (DE/BBO) [22], and adaptive hybrid backtracking search optimization [23]. Oppositional invasive weed optimization (OIWO) [24] improves the convergence rate of invasive weed optimization by incorporating opposite-based learning (OBL) [25]. Nevertheless, this OIWO relies heavily on the initial selection of control parameters to obtain the global best solution. Moreover, the solution is not unique for every trial, and this method also has the problem of long computational time.

In recent years, a new evolutionary optimization technique, called grey wolf optimization (GWO), which mimics the social hierarchy and hunting behavior of grey wolves, has been proposed by Mirjalili et al. [26]. There are several applications of GWO in power system optimization problems [27,28]. The outstanding characteristic of GWO compared with other stochastic heuristic algorithms is that it does not depend on accurate initialization of input parameters to obtain the global best solution. However, the parameters still need to be modified to avoid premature convergence, as well as to improve the convergence speed. To overcome the above limitation, a hybrid GWO algorithm with a robust learning mechanism (RLGWO) is introduced in this paper that incorporates opposite-based learning (OBL) as a candidate generation strategy. The main reason for selecting OBL is that it does not require any specific technique to speed up the convergence rate of different optimization algorithm. Additionally, the candidates generated by OBL are more likely to come closer to the global optimum solution than a solution using randomly generated candidates, because the technique simultaneously considers both the current population and its opposite. In only a short period of time, OBL, a new concept in computational intelligence, has attracted research attention with the aim of enhancing metaheuristic optimization algorithms to address large-scale ELD [24,29].

In this paper, the performance of a proposed grey wolf algorithm variant with robust tolerance-based adjust searching direction mechanism (RTASDM), called the robust learning grey wolf algorithm (RLGWO), is studied. The algorithm is tested on five test systems with different sizes and in consideration of power system constraints to evaluate the performance of the proposed approach in solving the ELD problem compared with other variants of the GWO. The reported results reveal the ability of RLGWO to archive superior solutions in terms of quality, consistency, and convergence rate compared to several other optimization methods.

2. Problem Formulation

Economic load dispatch is a classical optimization problem in power systems. The main goal is to obtain the cheapest fuel cost by allocating the optimal power output among the available generating units while satisfying various equality and inequality constraints.

2.1. Objective Function

The main objective of the ED problem is to minimize the total generation costs of a power system by determining the power output of generators that satisfies various constraints, such as the load demands of PD are met within an appropriate period (normally in one hour), the active balance constraint, and the limitation of the generator. The simplified cost function of ED can be written as:

$$\text{Minimize } F_{cost} = \sum_{i=1}^{g} F_i(P_i) \tag{1}$$

$$F_i(P_i) = a_i + b_i P_i + c_i P_i^2 \tag{2}$$

where F_{cost} is the total generation cost, F_i is the cost function of the ith generating unit in terms of P_i, which is the power output of ith generating unit. a_i, b_i and c_i are the fuel cost coefficients of ith generating unit. Finally, g is the total number of generating units.

Practically, the generators with multiple valve steam turbines exhibit a wide variation with respect to the fuel cost function. This is because the valve-point creates a ripple, whereby the cost function cannot be represented by a quadratic polynomial function as in (2), since it contains nonlinear characteristics. Therefore, the cost function considered in this paper is a combination of quadratic functions and sinusoidal functions, which are represented as follows:

$$\text{Minimize } F_{cost} = \sum_{i=1}^{g} F_i(P_i) = \sum_{i=1}^{g} \left(a_i + b_i P_i + c_i P_i^2 + \left| e_i \sin(f_i \times (P_{i,min} - P_i)) \right| \right) \tag{3}$$

where e_i and f_i are the constants of the ith generating unit reflecting the valve-point effect [30].

Additionally, thermal dispatching units are realistically supplied from different fuel sources with several types of fuel. The operation of each dispatching unit should be mathematically represented under multiple piecewise functions to emphasize the effect of multiple fuel sources. Both the valve-point effect and multiple fuel options should be combined to acquire a practical and reliable ELD. Finally, the cost function of the ith generating unit can be realistically expressed as below:

$$F_i(P_i) = \begin{cases} a_{i1} + b_{i1} P_{i1} + c_{i1} P_{i1}^2 + \left| e_{i1} \sin(f_{i1} \times (P_{i1,min} - P_{i1})) \right| & if\, P_{i1,min} \le P_i \le P_{i1} \\ a_{i2} + b_{i2} P_{i2} + c_{i2} P_{i2}^2 + \left| e_{i2} \sin(f_{i2} \times (P_{i2,min} - P_{i2})) \right| & if\, P_{i2,min} \le P_i \le P_{i2} \\ \cdots & \cdots \\ a_{ik} + b_{ik} P_{ik} + c_{ik} P_{ik}^2 + \left| e_{ik} \sin(f_{ik} \times (P_{ik,min} - P_{ik})) \right| & if\, P_{ik,min} \le P_i \le P_{ik} \end{cases} \tag{4}$$

2.2. Equality and Inequality Constraints

(a) Active power balance constraint The total generated power output $\sum_{i=1}^{g} P_i$ should be the same as the sum of the total power demand P_D and the total system loss P_{Loss}. This is represented as follows:

$$\sum_{i=1}^{g} P_i = P_D + P_{Loss} \tag{5}$$

(b) The limitations of the generator The power output of each generating unit should be limited to between its minimum and maximum power outputs, as shown in the inequality constraint in (6):

$$P_{i,min} \le P_i \le P_{i,max} \tag{6}$$

where $P_{i,min}$ and $P_{i,max}$ are the minimum and maximum real power output of the generating unit i.

3. GWO Algorithm

The Grey Wolf Optimizer (GWO) was modeled by Mirjalili et al. [26] based on the social hierarchy and the hunting behavior of grey wolves.

3.1. Social Hierarchy

The grey wolf belongs to the Canidae family, and they are considered to be apex predators, which means that they are at the top of the food chain. They mostly prefer to live in packs, and they have a very strict social dominance hierarchy, consisting of alpha, beta, delta, and omega [26].

To mathematically model the GWO based on the social hierarchy characteristics of wolves, the alpha (α) is considered to be the fittest solution. The beta (β) and the delta (δ) are consequently utilized as the second- and third-best solutions, respectively. The omega (ω) corresponds to the remainder of the candidate solutions. The hunting (optimization) in the GWO algorithm is guided by α, β, and δ. The ω follows these three leaders.

3.2. Encircling Prey

Group hunting is another interesting social behavior that emphasizes the social hierarchy of grey wolves, as described above. According to Muro et al. [31], the hunting process of grey wolves includes three main steps: (1) tracking, chasing, and approaching the prey; (2) encircling, pursuing, and harassing the prey until it stops moving; and (3) attacking the prey. The following equations show the mathematical modeling of encircling behavior [26]:

$$\vec{D} = \left| \vec{C} \cdot \vec{X}_p(t) - \vec{X}(t) \right| \tag{7}$$

$$\vec{X}(t+1) = \vec{X}_p(t) - \vec{A} \cdot \vec{D} \tag{8}$$

where $\vec{X}_p$ is the position vector of the prey, $\vec{X}$ is the position vector of the grey wolf, t indicates the current iteration, $\vec{A}$ and $\vec{C}$ are coefficient vectors.

The vectors $\vec{A}$ and $\vec{C}$ are calculated as follows:

$$\vec{A} = 2\vec{a} \cdot \vec{r}_1 - \vec{a} \tag{9}$$

$$\vec{C} = 2\vec{r}_2 \tag{10}$$

where $\vec{a}$ decreases linearly from 2 to 0 over the course of multiple iterations, and $\vec{r}_1$, $\vec{r}_2$ are random vectors [0,1].

3.3. Hunting

The alpha, beta, and delta are assumed to have the ability to recognize potential locations of prey, since they hold the three best solutions obtained so far. Hence, their solution positions are utilized to update the positions of all of the other (omega) wolves. The updated position formulation proposed above is as follows:

$$\vec{D}_\alpha = \left| \vec{C}_1 \cdot \vec{X}_\alpha - \vec{X} \right|, \vec{D}_\beta = \left| \vec{C}_2 \cdot \vec{X}_\beta - \vec{X} \right|, \ \vec{D}_\delta = \left| \vec{C}_3 \cdot \vec{X}_\delta - \vec{X} \right| \tag{11}$$

$$\vec{X}_1 = \vec{X}_\alpha - \vec{A}_1 \cdot \vec{D}_\alpha, \ \vec{X}_2 = \vec{X}_\beta - \vec{A}_2 \cdot \vec{D}_\beta, \ \vec{X}_3 = \vec{X}_\beta - \vec{A}_3 \cdot \vec{D}_\beta \tag{12}$$

$$\vec{X}(t+1) = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3} \tag{13}$$

After repeatedly applying the encircling and hunting techniques, the prey (the fittest solution) will be located.

4. Robust Learning-Based GWO Algorithm (RLGWO)

All omega members of the hunting group in the original GWO learn from the first three best leaders to update their position until the termination condition is reached, even if the fittest solution (alpha) is trapped in a local optimum. This kind of learning algorithm can work well in the exploitation phase and has the ability to converge rapidly, but is invalid when solving problems with such large and complex search spaces. Some methods have been proposed for GWO incorporating strategies to restrict the learning mechanism of omega to maintain the diversity of the population, like EEGWO [32] and GWO-ABC [33]. These strategies result in good exploration performance. However, it takes a longer time to reach a solution and slows down the convergence rate of the algorithm.

The algorithm (RLGWO) proposed in this paper achieves a balance between exploitation and exploration. In RLGWO, a robust tolerance-based adjust searching direction mechanism is used. This method gives omega the ability to adjust their search direction to avoid falling into local optima and to diminish the size of the search space. Additionally, an opposition learning-based candidate grey wolf strategy is utilized to generate candidate leaders that replace the position of alpha as well as beta and delta, in which the hunting group can perform in different areas within the search space. Subsequently, looking toward guaranteeing the efficiency and accuracy of the algorithm, a potential position update scheme is introduced to adapt the potential ability of the candidate leader to guide the grey wolf for exploitation in divergent dimensions.

4.1. Robust Tolerance-Based Adjust Searching Direction Mechanism (RTASDM)

It is widely known that the grey wolf in the original GWO had a high possibility of getting stuck in local optima when operating in a large and complex search space. In Figure 1a, the hunting mechanism for a one-dimensional problem is presented, where the blue curve illustrates the object function. P_α is the alpha position (the best solution) that leads the rest of the population. In Figure 1a, it is clear that each omega can move along the guiding direction of the alpha fitness value P_α and the hunting group will be trapped in a local optimum after several iteration. Assume $f(X_\omega^n)^k$ indicates the objective value of the nth omega in the population at the kth iteration. X_ω^n will be updated in the next iteration using Formula (13). Then, it will generate a new fitness denoted by $f(X_\omega^n)^{k+1}$. The total difference between $f(X_\omega^n)^{k+1}$ and $f(X_\omega^n)^k$ in the population can be described as shown in (14), where N is the population size.

$$\sum_{n=1}^{N}\left(f(X_\omega^n)^k\right) - \sum_{n=1}^{N}\left(f(X_\omega^n)^{k+1}\right) = A\ (A \in \mathbb{R}) \tag{14}$$

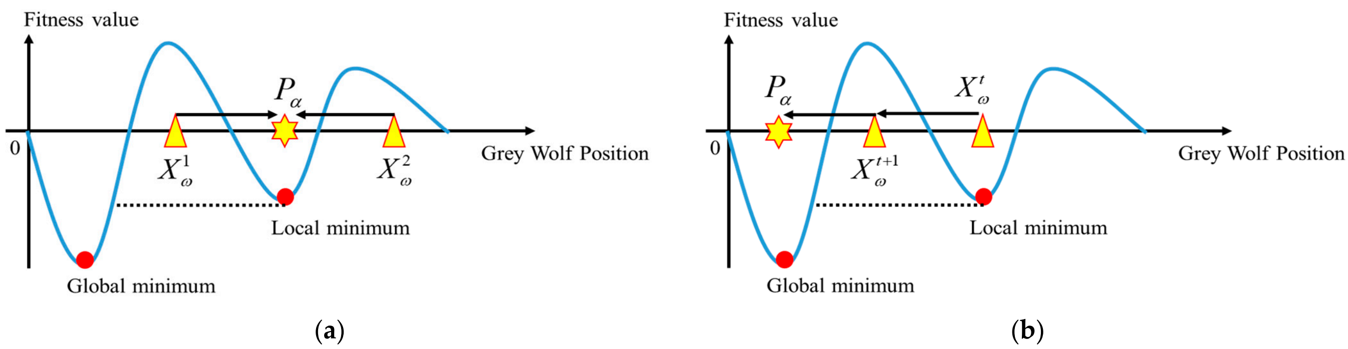


Figure 1. The hunting mechanism of GWO in one-dimensional minimization. (**a**) P_α falls into a local optimum; and (**b**) P_α reaches the global optimum.

After the course of iteration, the solution may finally converge to the optimum solution (global or local optimum). This situation indicates that A is more likely to be close to 0. Assuming that φ is a small value around 0, we can determine when the hunting group is going to converge by setting A that belongs to a range of $[-\varphi, \varphi]$. Therefore, Formula (14) can be rewritten as:

$$\sum_{n=1}^{N}\left(f(X_{\omega}^{n})^{k}\right)-\sum_{n=1}^{N}\left(f(X_{\omega}^{n})^{k+1}\right)\in[-\varphi,\varphi] \tag{15}$$

To avoid the hunting groups getting stuck in the local optimum and to ensure the efficiency of the algorithm, the omega's search direction can be adjusted every time Equation (15) is satisfied. Nevertheless, when dealing with a large and complex space, we cannot rely on the circumstances described above to change the search direction of the omega. It can be clearly seen in Figure 1b that when the alpha position (global best solution) gets close to the global optimum, it leads the other members of the hunting group to search in the direction towards P_{α}. This also means that the solution of the generation $k+1$ may not be improved by the alpha, while also satisfying the conditions shown in Equation (15). Therefore, depending on the potential ability of P_{α}, the hunting group can be led to a promising global optimum over the next few iterations [34].

After several iterations, as the number of satisfied conditions increases, meaning that there is no difference between the current solution and previous solutions, it can be concluded that the grey wolf is stuck around a local optimum. Therefore, the grey wolf needs to modify its search direction. Let h be a tolerance variable, where h is initially set to 0 and is used as a counter. In cases where (15) is satisfied, h can be updated via Equation (16), as follows.

$$h=h+1 \tag{16}$$

As the value of h becomes greater, the probability of becoming trapped in the local optima of a hunting group also increases [34]. However, in cases where P_{α} is searching around the global optimum, as described above, the omega should not change their search direction, but rather move along the alpha direction. Therefore, we introduce the probability P_{adjust}, which enables the omega to adjust its search direction. P_{adjust} can be achieved experimentally using Equation (17), below, where k is the current iteration and $MaxIt$ is the maximum number of iterations.

$$P_{adjust}=\frac{exp(h)-1}{exp\left(10+\frac{k\times10}{MaxIt}\right)-1} \tag{17}$$

As can be seen from Figure 2, the P_{adjust} is not fixed throughout the course of iterations; rather, its value is updated according to h and k. When the value of P_{adjust} is larger than a random number with a range of [0,1], the leader alpha (beta or gamma) will be replaced by another candidate solution, which will continue to guide the hunting group.

Algorithm 1, below, shows the details of the approach.

It is clear from Figure 2 and Algorithm 1 that the P_{adjust} depends on h and k, especially the tolerance value h. When the value of h is small, P_{adjust} has a high probability of becoming smaller than the random number; then, the first three best leaders continue to guide the hunting group, and their leading ability will be useful for the next several iterations. When h increases to the threshold, P_{adjust} increases dramatically. This indicates that the number of solutions has still not improved over the next several iterations, and the hunting group is more likely to become trapped in a local optimum. Therefore, the value of P_{adjust} is probably greater than the random number, and a new leader will be used to lead the omega search direction.

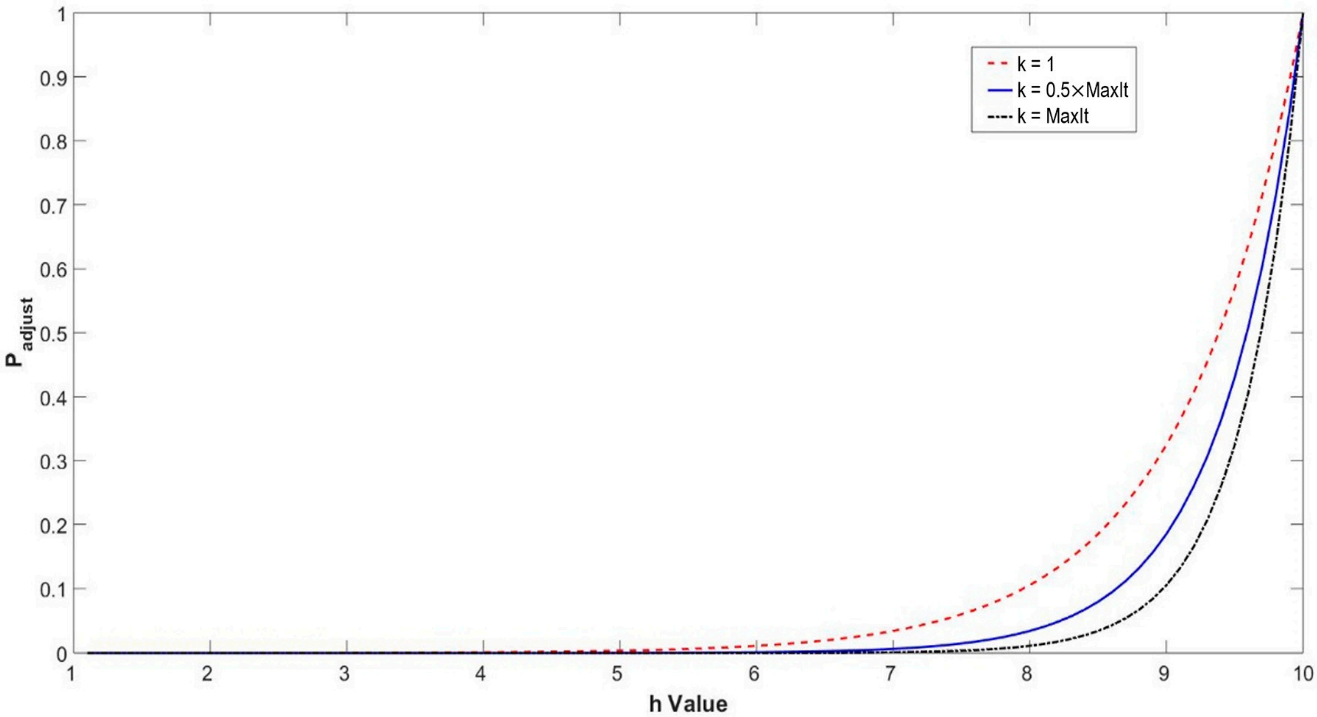


Figure 2. Adjustment probability at different iteration periods.

Algorithm 1. Robust tolerance-based adjust searching direction mechanism
1: At iteration kth;
$2:$ Initialize $h = 0$
$3:$ if $\sum_{n=1}^{N}\left(f(X_{\omega}^{n})^{k}\right) - \sum_{n=1}^{N}\left(f(X_{\omega}^{n})^{k+1}\right) \in [-\varphi, \varphi]$ then
$4:\ h = h + 1$
5: end if
6: generate a random number in the range of [0,1];
$7:$ if $\left(\frac{exp(h)-1}{exp\left(10+\frac{k\times 10}{MaxIt}\right)-1} > rand()\right)$ then
8: Finding another candidate solution to replace alpha (beta or gamma)
9: end if

On top of that, P_{α} is more likely to get close to the global optimum when the number of iterations is increased, especially with numbers of iterations greater than half, as can be seen in Figure 3. Therefore, to ensure convergence, the value of h should be increased to get rid of the omega and change its search direction.

4.2. Opposition-Based Learning for Candidate Generation Strategy

Opposition-based learning (OBL) [25] has recently been utilized to accelerate the convergence rate of several optimization algorithms. The OBL technique can be used to generate potential candidate solutions by considering both the current population and its opposite population. It has been proved in many studies worldwide that opposition candidate solutions are more likely to get closer to the global optimum solution than a randomly generated candidate solution. There have been many advanced applications of this learning mechanism in several soft computing techniques, as reported in [35–38].

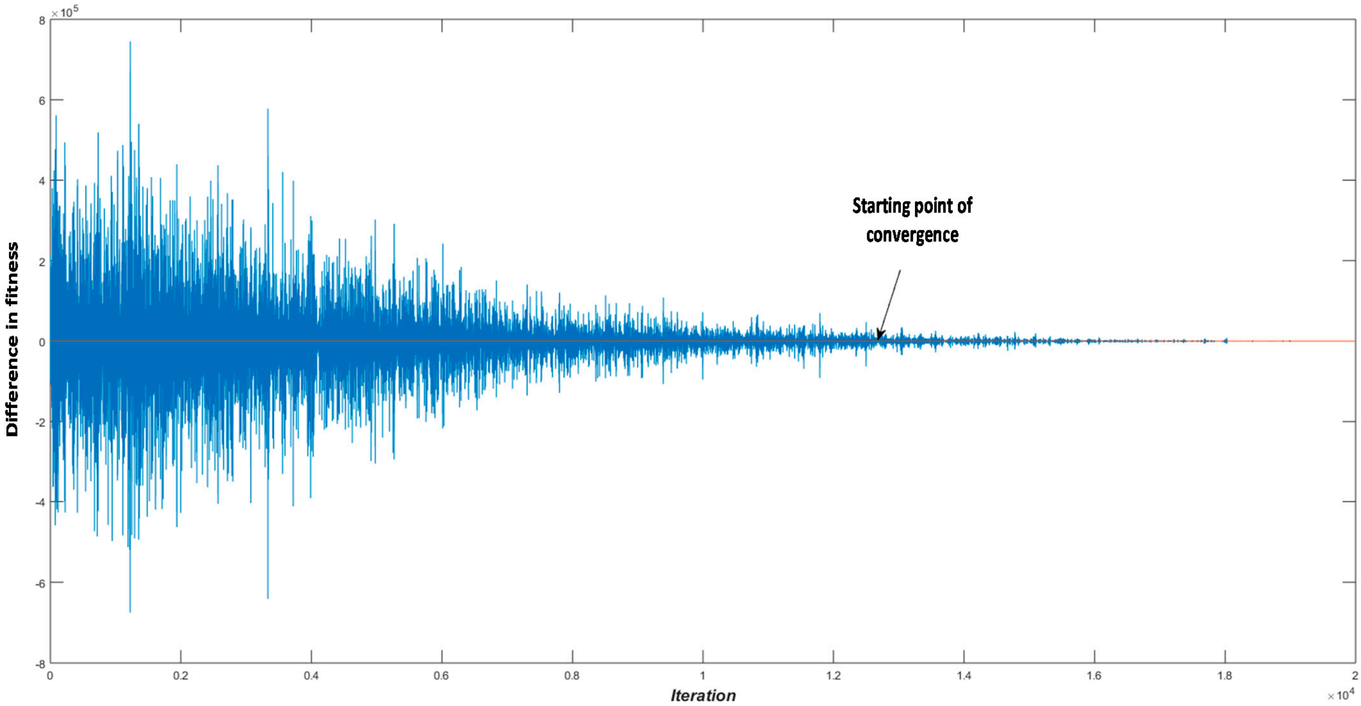


Figure 3. The convergence rate of the search mechanism.

The two definitions below show the important aspects of OBL, the opposite number and opposite point [25]:

Definition— Let $x \in \mathbb{R}$ be a real number defined in a certain interval: $x \in [a, b]$. The opposite number x^{opp} is defined as follows:

$$x^{opp} = a + b - x \tag{18}$$

Definition— Let $P(x_1, x_2, \ldots, x_n)$ be a point in an n-dimensional coordinate system with $x_1, x_2, \ldots, x_n \in \mathbb{R}$ and $x \in [a_i, b_i]$. The opposition point P^{opp} is completely defined by its coordinates $x_1^{opp}, x_2^{opp}, \ldots,\ x_n^{opp}$ where

$$x_1^{opp} = a_i + b_i - x_i\ i = 1,\ 2, \ldots, n \tag{19}$$

In the proposed RLGWO, after defining the replacement of the leader with a candidate solution in Section 4.1, a random candidate solution can easily be created in the search space to guide the hunting group to get rid of the current local optimum solution. However, the random candidate may not be guaranteed to improve the solution; in particular, when dealing with large and complex spaces, it is probable that it will lead the hunting group into another local optimum. Using the OBL can ensure the generation of a candidate more effectively.

In Figure 3, it is clear to see that the difference between the current and the previous solution in the first half of the iteration changes violently. This circumstance can be explained by the fact that the hunting group is carrying out the exploration phase. This means that the grey wolves are attempting to search the hold space to figure out promising areas for the global optimum. The alpha will have a significant influence on the hunting group, since it is the best solution. Therefore, in this certain period, replacing the alpha with its opposite position is a wise course of action to escape the local optimum.

The remaining half of the total course of iteration is the exploitation phase, when the grey wolf scales down the search space and concentrates on a certain area to find the optimum solution. To prevent the omega from moving away from the global optimum and to ensure the efficiency of the algorithm, the alpha acts as the main leader and the beta (or gamma) may be replaced by its opposite position. Since the beta and gamma have almost the same potential ability, the beta will be removed if the random number is greater than 0.5, and conversely, gamma will be substituted if it is less than this value.

The details of the generating candidate are described in Algorithm 2.

Algorithm 2. Opposition-based learning for candidate generation strategy

1: At iteration kth;
2 : Generate a random number $rand()$ between [0,1];
3 : if $P_{adjust} < rand()$
4: Keep alpha, beta, and gamma as the leader;
5: else
6 : if kth $< \frac{MaxIt}{2}$ then
7 : $X_\alpha \leftarrow \min(alpha) + max(alpha) - X_\alpha$;
8: else
9 : Generate a random number $rand()$ between [0,1];
10 : if $rand() > 0.5$
11 : $X_\beta \leftarrow \min(beta) + max(beta) - X_\beta$;
12: else
13 : $X_\delta \leftarrow \min(gamma) + max(gamma) - X_\delta$;

4.3. RLGWO Algorithm for Economic Load Dispatch

The computational mechanism for the proposed RLGWO algorithm to solve ELD problems is described in the following steps.

Step 1. Initialization

Step 1.1: Arbitrarily generate the initial value for all of the active power of the generating units belonging to their lower and upper real power operating limits except the last unit. Equation (4) is used to compute the amount of active power output of the last unit to guarantee whether it satisfies the inequality constraint or not. The solution will be discarded whenever it violates the inequality constraint. Let D be the dimension of the hunting group. The initial position of the grey wolves is given as a matrix, X, below:

$$X = \begin{bmatrix} X_1^1 & X_2^1 & \cdots & X_i^1 & \cdots & X_g^1 \\ X_1^2 & X_1^2 & \cdots & X_1^2 & \cdots & X_g^2 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ X_1^i & X_2^i & \cdots & X_i^i & \cdots & X_g^i \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ X_1^D & X_2^D & \cdots & X_i^D & \cdots & X_g^D \end{bmatrix} \tag{20}$$

Step 1.2: Substitute the matrix X into (3) and calculate the fuel cost for each solution of the current population.

Step 1.3: Evaluate the cost value of all search agents (grey wolves), which determine X_α, X_β, X_δ as the first three best solutions by simple comparison of their cost value.

Step 1.4: Set all the parameters, including $MaxIt$, $\varphi = 0$, coefficient vector $\vec{A}$ and $\vec{C}$, initial variable tolerance $h = 0$.

Step 2. Repeat this step until the stopping criterion is satisfied.

Step 2.1: Update the position of all search agents using (11)–(13).

Step 2.2: Calculate the fuel cost for all members of the hunting group using (3) and compare the results to figure out X_α, X_β, X_δ.

Step 2.3: Evaluate $A = \sum_{n=1}^{N} \left(f(X_\omega^n)^k \right) - \sum_{n=1}^{N} \left(f(X_\omega^n)^{k+1} \right)$, and then check whether A fulfills Condition (15) or not. In cases where A belongs to the range $[-\varphi, \varphi]$, h will be increased by 1.

Step 2.4: Compute the probability of P_{adjust} using (17) and select a random number $R \in [0, 1]$. If $P_{adjust} < R$, meaning that the three best leaders still potentially have the ability to guide all search agents, then return to Step 2. Otherwise, follow Algorithm 2 to generate a suitable candidate to lead the hunting group and move back to Step 2.

5. Simulation Results and Discussion

Five different benchmark test systems were applied to evaluate the performance of the proposed RLGWO algorithm. Its performance was also compared with that of several optimization techniques reported in the literature. The nature-inspired RLGWO was executed using MATLAB 7.1 (R2010a) on Intel (R) 1GB RAM, 2.60 GHz CPU.

Dissimilar to other metaheuristic computations that require suitable values for the algorithm's input parameters in order to improve their convergence rate, like particle swarm optimization (PSO) or evolutionary algorithms (ED), GWO has the advantage of being free from the initialization of input parameter. In this paper, the initial parameter for population size and the maximum number of iterations is selected as 50 and 200, respectively.

5.1. Test Systems and Results

5.1.1. Test System 1

This system contained 13 thermal units with a fuel cost function having a valve-point effect is used. The power demand, in this case, it is assumed to be 2520 MW, including transmission losses. The detailed parameters of the system were adopted from [24]. The solution obtained from the proposed RLGWO is compared with oppositional real coded chemical reaction optimization (ORCCRO) [29], a different version of PSO [19], shuffled differential evolution (SDE) [39], biogeography-based optimization (BBO) [13], DE/BBO [22], and OIWO [24], where Table 1 shows their best results. The convergence characteristic of the 13-unit test systems with the original GWO and RLGWO algorithms for fitness value are presented in Figure 4. This shows that the result of RLGWO is lower than the result of GWO.

Table 1. Best simulation results of the 13-unit system with loss ($P_D = 2520$ MW).

Unit Number	Methods			
	RLGWO	**SDE**	**OIWO**	**ORCCRO**
1	628.3112	628.32	628.3185	628.32
2	299.1788	299.2	299.1989	299.2
3	299.1798	299.2	299.1991	299.2
4	159.7297	159.73	159.7331	159.73
5	159.7297	159.73	159.7331	159.73
6	159.7297	159.73	159.7331	159.73
7	159.7284	159.73	159.733	159.73
8	159.7297	159.73	159.7331	159.73
9	159.7284	159.73	159.733	159.73
10	77.3839	77.4	77.3953	77.4
11	113.1011	113.12	113.1079	112.14
12	92.3254	92.4	92.3594	92.4
13	92.3891	92.4	92.3911	92.4
Total generating output	2560.2446	2560.44	2560.3686	2559.44
Power loss	40.2446	40.43	40.3686	39.43
Fuel cost	24,514.78	24,514.95	24,514.83	24,513.99

The statistical results of RLGWO and other published algorithms are listed in Table 2, showing the maximum, minimum, and average cost obtained using different methods over 50 trials. Figure 5 shows a visualization of the results presented in Table 2 of the different algorithms. The box graphs in Figure 5a show that the RLGWO always produces one of the lowest results for this problem. The ICA-PSO, on the other hand, has result statistics that are scattered over a wider range of cost. Figure 5b shows the time and number of hits of the minimum result. The RLGWO computational time is one of the fastest, with one of the highest number of hits to its minimum solution.

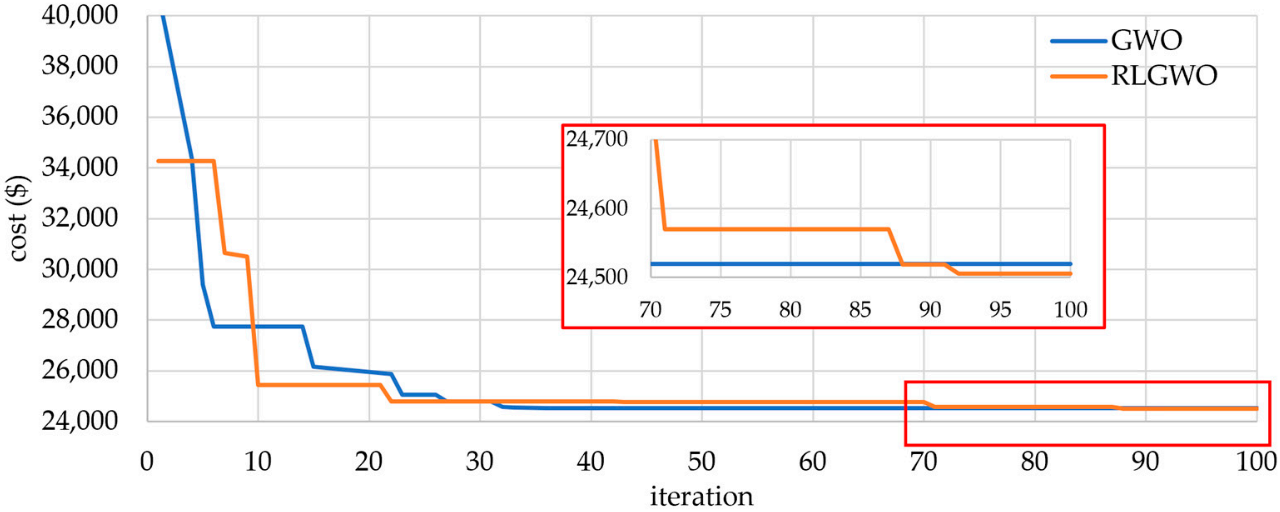


Figure 4. Comparative convergence characteristic of original GWO and proposed RLGWO algorithm for the 13-unit system.

Table 2. Comparison between different methods taken after 50 trials (13-unit system with loss).

Method	Generation Cost			Time (s)	No. of Hits of the Minimum Solution
	Max	Min	Average		
RLGWO	24,514.78	24,514.78	24,514.78	3.7	50
ORCCRO	24,513.91	24,513.91	24,513.91	8	50
OIWO	24,514.83	24,514.83	24,514.83	5.3	47
SDE	NA	24,514.88	24,516.31	NA	NA
DE/BBO	24,515.98	24,514.97	24,515.05	11	46
BBO	24,516.09	24,515.21	24,515.32	15	44
ICA-PSO	24,589.45	24,540.06	24,561.46	10.4	NA

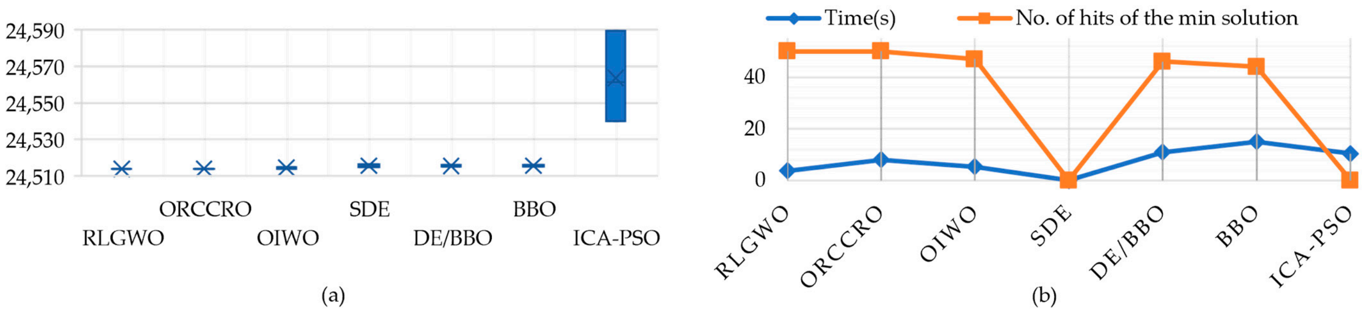


Figure 5. Results for different methods for the 13-unit system (50 trials): (**a**) boxplot of the final generation cost; (**b**) time and number of hits of the minimum solution.

5.1.2. Test System 2

The medium-size ELD problem with 40 thermal units with the valve-point loading effect and transmission loss is considered. The total power demand is 10,500 MW, with input data taken from [24]. The total power output and fuel cost were determined for the 40-unit systems using several metaheuristic techniques, including OIWO [24], BBO [13], DE/BBO [22], SDE [39], and GAAPI [40], alogn with the proposed RLGWO, and the results are given in Table 3. A comparison of the convergence characteristics of the 40-unit system between the original GWO and RLGWO algorithms with respect to fitness value is shown in Figure 6. The magnified results in Figure 6 show that RLGWO obtained a lower cost for this system.

Table 3. Best simulation results for the 40-unit system with loss ($P_D = 10,500$ MW).

Unit Number	Methods						
	RLGWO	**OIWO**	**DE/BBO**	**ORCCRO**	**BBO**	**SDE**	**GAAPI**
1	113.9954	113.9908	111.04	111.68	112.54	110.06	114
2	114	114	113.71	112.16	113.22	112.41	114
3	119.99885	119.9977	118.64	119.98	119.51	120	120
4	183.0428	182.5131	189.49	182.18	188.37	188.72	190
5	88.1189	88.4227	86.32	87.28	90.41	85.91	97
6	140	140	139.88	139.85	139.05	140	140
7	299.99995	299.9999	299.86	298.15	294.97	250.19	300
8	296.0327	292.0654	285.42	286.89	299.18	290.68	300
9	299.94085	299.8817	296.29	293.38	296.46	300	300
10	279.7137	279.7073	285.07	279.34	279.89	282.01	205.25
11	197.0559	168.8149	164.69	162.35	160.15	180.82	226.3
12	94.08905	94	94	94.12	96.74	168.74	204.72
13	484.1729	484.0758	486.3	486.44	484.04	469.96	346.48
14	484.19005	484.0477	480.7	487.02	483.32	484.17	434.32
15	484.044	484.0396	480.66	483.39	483.77	487.73	431.34
16	484.08385	484.0886	485.05	484.51	483.3	482.3	440.22
17	489.248	489.2813	487.94	494.22	490.83	499.64	500
18	489.27865	489.2966	491.09	489.48	492.19	411.32	500
19	511.328	511.3219	511.79	512.2	511.28	510.47	550
20	511.41705	511.335	544.89	513.13	521.55	542.04	550
21	536.7086	549.9412	528.92	543.85	526.42	544.81	550
22	548.3222	549.9999	540.58	548	538.3	550	550
23	523.33305	523.2804	524.98	521.21	534.74	550	550
24	523.32785	523.3213	524.12	525.01	521.2	528.16	550
25	523.4937	523.5804	534.49	529.84	526.14	524.16	550
26	523.44335	523.5847	529.15	540.04	544.43	539.1	550
27	10.0081	10.0086	10.51	12.59	11.51	10	11.44
28	10.0086	10.0068	10	10.06	10.21	10.37	11.56
29	10.03725	10.0123	10	10.79	10.71	10	11.42
30	87.83375	87.8664	90.06	89.7	88.28	96.1	97
31	190	190	189.82	189.59	189.84	185.33	190
32	189.99915	189.9983	187.69	189.96	189.94	189.54	190
33	190	190	189.97	187.61	189.13	189.96	190
34	199.997	199.994	199.83	198.91	198.07	199.9	200
35	200	200	199.93	199.98	199.92	196.25	200
36	164.86345	164.8283	163.03	165.68	194.35	185.85	200
37	110	110	109.85	109.98	109.43	109.72	110
38	109.997	109.994	109.26	109.82	109.56	110	110
39	110	110	109.6	109.88	109.62	95.71	110
40	530.92635	550	543.23	548.5	527.82	532.43	550
Total generating output	11,456.05	11,457.2965	11,457.83	11,458.75	11,470	11,474.43	11,545.06
Power loss	956.05	957.2965	957.83	958.75	970.37	974.43	1045.06
Fuel cost	136,548.3499	136,452.677	136,950.77	136,855.19	137,026.82	138,157.46	139,864.96

Table 4 presents the maximum, minimum, and average fuel cost obtained using the RLGWO, DE/BBO, SDE, OIWO, and GAAPI methods over 50 trials. These results are visualized in Figure 7. The OIWO produced the lowest and densest results of the five algorithms, as seen in Figure 7a. The RLGWO was second in terms of identifying the lowest cost, but was still always able to identify the lowest cost. As seen in Figure 7b, RLGWO and ORCCRO found the minimum result 50 times among 50 trials, but the solution found by ORCCRO was higher than that determined using RLGWO. Furthermore, RLGWO found the solution the fastest among the five algorithms.

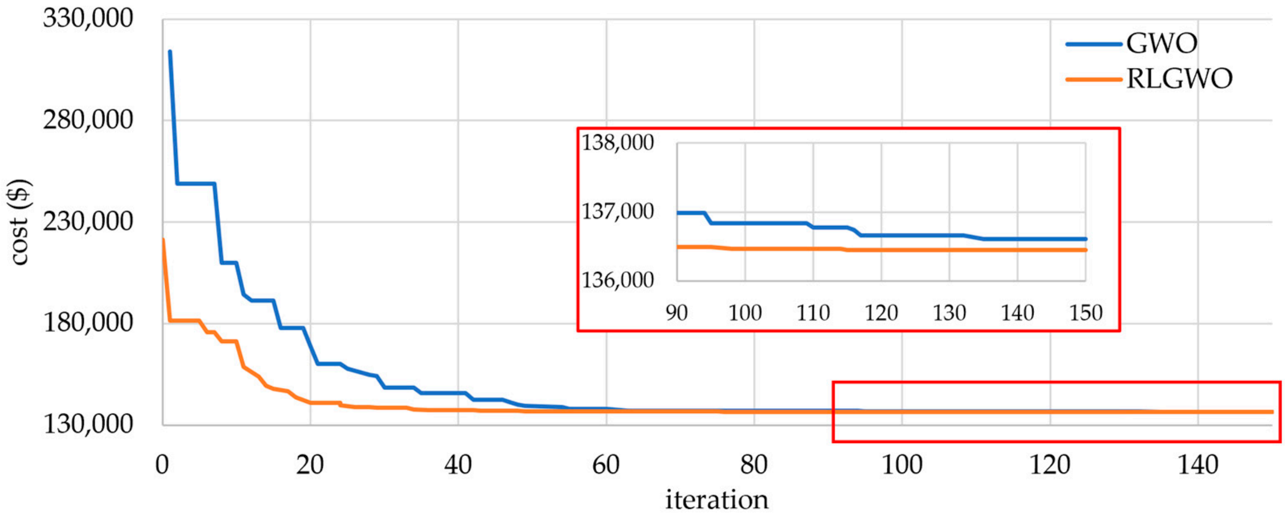
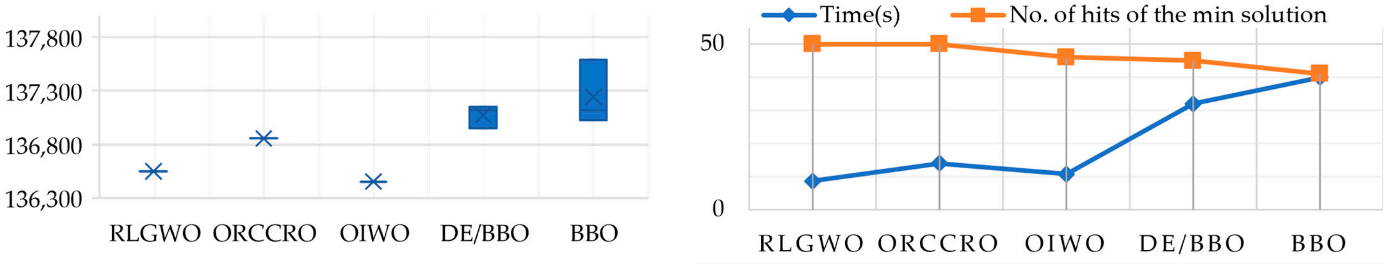


Figure 6. Comparative convergence characteristics of the original GWO and the proposed RLGWO algorithms for the 40-unit system.

Table 4. Comparison between different methods taken after 50 trials (40-unit system with loss).

Method	Generation Costs			Time (s)	No. of Hits of the Minimum Solution
	Max	Min	Average		
RLGWO	136,548.3499	136,548.3499	136,548.3499	8.6	50
ORCCRO	136,855.19	136,855.19	136,855.19	14	50
OIWO	136,452.677	136,452.677	136,452.677	10.7	46
DE/BBO	137,150.77	136,950.77	137,116.58	32	45
BBO	137,587.82	137,026.82	137,116.58	40	41



(a) (b)

Figure 7. Results for different methods for the 40-unit system (50 trials): (**a**) boxplot of the final generation cost; (**b**) time and number of hits of the minimum solution.

5.1.3. Test System 3

A test system with 110 generating units possessing quadratic fuel cost characteristics is utilized. The details of the input parameters were taken from [41]. The load demand is assumed to be 15,000 MW. The best solution obtained by the proposed RLGWO can be seen in Table 5. A comparison of the convergence characteristics of 110 generators between the GWO and RLGWO methods with respect to fitness value is presented in Figure 8.

The maximum, minimum, and average achieved over 50 trials are given in Table 6. Figure 9 presents a visualization of Table 6. The RLGWO produced the lowest cost among the five algorithms and was able to find the minimum cost the most times (as can be seen from Figure 9b). Still, it is the fastest algorithm among the five algorithms tested.

Table 5. Best simulation results for the 110-unit system with loss ($P_D = 15,000$ MW).

Unit	Power Output (MW)	Unit	Power Output (MW)	Unit	Power Output (MW)	Unit	Power Output (MW)	Unit	Power Output (MW)
1	2.34	24	350.0000	47	5.4008	70	360.0000	93	440.0000
2	2.3985	25	400.0000	48	5.4000	71	400.0000	94	499.972
3	2.3985	26	400.0000	49	8.4012	72	400.0000	95	600.0000
4	2.3991	27	499.9991	50	8.4000	73	107.8423	96	469.2669
5	2.3991	28	500.0000	51	8.4000	74	188.8107	97	3.6000
6	4.0006	29	199.9902	52	12.0000	75	89.9956	98	3.6000
7	4.0000	30	99.9986	53	12.0000	76	49.9991	99	4.4000
8	4.0000	31	10.0003	54	12.008	77	160.0137	100	4.4002
9	4.0000	32	19.9852	55	12.0006	78	291.3489	101	10.0075
10	63.0201	33	79.4518	56	25.2000	79	176.9972	102	10.003
11	59.2256	34	250.0000	57	25.2000	80	97.7541	103	20.0037
12	35.5324	35	359.9998	58	35.0000	81	10.0007	104	20.0036
13	57.4253	36	399.9929	59	35.0000	82	12.3103	105	40.0000
14	25.0000	37	39.9998	60	45.0013	83	20.0453	106	40.0011
15	25.0000	38	69.9965	61	45.0013	84	199.9869	107	50.0000
16	25.00011	39	99.9896	62	45.0000	85	324.5166	108	30.0000
17	155.0000	40	120.0000	63	184.9962	86	439.9885	109	40.0000
18	155.0000	41	156.8012	64	184.9953	87	18.8657	110	20.0000
19	155.0000	42	219.9997	65	184.9997	88	23.3351		
20	154.9936	43	440.0000	66	185.0000	89	84.4017	Fuel Cost ($/h)	197,952.36
21	68.89	44	560.0000	67	70.0000	90	91.9005		
22	68.89	45	660.0000	68	70.0000	91	58.2891		
23	68.89	46	619.5389	69	70.0006	92	98.38849		

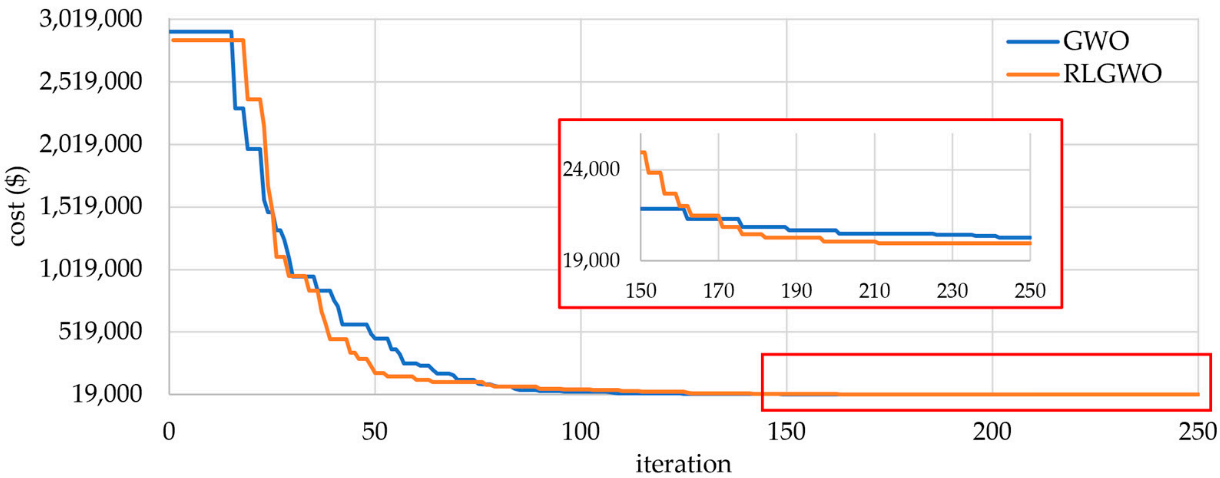


Figure 8. Comparative convergence characteristics of the original GWO and the proposed RLGWO algorithms for the 110-unit system.

Table 6. Comparison between different methods taken after 50 trials (110-unit system with loss).

Method	Generation Costs			Time (s)	No. of Hits of the Minimum Solution
	Max	Min	Average		
RLGWO	197,952.88	197,952.36	197,952.62	28	49
ORCCRO	198,016.89	198,016.29	198,016.32	45	48
OIWO	197,989.93	197,989.14	197,989.41	31	46
DE/BBO	198,828.57	198,231.06	198,326.66	132	43
BBO	199,102.59	198,241.166	198,413.45	115	41

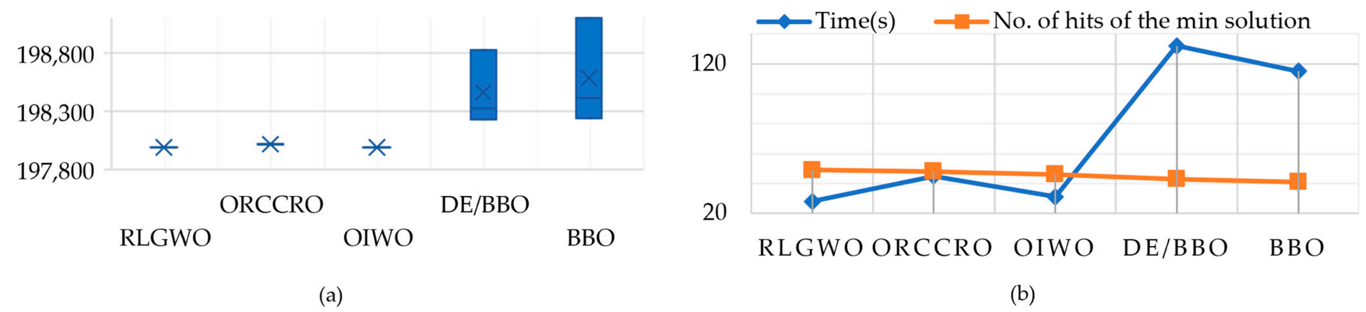


Figure 9. Results for different methods for the 110-unit system (50 trials): (**a**) boxplot of the final generation cost; (**b**) time and number of hits of the minimum solution.

5.1.4. Test System 4

This case study considers 140 generators belonging to Korea's power system. Twelve generating units possessing a cost function with the valve-point loading effect are utilized, while ramp rate limits, prohibited operating zones and system losses are neglected; the input parameters are available in [14]. The cheapest operating cost obtained using the RLGWO technique is shown in Table 7. The convergence characteristics for 140 generators using GWO and RLGWO methods for the fitness value are presented in Figure 10.

Table 7. Best simulation results of 140-unit system with loss ($P_D = 49,342$ MW).

Unit	Power Output (MW)	Unit	Power Output (MW)	Unit	Power Output (MW)	Unit	Power Output (MW)	Unit	Power Output (MW)
1	116.74945	29	501	57	103	85	115	113	94
2	189	30	501	58	198	86	207	114	94
3	190	31	506	59	312	87	207	115	244
4	190	32	506	60	281.8723	88	175	116	244
5	168.53975	33	506	61	163	89	175	117	244
6	190	34	506	62	95	90	175	118	95
7	490	35	500	63	160.442	91	175	119	95
8	490	36	500	64	160	92	580	120	116
9	496	37	241	65	490	93	645	121	175
10	496	38	241	66	196.13105	94	984	122	2
11	496	39	774	67	490	95	978	123	4
12	496	40	769	68	489.80065	96	682	124	15
13	506	41	3	69	130	97	720	125	9
14	509	42	3	70	234.70625	98	718	126	12
15	506	43	250	71	137	99	720	127	10
16	505	44	248.19195	72	325.6586	100	964	128	112
17	506	45	250	73	195	101	958	129	4
18	506	46	250	74	175.3892	102	1007	130	5
19	505	47	245.45	75	175	103	1006	131	5
20	505	48	250	76	175.9936	104	1013	132	50
21	505	49	250	77	175.4087	105	1020	133	5
22	505	50	250	78	330	106	954	134	42
23	505	51	165	79	531	107	952	135	42
24	505	52	165	80	531	108	1006	136	41
25	537	53	165	81	382.3908	109	1013	137	17
26	537	54	165	82	56	110	1021	138	11.211
27	549	55	180	83	115	111	1015	139	7
28	549	56	180	84	115	112	94	140	26.06515
Fuel Cost ($/h) = 1,559,390					Load demand = 49,342 MW				

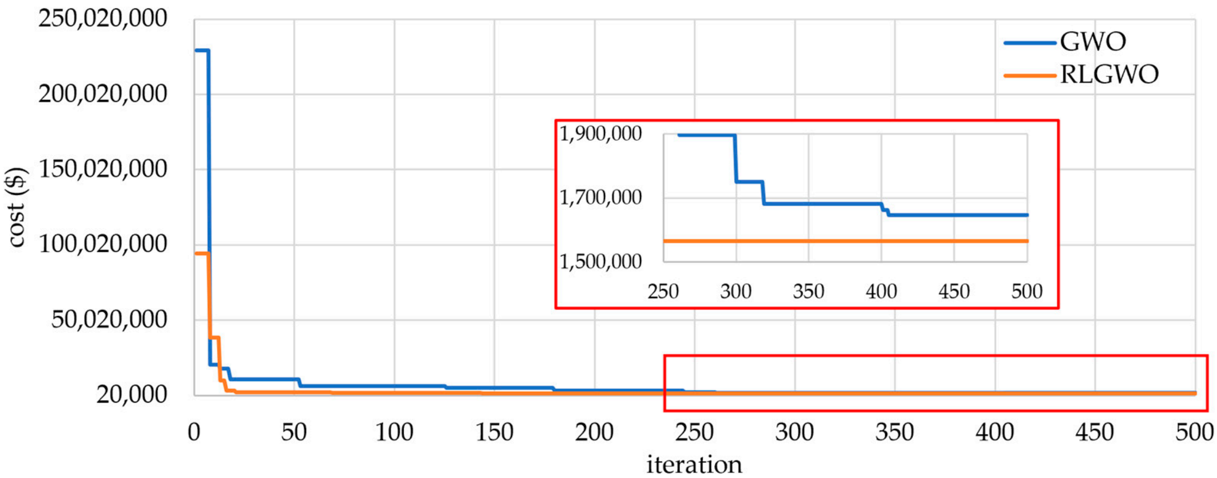


Figure 10. Comparative convergence characteristic of original GWO and proposed RLGWO algorithm for the 140-unit system.

The statistical results, listed in Table 8, present the maximum, minimum, and average cost obtained by the OIWO method and the proposed RLGWO method over 50 trials. Figure 11 presents a visualization of Table 8. For this test problem, RLGWO produced a lower result than OIWO, as seen in Figure 11a. Compared to OIWO, it found the lowest cost more times out of 50 trials, and was faster than OIWO. This may mean that the increase in problem size affects the RLGWO less than OIWO.

Table 8. Comparison between different methods taken after 50 trials (140-unit system with loss).

Method	Generation Costs			Time (s)	No. of Hits of the Minimum Solution
	Max	Min	Average		
RLGWO	1,559,389.896	1,559,389.595	1,559,389.746	44.2	49
OIWO	1,559,405.88	1,559,405.45	1,559,405.61	46.8	46

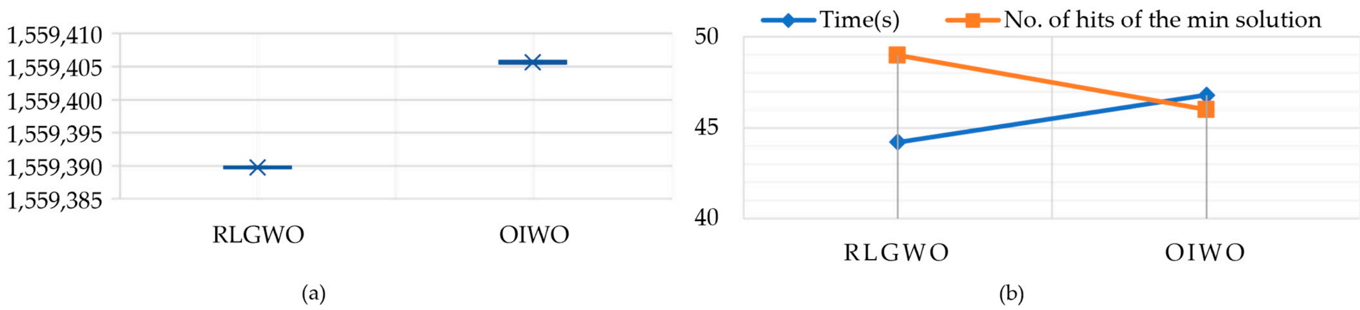


Figure 11. Results obtaind using different methods for the 140-unit system (50 trials): (**a**) boxplot of the final generation cost; (**b**) time and number of hits of the minimum solution.

5.1.5. Test System 5

This case considers a complex system with 160 testing units having multiple options and a valve-point effect. The total load demand is 43,200 MW. The input parameters for 160 generators are generated by multiplying those for the 10-unit system up to reflect a 160-unit system. The data system for 10 generators was taken from [14]. Transmission loss is ignored in this case. The best generating cost obtained by the proposed RLGWO is shown in Table 9. The convergence characteristic of 160 generators with GWO and RLGWO methods for fitness value is presented in Figure 12.

Table 9. Best simulation results for the 160-unit system with loss ($P_D = 43,200$ MW).

Unit	Power Output (MW)	Unit	Power Output (MW)	Unit	Power Output (MW)	Unit	Power Output (MW)	Unit	Power Output (MW)	Unit	Power Output (MW)
1	221.00035	28	231.8196	55	270.9179	82	201.35535	109	430.44955	136	237.9752
2	205.7281	29	425.93425	56	248.48075	83	314.73155	110	269.64185	137	295.55395
3	317.75575	30	279.58085	57	281.2539	84	237.5059	111	217.76425	138	235.6813
4	234.41735	31	196.63605	58	238.18815	85	275.93495	112	205.1143	139	430.2502
5	2912142	32	207.35155	59	412.3851	86	224.36975	113	318.7791	140	268.2663
6	244.9315	33	314.11745	60	278.72325	87	284.14205	114	245.6407	141	223.00885
7	297.5845	34	229.92935	61	229.1329	88	247.8263	115	271.79755	142	197.70995
8	229.63355	35	280.2443	62	212.27065	89	419.6053	116	246.03355	143	325.12455
9	382.95085	36	242.2372	63	310.3729	90	274.05395	117	288.0778	144	229.6929
10	274.7838	37	299.5748	64	244.48875	91	205.04785	118	237.1986	145	270.23625
11	202.63435	38	238.49285	65	254.77565	92	206.14835	119	379.98875	146	237.61965
12	203.3912	39	385.28865	66	242.9497	93	315.2499	120	289.60535	147	293.18245
13	316.69635	40	306.1279	67	297.698	94	249.54195	121	227.5547	148	241.3283
14	240.61475	41	223.84795	68	236.58335	95	283.55655	122	209.27635	149	408.69515
15	274.9389	42	210.5801	69	391.7068	96	233.38845	123	308.35835	150	273.40185
16	234.0219	43	310.6721	70	280.0213	97	292.58485	124	225.3519	151	201.19515
17	281.37655	44	236.8065	71	218.70865	98	232.1082	125	258.85355	152	211.8315
18	232.42195	45	288.09185	72	215.8047	99	398.45225	126	243.2848	153	328.14365
19	417.2048	46	251.70135	73	311.6729	100	283.92175	127	289.6619	154	244.37645
20	296.69975	47	264.1946	74	245.34615	101	222.28885	128	243.20055	155	277.40895
21	211.53615	48	226.43815	75	274.99975	102	211.05235	129	430.75805	156	231.75945
22	205.97795	49	396.5202	76	228.8927	103	321.53435	130	263.69975	157	274.733
23	327.2297	50	291.1473	77	295.2527	104	231.10145	131	208.5112	158	237.31805
24	237.09925	51	199.54815	78	242.0892	105	255.53405	132	214.2303	159	410.52905
25	280.7248	52	211.2342	79	391.79425	106	236.2335	133	310.9484	160	282.7046
26	230.7831	53	322.1792	80	275.43885	107	283.8343	134	234.9219	Fuel Cost (S/h)	9914.7123
27	269.3144	54	237.0891	81	220.4749	108	238.32985	135	263.6612		

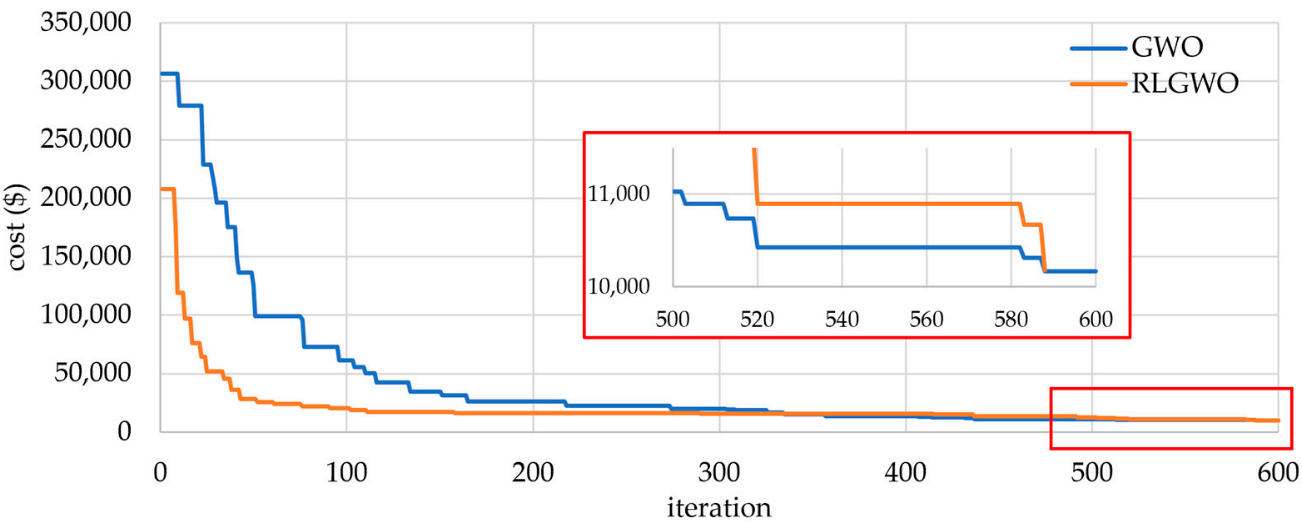


Figure 12. Comparative convergence characteristic of original GWO and proposed RLGWO algorithm for the 160-unit system.

The maximum, minimum, and average fuel costs acquired by various techniques are represented in Table 10. For this 160-unit system, it can be clearly seen that the proposed algorithm still produced the lowest result (Figure 13a), and the number of times this minimum result was found was the greatest among the algorithms tested (Figure 13b). Furthermore, the RLGWO is still the fastest algorithm of the five algorithms.

Table 10. Comparison between different methods taken after 50 trials (160-unit system).

Method	Generation Costs			Time (s)	No. of Hits of the Minimum Solution
	Max	Min	Average		
RLGWO	9916.4457	9914.7123	9915.579	13.7	48
ORCCRO	10,004.45	10,004.2	10,004.21	19	48
OIWO	9983.998	9981.9834	9982.991	17.3	46
DE/BBO	10,010.25	10,007.05	10,007.56	35	42
BBO	10,010.59	10,008.71	10,009.16	44	40

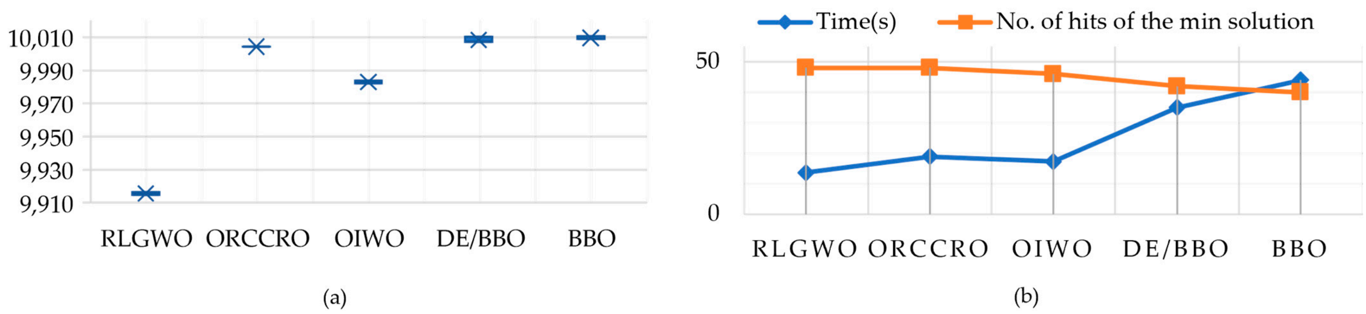


Figure 13. Results for different methods for the 160-unit system (50 trials): (**a**) boxplot of the final generation cost; (**b**) time and number of hits of the minimum solution.

5.2. Comparative Study

5.2.1. Solution Quality

Tables 1, 3, 5, 7 and 9 present the cheapest generation cost determined by the RLGWO algorithm for five different test systems. It can be clearly seen that the proposed approach usually provides a better solution compared to the results obtained by various most technique. Additionally, the maximum, minimum, and average values acquired using different methods are illustrated in Tables 2, 4, 6, 8 and 10. These results emphasize the ability of RLGWO to achieve better solutions than most existing techniques.

5.2.2. Computational Efficiency and Robustness

Addressing a large and complex system can increase the time consumption of any algorithm. Therefore, we used execution time to judge the computational efficiency of the optimization technique. The results shown in Tables 2, 4, 6, 8 and 10, with their corresponding figures, Figures 5, 7, 9, 11 and 13, respectively, prove that the proposed RLGWO requires shorter CPU time to obtain the minimum fuel cost when compared to other reported techniques, except for in the cases of Test 1 and Test 2, where the proposed algorithm ranked second to the least cost. Additionally, those tables and figures also reveal the robustness of the proposed approach. Of the 50 trials performed for the five different test systems, RLGWO obtained the minimum costs 50, 50, 49, 49, and 48 times, respectively. In other words, the efficiency of the RLGWO algorithm for the first two test systems is 100%, and in the remaining cases is 98% or 96%. Additionally, OIWO obtained the minimum cost 47, 46, 46, 46, and 46 times. These results confirm that the performance of RLGWO is outstanding compared to several other methods.

Consequently, the results mentioned above emphasize the ability of RLGWO to obtain high-quality solutions in a way that achieves computational efficiency and robustness.

6. Conclusions and Future Work

This paper presented a newly developed RLGWO algorithm for solving various complex, large-scale economic dispatch problems. To evaluate the feasibility and robustness of the proposed algorithm, five test systems, with 13, 40, 110, 140, and 160 units, respectively, were used. The simulation results revealed the competitive performance of RLGWO when compared to other optimization methods, successfully determining the cheapest generation cost in all cases. GWO is considered a straightforward optimization technique that obviates the need for the initialization of input parameters and has demonstrated its superiority in several optimization problems. The robust tolerance-based adjust searching direction mechanism and opposite-based learning combined with GWO allow it to improve the convergence rate and generate promising search agents. Future work for this research will include applying RLGWO in solving complex real-world power system problems.

Author Contributions: Conceptualization, T.-C.T. and C.-C.L.; methodology, T.-C.T.; software, C.-C.L.; formal analysis, T.-C.T. and C.-C.K.; writing—original draft preparation, T.-C.T.; writing—review and editing, T.-C.T.; visualization, C.-C.L.; supervision, C.-C.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: We would like to acknowledge project NSTC112-2622-8-011-011-TE1 for the assistance in our research.

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

References

1. Chakraborty, S.; Senjyu, T.; Yona, A.; Saber, A.; Funabashi, T. Solving economic load dispatch problem with valve-point effects using a hybrid quantum mechanics inspired particle swarm optimisation. *IET Gener. Transm. Distrib.* **2011**, *5*, 1042–1052. [CrossRef]
2. Wood, A.J.; Wollenberg, B.F.; Sheblé, G.B. *Power Generation, Operation, and Control*, 3rd ed.; Wiley-Interscience: Hoboken, NJ, USA, 2014.
3. Lee, K.Y.; Park, Y.M.; Ortiz, J.L. Fuel-cost minimisation for both real-and reactive-power dispatches. *IEE Proc. C (Gener. Transm. Distrib.)* **1984**, *131*, 85–93. [CrossRef]
4. Sasson, A.M. Nonlinear Programming Solutions for Load-Flow, Minimum-Loss, and Economic Dispatching Problems. *IEEE Trans. Power Appar. Syst.* **1969**, *PAS-88*, 399–409. [CrossRef]
5. Liang, Z.-X.; Glover, J.D. A zoom feature for a dynamic programming solution to economic dispatch including transmission losses. *IEEE Trans. Power Syst.* **1992**, *7*, 544–550. [CrossRef]
6. Wong, K.; Fung, C. Simulated annealing based economic dispatch algorithm. *IEE Proc. C Gener. Transm. Distrib.* **1993**, *140*, 509–515. [CrossRef]
7. Chen, P.-H.; Chang, H.-C. Large-scale economic dispatch by genetic algorithm. *IEEE Trans. Power Syst.* **1995**, *10*, 1919–1926. [CrossRef]
8. Lin, W.-M.; Cheng, F.-S.; Tsay, M.-T. An improved tabu search for economic dispatch with multiple minima. *IEEE Trans. Power Syst.* **2002**, *17*, 108–112. [CrossRef]
9. Ruey-Hsum, L. A neural-based redispatch approach to dynamic generation allocation. *IEEE Trans. Power Syst.* **1999**, *14*, 1388–1393. [CrossRef]
10. Yalcinoz, T.; Short, M.J. Neural networks approach for solving economic dispatch problem with transmission capacity constraints. *IEEE Trans. Power Syst.* **1998**, *13*, 307–313. [CrossRef]
11. Gaing, Z.-L. Particle swarm optimization to solving the economic dispatch considering the generator constraints. *IEEE Trans. Power Syst.* **2003**, *18*, 1187–1195. [CrossRef]
12. Park, J.B.; Shin, J.R.; Lee, K.Y. Closure to discussion of "An improved particle swarm optimization for nonconvex economic dispatch prob-lems". *IEEE Trans. Power Syst.* **2010**, *25*, 2010–2011. [CrossRef]
13. Bhattacharya, A.; Chattopadhyay, P.K. Biogeography-Based Optimization for Different Economic Load Dispatch Problems. *IEEE Trans. Power Syst.* **2009**, *25*, 1064–1077. [CrossRef]
14. Chiang, C.-L. Improved genetic algorithm for power economic dispatch of units with valve-point effects and multiple fuels. *IEEE Trans. Power Syst.* **2005**, *20*, 1690–1699. [CrossRef]
15. Djokić, S.Ž.; Milanović, J.V.; Rowland, S.M. Advanced voltage sag characterisation II: Point on wave. *IET Gener. Trans-Mission. Distrib.* **2007**, *1*, 146–154. [CrossRef]
16. Noman, N.; Iba, H. Differential evolution for economic load dispatch problems. *Electr. Power Syst. Res.* **2008**, *78*, 1322–1331. [CrossRef]
17. Hou, Y.H.; Wu, W.; Xiong, X. Economic dispatch of power systems based on generalized ant colony optimization method. *Zhongguo Dianji Gongcheng Xuebao/Proc. Chin. Soc. Electr. Eng.* **2003**, *23*, 59.

18. Coelho, L.D.S.; Mariani, V.C. Correction to "Combining of Chaotic Differential Evolution and Quadratic Programming for Economic Dispatch Optimization with Valve-Point Effect". *IEEE Trans. Power Syst.* **2006**, *21*, 1465. [CrossRef]
19. Vitthaladevuni, P.K.; Alouini, M.S. Discussion of "Economic Load Dispatch-A Comparative Study on Heuristic Optimi-zation Techniques With an Improved Coordinated Aggregation-Based PSO". *IEEE Trans. Broadcast.* **2003**, *49*, 408. [CrossRef]
20. Meng, K.; Wang, H.G.; Dong, Z.; Wong, K.P. Quantum-Inspired Particle Swarm Optimization for Valve-Point Economic Load Dispatch. *IEEE Trans. Power Syst.* **2009**, *25*, 215–222. [CrossRef]
21. Roy, P.; Roy, P.; Chakrabarti, A. Modified shuffled frog leaping algorithm with genetic algorithm crossover for solving economic load dispatch problem with valve-point effect. *Appl. Soft Comput.* **2013**, *13*, 4244–4252. [CrossRef]
22. Bhattacharya, A.; Chattopadhyay, P.K. Hybrid Differential Evolution With Biogeography-Based Optimization for Solution of Economic Load Dispatch. *IEEE Trans. Power Syst.* **2010**, *25*, 1955–1964. [CrossRef]
23. Dai, C.; Hu, Z.; Su, Q. An adaptive hybrid backtracking search optimization algorithm for dynamic economic dispatch with valve-point effects. *Energy* **2021**, *239*, 122461. [CrossRef]
24. Barisal, A.K.; Prusty, R.C. Large scale economic dispatch of power systems using oppositional invasive weed optimization. *Appl. Soft Comput.* **2015**, *29*, 122–137. [CrossRef]
25. Tizhoosh, H.R. Opposition-Based Learning: A New Scheme for Machine Intelligence. In Proceedings of the International Conference on Com-putational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC'06), Vienna, Austria, 28–30 November 2005.
26. Mirjalili, S.; Mirjalili, S.M.; Lewis, A. Grey Wolf Optimizer. *Adv. Eng. Softw.* **2014**, *69*, 46–61. [CrossRef]
27. El-Fergany, A.A.; Hasanien, H.H. Single and Multi-objective Optimal Power Flow Using Grey Wolf Optimizer and Dif-ferential Evolution Algorithms. *Electr. Power Compon. Syst.* **2015**, *43*, 1548–1559. [CrossRef]
28. Shakarami, M.R.; Faraji, I. Design of SSSC-based Stabilizer to Damp Inter-Area Oscillations Using Gray Wolf Optimization Algorithm. 2015. Available online: https://sid.ir/paper/907750/en (accessed on 13 February 2023).
29. Bhattacharjee, K.; Bhattacharya, A.; Dey, S.H.N. Oppositional Real Coded Chemical Reaction Optimization for different economic dispatch problems. *Int. J. Electr. Power Energy Syst.* **2014**, *55*, 378–391. [CrossRef]
30. Sinha, N.; Chakrabarti, R.; Chattopadhyay, P.K. Evolutionary programming techniques for economic load dispatch. *IEEE Trans. Evol. Comput.* **2003**, *7*, 83–94. [CrossRef]
31. Muro, C.; Escobedo, R.; Spector, L.; Coppinger, R. Wolf-pack (Canis lupus) hunting strategies emerge from simple rules in computational simulations. *Behav. Process.* **2011**, *88*, 192–197. [CrossRef]
32. Long, W.; Jiao, J.; Liang, X.; Tang, M. An exploration-enhanced grey wolf optimizer to solve high-dimensional numerical optimization. *Eng. Appl. Artif. Intell.* **2018**, *68*, 63–80. [CrossRef]
33. Gaidhane, P.J.; Nigam, M.J. A hybrid grey wolf optimizer and artificial bee colony algorithm for enhancing the performance of complex systems. *J. Comput. Sci.* **2018**, *27*, 284–302. [CrossRef]
34. Wang, F.; Zhang, H.; Li, K.; Lin, Z.; Yang, J.; Shen, X.-L. A hybrid particle swarm optimization algorithm using adaptive learning strategy. *Inf. Sci.* **2018**, *436–437*, 162–177. [CrossRef]
35. Shaw, B.; Mukherjee, V.; Ghoshal, S. A novel opposition-based gravitational search algorithm for combined economic and emission dispatch problems of power systems. *Int. J. Electr. Power Energy Syst.* **2012**, *35*, 21–33. [CrossRef]
36. Chatterjee, A.; Ghoshal, S.; Mukherjee, V. Solution of combined economic and emission dispatch problems of power systems by an opposition-based harmony search algorithm. *Int. J. Electr. Power Energy Syst.* **2012**, *39*, 9–20. [CrossRef]
37. Mandal, B.; Roy, P.K. Optimal reactive power dispatch using quasi-oppositional teaching learning based optimization. *Int. J. Electr. Power Energy Syst.* **2013**, *53*, 123–134. [CrossRef]
38. Ergezer, M.; Simon, D. Oppositional biogeography-based optimization for combinatorial problems. In Proceedings of the 2011 IEEE Congress of Evolutionary Computation (CEC), New Orleans, LA, USA, 5–8 June 2011.
39. Reddy, A.S.; Vaisakh, K. Shuffled differential evolution for large scale economic dispatch. *Electr. Power Syst. Res.* **2013**, *96*, 237–245. [CrossRef]
40. Ciornei, I.; Kyriakides, E. A GA-API Solution for the Economic Dispatch of Generation in Power System Operation. *IEEE Trans. Power Syst.* **2011**, *27*, 233–242. [CrossRef]
41. Orero, S.O.; Irving, M.R. Large scale unit commitment using a hybrid genetic algorithm. *Int. J. Electr. Power Energy Syst.* **1997**, *19*, 45–55. [CrossRef]