Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

solving the constrained economic load distribution problem: combination of gray wolf optimizer and colonial competitive algorithm

Roqia Rateb
Al-Hussein Bin Abdallah II For IT, The Hashemite, University
Maytham N Meqdad
Al-Mustaqbal University
Taybeh Salehnia
salehnia.taybeh@gmail.com
Razi University

Article

Keywords: Engineering optimization, Colonial competitive GWO (CCGWO), Power system operation, constrained economic load dispatch (ELD)

Posted Date: August 31st, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10637810/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

solving the constrained economic load distribution problem: combination of gray wolf optimizer and colonial competitive algorithm

Roqia Rateb[1], Maytham N Meqdad[2], Taybeh Salehnia[3]

[1] Department of Information Technology, Faculty of Prince, Al-Hussein Bin Abdallah II For IT, The Hashemite, University, P.O. Box 330127, Zarqa 13133, Jordan, roqiashorman@yahoo.com.
[2] Intelligent Medical Systems Department, College of Sciences, Al-Mustaqbal University, 51001, Babylon, Iraq, maytham.meqdad@uomus.edu.iq.
[3] Department of Computer Engineering and Information Technology, Razi University, Kermanshah, Iran, salehnia.taybeh@gmail.com.

* Correspondence: salehnia.taybeh@gmail.com

Abstract Today, contemporary power systems face the economic load dispatch (ELD) problem as a pivotal challenge due to its profound economic and operational implications. ELD represents a multifaceted optimization conundrum involving the adjustment of power outputs from various generating units while adhering to a myriad of constraints imposed by the power grid infrastructure, generation facilities, and fuel resources. This study introduces an enhanced variant of the Grey Wolf Optimizer (GWO), called the Colonial Competitive Grey Wolf Optimizer (CCGWO), which integrates socio-political evolution modeling principles. The CCGWO algorithm is tailored explicitly for addressing ELD problems, aiming to mitigate a key limitation observed in conventional GWO methods: susceptibility to local optima entrapment, especially in intricate problem domains. By dividing the population into distinct groups and simulating colonial competition dynamics among these groups, CCGWO fosters robust exploration capabilities. Moreover, all individuals actively engage in the search process, thereby fostering enhanced agility in navigating away from local optima. Experimental evaluations conducted across various test power systems comprising 6, 15, 110, and 140 generators validate the efficacy and superiority of the proposed CCGWO approach over conventional GWO techniques for tackling a broad spectrum of ELD challenges.

Keywords: Engineering optimization, Colonial competitive GWO (CCGWO), Power system operation, constrained economic load dispatch (ELD)

NOMENCLATURE

Indices

S	Index for slack generator.
i, h	Indices for thermal generation units.
k	Index for fuel types.
z	Index prohibited operating zones (POZ).
m	Index for grey wolves.
n	Index for the groups of grey wolves.
α, β, δ	Indices for alpha, beta, and delta wolves.

Constants

N_g	Number of thermal generating units.
P_D	Load demand
$P_{gi,min}$	Minimum power generation of the thermal unit.
$P_{gi,max}$	Maximum power generation of the thermal unit.
UR_i, DR_i	Ramp-up and ramp-down limits of the generator.
a, b, c, e, f	Cost coefficients of the thermal unit.
B_{ih}, B_{0i}, B_{00}	The coefficients matrices of the active power losses.
n_i	The number of POZ for generator i.
$P_{gi,z}^L$, $P_{gi,z}^U$	Minimum and maximum limits of the z^{th} POZ.
$Iter_{max}$	Maximum number of iterations of the algorithm.
$Npop$	The total number of wolves (i.e., population members).
Ngr	The number of groups into which the entire wolf population is divided.

Variables

F_i	Fuel cost of the i^{th} thermal generating unit.
P_{gi}	Output power of the unit i.
P_L	Power system losses.
P_{gi}^0	The active power output of the i^{th} generator in the preceding period.
P_{gS}	The output power of the slack generator.
$\vec{X}_m$	The position vector of the m^{th} grey wolf (i.e., the m^{th} candidate solution).
$\vec{X}_{m,best}$	The stored best personal position for the m^{th} grey wolf.
TP_n	The power of the n^{th} group ($n = 1,2,..,Ng$).

Parameters

t	Iteration counter of the algorithm.
$\vec{r}_1$, $\vec{r}_2$	Vectors with uniformly distributed random numbers in the range [0, 1].

INTRODUCTION

System engineers strive to achieve optimal performance of power systems by minimizing design and operation costs while meeting technical constraints. This fact is done through various optimization problems, such as optimal power flow (OPF) [1], [2], demand response model [3], and forecasting and optimal load allocation in combined energy systems [4]. In this regard, solving the economic load dispatch (ELD) problem [5], [6] is crucial for the power system operators in the optimal management of generation centers. From the operation point of view, the ELD problem [5], [6] is mainly solved by minimizing the fuel costs and considering some constraints, including generation units' limitations, prohibited operating zones, the effects of steam valves, and transmission line losses [7]. Despite the improvement of technology in the field of renewable energies like solar and wind units [8], [9], a large share of the power demand is still supplied by thermal power plants that use fossil fuels [10]. The sum of generation costs can be significantly reduced by optimal management and scheduling of such units, called the ELD problem [11]. In other words, the ELD problem is solved by optimally planning the generating powers of all plants in a certain period so that the required load demand is supplied in each specific time horizon while meeting the power system constraints [12], [13].

The effects of the steam valves on ELD problems are considered by inserting a nonlinear term into the objective function, leading to a non-smooth cost function for the generation unit, which increases the complexity of the problem. Since classical methods (that typically use the derivative of the objective function) cannot solve such non-smooth and non-convex problems, metaheuristic algorithms have been presented as efficient alternative tools in the literature [14], [15], [16]. The main features of this class of optimization algorithms are exploration (global search) and exploitation (local search in the vicinity of the optimal point), which make them suitable options for solving non-convex problems [17], especially for optimization in complex systems like unbalanced active distribution networks [18], and wireless power transfer systems [19].

Numerous metaheuristic algorithms have recently been employed to solve non-smooth and non-convex ELD problems. A new social spider optimization (ISSO) was proposed in [20], which can find good quality solutions more quickly than the SSO method while reducing the number of control parameters and computational processes needed. The overall generation cost can be reduced using ISSO, and higher search speeds can be achieved compared to all other techniques considered. In [21], a hybrid algorithm called the evolutionary simplex adaptive Hooke-Jeeves algorithm (ESAHJ) was proposed that combines a genetic algorithm (GA) with the Hooke-Jeeves algorithm. The algorithm's performance was tested by applying it to five different generating systems.

In [22], a technique was suggested to improve the global search ability of particle swarm optimization (PSO) algorithms. The technique involves using a real-valued mutation (RVM) operator with the constriction parameter and inertia weight. The cuckoo search algorithm (CSA) [23] was employed to tackle non-convex ELD problems that account for generator and system features, such as valve-point effects (VPE), multiple fuel types (MF), prohibited operating zones (POZ), spinning reserve, and power losses. A hybrid algorithm consisting of a GA and a nonlinear time-varying double-weighted PSO was proposed in [24] to address the combined environmental economic dispatch (CEED) problem. Goudarzi et al. incorporated two new correctional techniques to screen and unceasingly correct the particles' position and velocity to improve this algorithm's exploitation capability. They conducted various numerical experiments on multiple scales to demonstrate the MGAIPSO algorithm's practicality.

Reference [25] introduced a new optimization method called civilized swarm optimization (CSO), which combines the society-civilization algorithm (SCA) with PSO. The biogeography-based optimization (BBO) algorithm [26] was suggested as a method for solving ELD problems with a variety of constraints, including VPE, transmission losses (TL), ramp rate limits (RRL), MF options, and POZ. An accelerated BBO method that uses neighborhood search, aBBOmDE, was proposed in [27]. It improves the performance of BBO by incorporating a modified mutation and clear duplicate operators. The method also employs a modified DE algorithm (mDE) as a neighborhood search operator to enhance the fitness of solutions beyond a certain threshold. The mDE algorithm uses the DE/best/1/bin mutation operator to explore solutions in the vicinity of the best solution. The length of the local search was optimized to balance search capability and computational cost. The method retains BBO's exploitation ability by keeping the migration mechanism the same as that in BBO. In [28], a self-organizing hierarchical PSO (SOH-PSO) algorithm was proposed to overcome the problem of premature convergence. This improvement was achieved by taking two steps: firstly, the velocity vector was reinitialized whenever it stagnated, and secondly, time-varying acceleration coefficients (TVAC) were used to balance the local and global searches. The utilization of chaos maps in the chaotic modified imperialist competitive algorithms (CMICA) [29] allows for more efficient solution finding compared to the original imperialist competitive algorithm (ICA) method. The CMICA methods leverage these chaos maps to achieve superior solutions faster. In [30], a bacterial foraging optimization (BFO) algorithm was proposed to solve non-convex ELD problems involving transmission losses, ramp rate limits, and prohibited zones. The algorithm uses a simplex method called Nelder-Mead in addition to the BFO algorithm to explore the search space. This method allows for the detection of local minima in the current location. An enhanced variant of the Big Bang–Big Crunch optimization (HBB-BC) algorithm was introduced in [31]. The proposed method was tested on three distinct systems: a six-unit system considering losses, a 15-unit system considering transmission loss for ELD, and a large-scale system with 40 generating units accounting for valve-point loading effects to evaluate its performance. Comparative analysis with other optimization methods from recent literature demonstrated the feasibility of this technique for effectively addressing highly nonlinear ELD problems. An alternative firefly method incorporating a non-homogeneous population (NhFA-Rnp) was developed in [32]. Conducted experimentation involving ten benchmark functions verified the effectiveness of this novel approach. Subsequently, it was applied to solve two distinct ELD problems: a 15-unit scenario considering transmission line losses, power and ramp limits, and POZ for each generator, and a non-convex system comprising 13 units with VPE embedded in the cost function and power generation constraints. The study outcomes indicated that the proposed non-homogeneous variant outperformed the original firefly algorithm in terms of solution quality. Notably, this improvement was observed regardless of the method for handling power balance constraints, including penalty and non-penalty approaches. A recently proposed optimization method, referred to as the chaotic teaching-learning-based optimization with Lévy flight (CTLBO) [33], incorporates two distinct mechanisms for population evolution. The first part undergoes a teaching-learning process, while the second part is subjected to a Lévy flight. To enhance diversity within the population, a chaos perturbation is applied to a randomly selected portion [33]. An arithmetic-average-bound crossover and wavelet mutation-based real-coded GA (RCGA) [34] was employed to tackle non-convex solution spaces in five test systems. Comparative analysis with other approaches demonstrated the effectiveness of the proposed RCGA in achieving desirable results. An orthogonal learning competitive swarm optimizer (OLCSO) [35] was introduced to tackle ELD problems. It uses an orthogonal learning (OL) strategy to efficiently obtain more valuable information from the winner and loser particles in the CSO. The OL strategy helps create better solutions for OLCSO, resulting in a more

systematic search approach [35]. The economic emission dispatch (EED) problem was effectively solved using a modulated PSO (MPSO) algorithm [36]. The traditional PSO algorithm was modified by incorporating velocity modulation techniques to enhance the exploration and exploitation capabilities within the search space [36]. [37] proposed a granular computing method (GrC) to enhance the accuracy of solving and reduce computational dimensions in ELD problems with VPE. The GrC approach involved a two-step process consisting of coarse fine granular computing, which was interconnected with the load demand and optimized using the intelligent algorithm. A completely decentralized method [38] was introduced, offering a highly efficient solution to the non-convex formulation of economic load dispatch (ELD) problems while accurately considering transmission losses in a decentralized manner. Three case studies were conducted to validate the proposed approach, showcasing its ability to combine the benefits of both centralized and fully decentralized mechanisms in addressing ELD challenges. The obtained results serve as evidence of the effectiveness and advantages of the proposed approach. A modified version of the crow search algorithm (MCSA) [39] was introduced, enhancing the search capability of crows and introducing a novel approach for destination selection. The applicability of MCSA in the economic load dispatch (ELD) problem was demonstrated by applying it to five widely recognized test systems. The results were compared to other methods in terms of solution quality, robustness, and computation time, highlighting the superior performance of the proposed technique [39]. A novel multi-population DE (MPDE) was introduced [40], which was evaluated on test systems comprising 13, 40, 80, and 140 units. The simulation results demonstrated that the MPDE algorithm exhibited significantly reduced variances compared to other intelligent algorithms. An enhanced version of the genetic algorithm, referred to as IGA, was developed [41]. The IGA incorporated an improved evolutionary direction operator and a migration mechanism, enabling it to efficiently explore and search for solutions. The clustering CSO (CCSO) [42] adopts a unique approach compared to CSA, as it utilizes a clustering mechanism to generate the step size instead of random generation. Furthermore, the step size in CCSO is dynamically adjusted based on the discrepancy in average fitness values between individual clusters and the entire swarm. This adjustment aims to enhance the equilibrium between exploration and exploitation, leading to an improved balance in the search process for each solution. The Θ-modified bat algorithm (Θ-MBA) [43] introduces a transformation from Cartesian coordinates to polar coordinates in the search space. This conversion aims to enhance the search capability of the algorithm, enabling it to explore the solution space more effectively.

A combination of the ICA and PSO known as HICA-PSO [44] is proposed to address the multi-area economic dispatch (MAED) problem, a critical issue in the economic operation of interconnected power systems. The multi-strategy ensemble BBO (MsEBBO) [45] effectively maintained a balance between global exploration and local exploitation. The heat transfer search (HTS) approach [46] is utilized to solve the ELD problem by including wind power integration. This method incorporates thermodynamics and heat transfer principles to enhance the optimization process. A refined and adaptable differential evolution algorithm (IL-SHADE) [47] was evaluated for its effectiveness in solving practical economic dispatch problems (EDPs) in three test power systems. Through comparisons with recent advanced techniques, IL-SHADE demonstrated superior performance over L-SHADE and other mentioned methods, confirming its efficiency as an optimizer for EDPs. The exchange market algorithm (EMA) [48], which draws inspiration from stock exchange trading, addresses optimization problems. Extensive examination of the trading techniques used by successful individuals in stock markets has influenced the development of this algorithm. An enhanced variant of the Jaya algorithm, referred to as IJaya [49], exhibits improved performance compared to the original algorithm. IJaya achieves a faster convergence rate by utilizing a distance-varying acceleration coefficient (DVAC) and enhancing population diversity by implementing a self-adaptive population mechanism (SAPM). Jaya-SML [50] was assessed in the context of ELD problems, considering various constraints such as power balance, generating capacity limits, ramp rate limits, POZ, VPE, and MF options. In [51], a set of novel hybrid algorithms was introduced. The study focused on solving ELD problems by incorporating modern heuristic techniques, including the modified shuffled frog leaping algorithm (MSFLA), global-best harmony search algorithm (GHS), and hybrid algorithms like SFLA-GHS and shuffled DE (SDE). The advantages of these proposed approaches over traditional heuristics include the avoidance of local optima trapping, enhanced capability to identify nearby extreme points, and improved handling of constraints. In [52], a new hybrid algorithm called fuzzy-based hybrid PSO-DE (FBHPSO-DE) was designed to solve the multi-objective ELD (MOELD) problem. The proposed algorithm was tested on three test systems to assess its performance. The chaotic self-adaptive differential harmony search algorithm (CSADHS) [53] addressed ELD problems involving VPE, RRL, POZ, and MF options. The effectiveness and suitability of the proposed algorithm were demonstrated on test systems comprising 13, 15, and 40 generating units with valve-point effect, as well as 10 generators. The social optimization algorithm (SOA) [54] draws inspiration from the social behavior of humans and emulates an ideal society that adheres to the principles of justice, including equal opportunities and community values. An enhanced optimization technique, utilizing chaotic behavior and honey bee mating principles (CIHBMO) [55], was applied to three distinct test systems and

benchmarked against other existing methods in the literature. The results demonstrate the efficacy and swiftness of the proposed approach in addressing ELD problems characterized by non-smooth and non-continuous fuel cost functions [55]. In [56], a novel approach called opposition-based real-coded chemical reaction optimization (ORCCRO) was introduced. This method utilized opposition-based learning for population initialization and generation-wise updates. Instead of using pseudo-random numbers, quasi-opposite numbers were employed to enhance the convergence rate of the CRO. The simulation results demonstrated that the proposed technique outperforms various existing optimization methods regarding solution quality and computational efficiency. An enhanced chicken swarm optimization (ICSO) [57] was evaluated, and the findings indicated that the ICSO outperformed both PSO and CSO in terms of convergence speed, global search capability, and total fuel cost across various test systems. The generating unit's input-output non-linearity was considered to satisfy equality constraints, and the test system incorporated both equality and inequality constraints. The results also showcased the effectiveness of the proposed approach in handling the non-linearities and constraints. The Ray optimization (RO) algorithm [58] was evaluated by conducting a thorough numerical comparison with recent research works in the field. This was done to verify the results and demonstrate the potential of the proposed approach. A hybrid GWO (HGWO) [59] was developed by incorporating crossover and mutation operations to enhance its performance. HGWO was applied to solve four ELD problems involving different numbers of generating units (6, 15, 40, and 80), taking into account POZ, VPE, and RRL constraints. The problems were solved both with and without considering transmission losses [59]. A novel hybrid algorithm, C-MIMO-CSO [60], was developed by combining ion-motion and crisscross search algorithms. The performance of the proposed optimizer was assessed by comparing its results in terms of statistical indices with those of other state-of-the-art optimizers. The comparison highlighted the promising potential of the C-MIMO-CSO algorithm for solving both unconstrained benchmark functions and electric power load dispatch problems. In [61], a highly efficient CFA (Coulomb's and Franklin's Algorithm) was presented. The effectiveness of CFA in achieving accurate outcomes across various domains was examined and substantiated through its application to ELD problems, including test cases with different sizes: 6, 10, 15, and 110 units. Comparative assessments were conducted against alternative algorithms inspired by similar principles and against findings reported in existing literature. The simulation results provided compelling evidence of the well-structured and effective performance of the CFA algorithm in addressing a wide range of nonlinear optimization problems. A revised variant of the CSA denoted as MCSA (modified CSA), was introduced in [62]. In contrast to the conventional CSA, the proposed approach in [62] incorporates a self-adjusting step size and employs several neighbor-study techniques to improve the effectiveness of the search process. Furthermore, a refined lambda iteration approach was employed to generate fresh solutions [62]. In [63], an enhanced GWO algorithm (AGWO) was introduced to address practical considerations such as VPE, POZ, RRL, generator capacity limits, and power demand constraints in ELD problems. Heuristic search techniques were utilized to manage equality constraints. The effectiveness of the proposed approach was verified on commonly used benchmark functions and medium to large-scale electrical power systems. The results demonstrated the promising capability of the suggested method in solving ELD problems. In [64], a fresh combination of GRASP and DE algorithms known as C-GRASP–SaDE was introduced. It was employed in power systems comprising 13, 40, and 140 generators, where the fuel cost function incorporated the valve-point loading effect to validate its feasibility and performance. The simulation outcomes demonstrated the proposed technique's superior performance compared to the conventional C-GRASP approach. A novel symbiotic organisms search algorithm variation incorporating a disruptive operator (DSOS) was introduced [65]. The results of the simulations demonstrated that the DSOS algorithm outperforms other intelligent algorithms in achieving significantly lower variances. A random drift PSO (RDPSO) [66] was developed based on the concept of unguided motion exhibited by electrons in the presence of an external electric field within conductive materials. The RDPSO incorporated a novel set of evolutionary equations to effectively enhance its ability to explore global search spaces. A decentralized auction-driven algorithm (AA) [67] employed a consensus mechanism to distribute bids across network agents and reach a consensus on the auction outcome. This facilitated the optimization of power distribution among generating units and the minimization of generation costs. Simulations conducted on standard test systems demonstrated the effectiveness of this approach. An innovative adaptation of the CSA (ORCSA) [68] demonstrated superior solution quality compared to several alternative methods. This highlights the efficacy and efficiency of ORCSA as a viable approach for solving ELD problems. The ant lion optimization (ALO) was proposed in [69]. However, in the study cases, ALO did not outperform all other methods, especially when compared to advanced versions of existing meta-heuristic algorithms. Consequently, it can be inferred that although ALO can be considered as a suitable optimization tool for addressing ELD problems, there is a need for further improvements to enhance the quality of optimal solutions and the speed of convergence. The ALO was proposed in [69]. Nevertheless, ALO did not yield superior outcomes compared to all other approaches investigated in the study, particularly when compared to enhanced versions of established meta-heuristic algorithms. Therefore, it can be inferred that although ALO can be

considered as a viable optimization tool for tackling ELD problems, further enhancements are required to improve the quality of the obtained solutions and the rate at which convergence is achieved. The simplified swarm optimization algorithm (SSOA) was introduced in [70], providing a comprehensive and successful approach to finding efficient and reliable solutions for ELD problems in a fast and timely manner. The algorithm was evaluated on four different test systems with varying numbers of unit (10, 15, 40, and 80) using the suggested optimization method. An enhanced PSO (IODPSO-G) was proposed in [71] to address the ELD problems in both single-area and multi-area power systems, considering the nonlinearity of generator features. A novel optimization algorithm called the continuous quick team search optimizer (CQGSO) was introduced in [72]. The effectiveness of the suggested approach was validated through experiments conducted on five different test systems. The results from case studies and numerical examples demonstrated that the proposed CQGSO algorithm outperformed many of the recently published techniques regarding solution quality. The water cycle algorithm (WCA) was utilized in [73] to address the single- and multi-objective ELD problems and determine each generator's optimal active power output. Three optimization objectives were considered, either individually or simultaneously: (i) minimizing fuel cost while accounting for VPE, (ii) reducing emission rates, and (iii) minimizing total transmission losses. GWO [11] demonstrated superior solution quality in terms of cost, convergence, and robustness for the ELD problems investigated. A chaotic variant of the artificial bee colony algorithm, known as MABC, was introduced [74-77]. This approach integrates a chaotic local search (CLS) to enhance the original ABC algorithm's ability to explore and find feasible optimal solutions for the multi-objective EED problem. The objective is to minimize three conflicting objective functions related to emissions and economic considerations. The krill herd algorithm, specifically KHA-IV, was utilized [78-81]. Various versions of the KHA algorithm were effectively implemented to solve six distinct ELD problems in small, medium, and large-scale power systems.

In summary, if we review the metaheuristic algorithms used for the ELD problem, they can be grouped into four categories based on their underlying mechanisms. This classification helps clarify what each group has been able to address and, just as importantly, what limitations still remain.

1. Improved evolutionary algorithms

Methods such as MPDE, MCSA, and RCGA try to prevent premature convergence by tuning control parameters or adding new operators. Their main advantage is that they are relatively easy to implement. However, when the cost function becomes non-smooth—for example, after considering the valve-point effect—they tend to get trapped in local optima. In practice, after a few iterations the population diversity drops, the search slows down, and exploration becomes weak. What is still missing is a mechanism that preserves diversity up to the final iterations, not only at the beginning.

2. Swarm-based algorithms

Algorithms such as the standard GWO, BFO, and NhFA are inspired by collective behavior in nature. Their distributed structure gives them a solid ability to explore the search space. A common drawback, however, is the lack of a well-defined competition mechanism within the population. Useful information found by one agent in a certain region does not spread efficiently to the others, so part of the algorithm’s potential is wasted. Adding an intra-group competition scheme can improve information sharing and also help reduce premature convergence.

3. Hybrid approaches

Methods such as CSO combined with PSO, or FBHPSO-DE, aim to balance exploration and exploitation by blending two or more algorithms. They often outperform single-method approaches. The trade-off is higher runtime and more parameters to tune. In large-scale systems—for instance, with 140 units—this added complexity can become a real limitation.

4. Chaos-based algorithms

Methods such as CMICA and CTLBO replace random numbers with chaotic maps. This idea improves the distribution of the initial population and helps the algorithm cover the search space more effectively. The problem is that the impact of chaos often fades as the search progresses, and the algorithm gradually falls back into its original behavior. The remaining gap is clear: chaos should support the optimization process throughout the search, not only at the starting stage.

5. What this paper aims to do

Looking across these four categories, three clear gaps emerge: first, the lack of structured competition in swarm-based algorithms; and second, the absence of a consistently maintained balance between exploration and exploitation throughout the search. The proposed CCGWO targets all two. It introduces competitive grey wolf groups to enforce intra-group competition, and an internal crossover operator to preserve population diversity. Put simply, CCGWO is not trying to promote a single new component on its own; it argues that a well-designed combination of a few simple ideas can close a gap that none of them could address effectively in isolation.

The main contributions of this paper are as follows:

1) The colonial competitive grey wolf optimizer (CCGWO), a novel extension of the grey wolf optimizer (GWO) based on incorporating a new socio-political evolution modeling, is designed.

2) The colonial competitive operator enhances the performance of the conventional GWO algorithm by improving its convergence speed, exploration, and exploitation capabilities.

3) An enhanced operator to update the population effectively increases the power of local search compared to the original GWO algorithm.

4) CCGWO is designed to effectively optimize non-convex and nonlinear ELD problems in various power systems. For example, for addressing real-world constraints in different power systems, including generation units' limitations, prohibited operating zones, the effects of steam valves, and transmission line losses.

GWO is one of the most widely used new metaheuristic optimization algorithms, and it has proved itself to be a powerful optimization method for different real-world problems. So, in this paper, an enhanced version of GWO, based on socio-political evolution modeling, called CCGWO, has been developed and proposed to solve the non-smooth and non-convex ELD problems. The essential limits of the ELD problem, introduced in most articles, are modeled in the proposed method. In addition, four standard test systems are simulated, and statistical indices are extracted to compare the proposed CCGWO, GWO, and some other recent algorithms.

ELD problems formulation

The ELD problem is the optimal scheduling of the generation units, considering the nonlinear constraints of the power system and units. ELD is a challenging and nonlinear optimization problem that cannot be solved using classical gradient-based algorithms due to the problem's complexity and the objective function's non-differentiable behavior [75], [82-86].

A. Objective Function

Properly defining the objective function is important in solving an optimization problem. In the current study, the objective function of ELD is the unit's fuel cost, considering the steam valve's effects. Fuel costs are related to the expenses for the rotation of generators and electricity production. The total fuel cost of N_g thermal generating units, $F(P_g)$, can be defined as follows [87-90]:

$$F(P_g) = \sum_{i=1}^{N_g} F_i(P_{gi}) = \sum_{i=1}^{N_g} (a_i P_{gi}^2 + b_i P_{gi} + c_i), \quad (1)$$

where P_{gi} is the output power of the i^{th} unit in MW, and a_i, b_i, and c_i represent the fuel cost function coefficients of unit i in \$/MW^2, \$/MW, and \$. The absolute value of a sinusoidal expression is added to the objective function, as shown in Eq. (2), to consider the effect of the steam valve:

$$F(P_g) = \sum_{i=1}^{N_g} F_i(P_{gi}) = \sum_{i=1}^{N_g} \left(a_i P_{gi}^2 + b_i P_{gi} + c_i + \left|e_i \sin\left(f_i(P_{gi,min} - P_{gi})\right)\right|\right), \quad (2)$$

where f_i and e_i represent the coefficients related to the steam valve effect and $P_{gi,min}$ is the lower limit of the power output of the i^{th} generator.

Proper planning of generators can lead to significant savings in generation costs and better power system safety. The method for finding the optimal solution to the ELD problem should specify the power outputs of generation units considering the constraints such that the objective function of the problem is minimized.

B. Constraints of ELD problem

Some restrictions related to the power system and the generation units must be considered in the ELD problem [91], [92]. In general, the constraints of a constrained optimization problem are formulated as follows [93], [94]:

$$g(x, u) = 0, \quad (3)$$

$$h(x,u) \le 0. \tag{4}$$

Where g and h represent the equality and inequality constraints, respectively, u is a vector constituting control variables, and x is the state variables vector.

The limitation of the inequality in the generated powers:

Generators can produce active power within a boundary imposed by technical limitations. The power generation of the unit i^{th} is bounded as follows [95], [96]:

$$P_{gi,min} \le P_{gi} \le P_{gi,max}. \tag{5}$$

Where $P_{gi,max}$ and $P_{gi,min}$ are the upper and lower bounds for the output power of the unit i^{th}, respectively. The above limits prevent the lower cost unit from exceeding its maximum allowable capacity and the higher cost unit from producing less than its minimum allowable limit.

Limitation of equality in the balance of power:

The total power output of generators has to be the same as the total load consumption plus power losses [97]:

$$\sum_{i=1}^{N_g} P_{gi} = P_D + P_L. \tag{6}$$

Where P_D denotes the load demand and P_L represents the power system losses obtained using the following equation:

$$P_L = \sum_{i=1}^{N_g} \sum_{h=1}^{N_g} P_{gi} B_{ih} P_{gh} + \sum_{i=1}^{N_g} B_{0i} + B_{00}, \tag{7}$$

where P_{gi} and P_{gh} represent the power generated by units i^{th} and h^{th}, respectively, and B is the coefficients matrix of the active power losses.

Disparity restrictions imposed by prohibited operating areas (POZ):

Because of the technical restrictions of generator components or stability-related issues, power generators cannot operate in some areas between their minimum and maximum allowable capacity, known as POZ. So, the permitted operating range of the generating unit i^{th} is specified as follows [98], [99]:

$$\begin{aligned} & P_{gi,min} \le P_{gi} \le P_{gi,1}^{U}; \\ & \quad \vdots \\ & P_{gi,z-1}^{L} \le P_{gi} \le P_{gi,z}^{U}; \;\; z = 2, \dots, n_i, i = 1, \dots, N_g; \\ & \quad \vdots \\ & P_{gi,n_i}^{L} \le P_{gi} \le P_{gi,max}. \end{aligned} \tag{8}$$

Where n_i indicates the number of POZ for the generator i^{th}, and $P_{gi,z}^{L}$ and $P_{gi,z}^{U}$ are the minimum and maximum limits of the z^{th} POZ, respectively.

Inequality limit on ramp rate:

Caused by technical considerations, thermal power generators cannot instantly change their output power. The output power changes at a specific rate. The following equation models this constraint in the ELD problem [98], [99]:

$$\max(P_{gi,min}, P_{gi}^{0} - DR_i) \le P_{gi} \le \min(P_{gi,max}, P_{gi}^{0} + UR_i), \tag{9}$$

where P_{gi}^{0} denotes the active power output of the i^{th} generator in the preceding period and UR_i and DR_i represent the ramp-up and ramp-down limits of the generator, respectively. Using multifactorial evolution [100] to solve constrained ELD optimization problems is better.

C. Considering multiple fuels

If a generating unit utilizes several fuel types, its cost function is defined by several quadratic terms, each corresponding to one fuel type. Therefore, Eq. (2) may be expanded to consider multiple fuels as in Eq. (10) [98], [99]:

Where k indicates the k^{th} fuel type, and a_{ik}, b_{ik}, c_{ik}, e_{ik}, and f_{ik} represent the coefficients for cost function and steam valve effect associated with the k^{th} fuel type of the i^{th} generator.

$$F_i(P_{gi}) = \begin{cases} a_{i1}P_{gi}^2 + b_{i1}P_{gi} + c_{i1} + \left|e_{i1}\sin\left(f_{i1}(P_{gi,min} - P_{gi})\right)\right|, & 1^{st}\ \text{fuel}, P_{gi,min} \le P_{gi} \le P_{gi,1}; \\ a_{i2}P_{gi}^2 + b_{i2}P_{gi} + c_{i2} + \left|e_{i2}\sin\left(f_{i2}(P_{gi,min} - P_{gi})\right)\right|, & 2^{nd}\ \text{fuel}, P_{gi,1} < P_{gi} \le P_{gi,2}; \\ \cdots & \\ a_{ik}P_{gi}^2 + b_{ik}P_{gi} + c_{ik} + \left|e_{ik}\sin\left(f_{ik}(P_{gi,min} - P_{gi})\right)\right|, & k^{th}\ \text{fuel}, P_{gi,k-1} < P_{gi} \le P_{gi,k}. \end{cases} \quad (10)$$

For example, a combined cycle gas turbine (CCGT) plant has the capability to alternate between natural gas and oil as fuel sources, based on factors such as fuel supply and cost. The cost function provided captures the varying efficiency curves and associated costs of each fuel type.

D. Calculation of output power for SLACK unit

To ensure that the power balance Eq. (6) is met for the obtained solution, a slack generator is chosen so that its power output depends on that of the other $N_g - 1$ generators and the total load demand. The power output of this slack generator can be calculated using that of the other $N_g - 1$ generators as in Eq. (11) [91]:

$$P_{gS} = P_D + P_L - \sum_{\substack{i=1 \\ i\neq S}}^{N_g} P_{gi}, \quad (11)$$

where P_{gS} denotes the output power of the slack generator. By substituting Eq. (7) in Eq. (11):

$$P_L = B_{SS}P_{gS}^2 + P_{gS}\left(2\sum_{\substack{i=1 \\ i\neq S}}^{N_g} B_{Si}P_{gi} + B_{0S}\right) + \sum_{\substack{i=1 \\ i\neq S}}^{N_g}\sum_{\substack{h=1 \\ h\neq S}}^{N_g} P_{gi}B_{ih}P_{gh} + \sum_{\substack{i=1 \\ i\neq S}}^{N_g} B_{0i} + B_{00}. \quad (12)$$

Then, by putting (12) into (11), a quadratic relationship is resulted in the form:

$$A \cdot P_{gS}^2 + B \cdot P_{gS} + C = 0, \quad (13)$$

where A, B, and C can be calculated as follows:

$$A = B_{SS}, \quad (14)$$

$$B = 2\sum_{\substack{i=1 \\ i\neq S}}^{N_g} B_{Si}P_{gi} + B_{0S} - 1, \quad (15)$$

$$C = \sum_{\substack{i=1 \\ i\neq S}}^{N_g}\sum_{\substack{h=1 \\ h\neq S}}^{N_g} P_{gi}B_{ih}P_{gh} + \sum_{\substack{i=1 \\ i\neq S}}^{N_g} B_{0i} + B_{00} + P_D - \sum_{\substack{i=1 \\ i\neq S}}^{N_g} P_{gi}. \quad (16)$$

The output power of the slack generating unit is the positive root of (13). The two roots of (13) are calculated as below:

$$P_{gS} = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A}, \text{ where } B^2 - 4AC \ge 0. \quad (17)$$

Proposed Algorithm

One of the best ways to overcome the wide search space problems is the application of meta-heuristic algorithms. Among these types of algorithms inspired by nature, GWO is a well-known and novel method that has shown promising results. So, this paper presents a modified version of the GWO algorithm based on socio-political evolution modeling, named colonial competitive grey wolf optimizer (CCGWO).

A. Grey Wolf Optimizer

The GWO algorithm [101] is a recent swarm intelligence-based optimization algorithm inspired by the behavior of grey wolves in nature, which has been used dramatically by designers and engineers in recent years. The search process of GWO begins with randomly generating several grey wolves (candidate solutions). During the iterations, the most appropriate solution is considered alpha (α). Likewise, among the best solutions, the second and third ones are considered beta (β) and delta (δ), respectively. The remaining wolves are considered omega (ω). The distance of each candidate solution (e.g., the m^{th} one) from the prey is updated using the information of alpha, beta, and delta wolves as follows [101].

$$\vec{D}_m^{\alpha} = \left|\vec{C}_1 \odot \vec{X}_{\alpha} - \vec{X}_m\right|. \tag{18}$$

$$\vec{D}_m^{\beta} = \left|\vec{C}_2 \odot \vec{X}_{\beta} - \vec{X}_m\right|, \tag{19}$$

$$\vec{D}_m^{\delta} = \left|\vec{C}_3 \odot \vec{X}_{\delta} - \vec{X}_m\right|, \tag{20}$$

$$\vec{X}_m^{\alpha} = \vec{X}_{\alpha} - \vec{A}_1 \odot \vec{D}_m^{\alpha}, \tag{21}$$

$$\vec{X}_m^{\beta} = \vec{X}_{\beta} - \vec{A}_2 \odot \vec{D}_m^{\beta}, \tag{22}$$

$$\vec{X}_m^{\delta} = \vec{X}_{\delta} - \vec{A}_3 \odot \vec{D}_m^{\delta}, \tag{23}$$

$$\vec{X}_m(t+1) = \frac{1}{3}(\vec{X}_m^{\alpha} + \vec{X}_m^{\beta} + \vec{X}_m^{\delta}). \tag{24}$$

In Eq. (24), the parameter t stands for the current iteration of the algorithm, $\vec{A}_j$ and $\vec{C}_j$, j = 1, 2, 3, denote the coefficient vectors, and $\vec{X}_m(t)$ represents the position vector of the grey wolf m and $\odot$ is the Hadamard product (also known as the element-wise product or the element-by-element product) of two vectors of the same dimension: $\vec{A} \odot \vec{B} = (a^1, \dots, a^m) \odot (b^1, \dots, b^m) = (a^1 b^1, \dots, a^m b^m)$. Vectors $\vec{A}_j$ and $\vec{C}_j$ can be computed as:

$$\vec{A}_j = 2\vec{a} \odot \vec{r}_1 - \vec{a}, \tag{25}$$

$$\vec{C}_j = 2\vec{r}_2. \tag{26}$$

Where components of $\vec{a}$ decrease linearly from 2 to 0 during the iterations and $\vec{r}_1$ and $\vec{r}_2$ are randomly generated vectors from the uniform distribution in the range [0, 1]. It should be noted that in each iteration t, the value of $\vec{a}$ is the same for all solutions, but $\vec{A}_j$ and $\vec{C}_j$ are re-calculated for each solution by creating new random vectors $\vec{r}_1$ and $\vec{r}_2$.

B. Colonial competitive grey wolf optimizer

In the proposed CCGWO, the population is divided into several groups, and a colonial competition is performed among these groups. This way, the algorithm performs better in both the convergence rate and escaping from local optima .

Dividing the initial population between several groups:

To implement CCGWO, the initial population of wolves is divided into several groups. The total number of wolves is considered $Npop$, and the number of groups is represented by Ngr. After creating the initial population, they are arranged according to their objective function values, from the best to the worst. Then, the first Ngr wolves (the best wolves) are considered the alpha wolves, and one is assigned to any of Ngr groups. The remaining wolves are assigned one by one to these groups, and after reaching the last group, this process is restarted from the beginning. This process continues until all the wolves are assigned to any group. After

forming all groups, each group's second and third members, in terms of the value of the objective function, are considered beta and delta wolves.

New update equation:

In the original version of GWO, the hunting process is driven by alpha (α), beta (β), and delta (δ). Only these three solutions are stored, and the other individuals (omega wolves) are forced to update their positions using Eq. (24) according to the positions of the three best members. However, some responses already experienced by the other wolves may contain helpful information to get a better position. Therefore, in the proposed CCGWO algorithm, the best position that each wolf has ever experienced is stored as the best personal position $\vec{X}_{m,best}$, where $m = 1,2, \dots, Ng$. So, when updating the position of each wolf, its best personal position is used as follows:

$$\vec{D}_m^{PB} = \left|\vec{C}_4 \odot \vec{X}_{m,best} - \vec{X}_m\right|, \tag{27}$$

$$\vec{X}_m^{PB} = \vec{X}_{m,Best} - \vec{A}_4 \odot \vec{D}_m^{PB}. \tag{28}$$

It should be noted that Eq. (28) is the stored best personal position for the m^{th} wolf, equivalent to the best personal solution of the m^{th} particle in PSO. Now, the position of the m^{th} grey wolf from the n^{th} group is updated in each iteration as follows:

$$\vec{X}_m(t+1) = 0.5 \cdot \vec{X}_m^{\alpha} + 0.3 \cdot \vec{X}_m^{\beta} + 0.15 \cdot \vec{X}_m^{\delta} + \frac{0.05}{NP_n - 3} \sum_{m' \in \Omega} \vec{X}_{m'}^{PB}, \tag{29}$$

where $NP_n = Npop/Ng$ is the number of members of the n^{th} group, and Ω is the collection of omega wolves in the n^{th} group. It should be noted that since the top three wolves are alpha, beta, and delta in each group, the number of omega wolves in the n^{th} group is $NP_n - 3$. Therefore, the last correct expression of Eq. (29) is equal to the average of the best personal positions of all omega wolves in the n^{th} group with a coefficient of 0.05. As it can be deduced from Eq. (29), unlike Eq. (24), the weaker wolves are allowed to participate, albeit low, in determining their new position, which can lead to escape from local optimal solutions and enhance the exploration ability of GWO. Unlike the original GWO algorithm, the alpha, beta, and delta wolves do not have the same effects in determining the new position of each wolf in CCGWO. Appropriate weights were selected experimentally by testing different modes to determine each wolf's new position better.

Modeling colonial competition between grey wolf groups:

The power of each group is obtained from the following equation:

$$TP_n = 0.5 \cdot F\left(\vec{X}_n^{\alpha}\right) + 0.3 \cdot F\left(\vec{X}_n^{\beta}\right) + 0.15 \cdot F\left(\vec{X}_n^{\delta}\right) + \frac{0.05}{NP_n - 3} \sum_{m' \in \Omega} F(\vec{X}_{m'}), \tag{30}$$

where $F(\cdot)$ represents the value of the objective function and $\vec{X}_n^{\alpha}$, $\vec{X}_n^{\beta}$ and, $\vec{X}_n^{\delta}$ define the positions of alpha, beta, and delta of the n^{th} group, respectively.

Any group of wolves that cannot increase its power as well as its members' power and lose its competitive power will be gradually eliminated and disintegrated during the competition between groups. This approach means that weaker groups with weaker alpha wolves gradually lose their members during the iterations, from the weakest to the alpha wolf. The strongest group takes over these groups and increases the power of its colony. This process in the CCGWO algorithm is modeled in this way: the powers of all groups are obtained and sorted from the weakest to the strongest groups in each iteration. Then, the strongest group takes the weakest member of the weakest group in that iteration, which is executed until the last iteration. During the competition, if only alpha, beta, and delta are left for a group, the new position of the members and power will be calculated using Eqs. (31) and (32) instead of Eqs. (29) and (30). Due to the absence of omega wolves, there will no longer be an average expression.

$$\vec{X}_m(t+1) = 0.5 \cdot \vec{X}_m^{\alpha} + 0.3 \cdot \vec{X}_m^{\beta} + 0.2 \cdot \vec{X}_m^{\delta}\ , \tag{31}$$

$$TP_n = 0.5 \cdot F\left(\vec{X}_m^{\alpha}\right) + 0.3 \cdot F\left(\vec{X}_m^{\beta}\right) + 0.2 \cdot F\left(\vec{X}_m^{\delta}\right)\ . \tag{32}$$

If two members, alpha and beta, remain in the group, the above equations will be changed as follows:

$$\vec{X}_m(t+1) = 0.6 \cdot \vec{X}_m^{\alpha} + 0.4 \cdot \vec{X}_m^{\beta}, \tag{33}$$

$$TP_n = 0.6 \cdot F\left(\vec{X}_m^{\alpha}\right) + 0.4 \cdot F\left(\vec{X}_m^{\beta}\right). \tag{34}$$

Finally, if a group has only one member in an iteration (clearly, this single wolf is the alpha wolf) and it also becomes the weakest group, the group will be destroyed, and that alpha wolf will be given to the strongest group as a colony.

Convergence Analysis

Convergence in metaheuristic algorithms refers to the gradual movement of population members toward an optimal point—or a set of near-optimal solutions—over successive iterations. In the GWO, each agent updates its position based on the weighted average of the three leading wolves: alpha, beta, and delta. This mechanism ensures that the entire population systematically gravitates toward the best regions identified during the search. The proposed CCGWO algorithm maintains this fundamental structure, incorporating chaotic maps at specific steps primarily to enhance diversity and prevent stagnation.

From a theoretical standpoint, the convergence behavior is largely governed by the parameter $\vec{a}$. This parameter, which appears in the position update equations, is linearly decreased from 2 to 0 throughout the iterations. During the initial stages, when $\vec{a}$ is relatively large, the search agents have a wider range of movement, allowing the algorithm to focus on exploration (global search). As $\vec{a}$ diminishes, the step sizes become shorter, and the search gradually transitions from global exploration to exploitation (local refinement). Finally, as $\vec{a}$ approaches zero, all population members tend to cluster in the close vicinity of the best-known positions. This gradual and linear reduction of $\vec{a}$ ensures that the algorithm achieves stable convergence within a finite number of iterations. Another critical aspect of CCGWO is the role of the internal crossover operator. While the movement toward the leaders accelerates convergence, the crossover operator injects new information into the population by randomly combining components from different wolves. This dual strategy—balancing systematic convergence with diversity-enhancing mechanisms—allows the algorithm to move reliably toward the global optimum without becoming trapped in local optima. Consequently, the convergence rate remains balanced; it is neither too slow to be inefficient nor too fast to risk premature convergence.

To empirically evaluate the stability and reliability of the method, the algorithm was executed 25 times for each test system using different random initial populations. The statistical results, including the best, worst, and mean costs, reveal very low dispersion across these runs. For instance, the difference between the best and worst solutions remains below 0.2% in all cases, with the standard deviation typically staying under 0.1% of the mean. Such minimal variance indicates that the algorithm is robust against changes in the starting points and that its outputs are highly repeatable. Furthermore, the convergence curves presented in Figures 1, 3, and 4 clearly illustrate the smooth and monotonic decrease in the objective function until a steady state is reached in the final iterations. In summary, both the theoretical analysis of the position update mechanism and the empirical evidence from multiple runs confirm that CCGWO possesses stable convergence properties, reliable behavior, and sufficient resistance to stochastic disturbances.

Computational Complexity Analysis

The efficiency of a metaheuristic algorithm is not determined solely by the quality of its final solutions; its computational cost is also a crucial factor, particularly for large-scale problems where even small reductions in runtime are important. In this section, the computational complexity of the CCGWO algorithm is analyzed step by step. First, consider the CCGWO. Let $Npop$ denote the population size (number of wolves) and *D* represent the number of decision variables (problem dimension). In each iteration, three main stages are performed: (1) evaluation of the objective function for all individuals, (2) sorting the population to identify the alpha, beta, and delta wolves, and (3) updating the positions of all wolves based on the positions of these three leaders. The computational cost of these stages is $O\ (Npop\ *\ D)$ for the evaluation step (assuming the objective function requires $O\ (D)$ operations), $O\ (Npop\ *\ logNpop)$ for sorting, and $O\ (Npop\ *\ D)$ for the position update step. Therefore, the overall complexity of each CCGWO iteration can be expressed as $O\ (Npop\ *\ D + Npop\ *\ logNpop)$. For practical comparison, the average runtime of 25 independent runs was measured on the 140‑unit test system. The results show that CCGWO requires only about 3% more execution time than the standard GWO, mainly due to the is related to dividing the population between several groups and applying colonial competition between these groups. Note that in the multi-group case, we need a smaller population than in the single-group case, which causes the execution time and computational complexity of the proposed algorithm to be even reduced compared to the original algorithm. Overall, CCGWO achieves a noticeable improvement in solution quality and convergence

stability without introducing a significant increase in computational cost. This property makes the algorithm a suitable candidate for practical applications where both solution accuracy and computational efficiency are essential.

Algorithm Parameter Settings and Justification

The performance of any metaheuristic algorithm largely depends on the appropriate selection of its parameters. In this section, all parameters of the CCGWO algorithm are presented together, and the rationale behind each selected value is explained. The parameters are divided into two main categories: parameters inherited from the standard GWO, CGWO and parameters specific to CCGWO.

This category includes the parameters adopted from the original GWO algorithm, whose values are set according to the recommendations of the reference paper [101].

Population size ($Npop$):

This parameter determines the number of wolves in the population at each iteration. Increasing the population size improves the algorithm's ability to explore promising regions of the search space but also increases computational cost. Following the recommendation of [101] and considering the scale of the test systems, the population size was set to 30 for the 6-unit and 15-unit systems, 60 for the 110-unit system, and 90 for the 140-unit system. This choice provides a reasonable balance between search accuracy and computational efficiency.

Maximum number of iterations ($Iter_{max}$):

This parameter defines the stopping criterion of the algorithm. It should be large enough to allow sufficient convergence while remaining computationally practical. The maximum number of iterations was set to 100 and 500 for the smaller systems (6 and 15 units), 3000 for the 110-unit system, and 5000 for the 140-unit system. These values were selected based on preliminary observations of convergence curves, which showed that the algorithm typically converged well before reaching the maximum iteration limit.

Decreasing vector $\vec{a}$:

The components of this vector decrease linearly from 2 to 0 over the course of the iterations. This gradual reduction forms the core mechanism that balances exploration and exploitation in GWO. Larger values of ($\vec{a}$) in the early iterations allow wider exploration of the search space, whereas smaller values in later iterations guide the wolves toward the neighborhood of the best solutions. The linear reduction formula used in the original GWO algorithm [101] is adopted here without modification.

Random coefficient vectors ($\vec{r}_1$) and ($\vec{r}_2$):

These vectors consist of random values drawn from a uniform distribution in the interval ([0,1]). They introduce stochasticity into the search process and prevent the algorithm from behaving in a fully deterministic manner, which could otherwise lead to premature convergence. The use of a uniform distribution for these coefficients is standard practice in the literature.

Exploration–Exploitation Balance Analysis

A metaheuristic algorithm is successful only if it neither turns into blind trial-and-error (inefficient exploration) nor settles too quickly at the first promising point it finds (premature exploitation). In CCGWO, this balance is achieved through three main levers. First, the vector ($\vec{a}$), whose value decreases linearly from 2 to 0: when ($\vec{a}$>1), the wolves move away from the prey (exploration), and when ($\vec{a}$<1), they move closer to it (exploitation). Second, the multi-group structure: during the first half of the search, each group explores a different region independently, which helps keep diversity high across the whole population. Third, competition: at each iteration, the top-performing wolves are selected, and this selection pressure gradually drives the population toward more promising areas . To quantify this behavior, the population diversity metric (defined as the average distance of individuals from the centroid) was computed across iterations. The corresponding curve shows that, unlike the standard GWO, CCGWO preserves diversity for a longer period and then enters the exploitation phase with a controlled decline—neither collapsing abruptly nor wandering without direction.

Diversity Analysis

Population diversity is one of the key factors behind the success of metaheuristic algorithms. If diversity disappears too early, the algorithm tends to converge prematurely and becomes trapped in local optima. On the other hand, if diversity remains excessively high, convergence becomes difficult. CCGWO maintains a proper balance through three main mechanisms:

Population grouping: In the standard GWO, all wolves update their positions based on the same three leaders—alpha, beta, and delta. This single-center structure causes the population to gather rapidly around the leaders, which reduces diversity. In CCGWO, however, the population is divided into several independent groups, each with its own alpha, beta, and delta leaders. In other words, instead of one center of attraction, there are multiple attraction centers active at the same time in different regions of the search space. As a result, wolves gather around several distributed regions, and diversity across the whole population is preserved for a longer time.

Imperialist competition: Static grouping alone may gradually cause some groups to become stagnant. To avoid this, CCGWO performs an imperialist competition stage at each iteration. The groups are ranked based on their power, which is calculated from Eq. 30 using a weighted combination of the objective-function values of the leaders and omega wolves. The strongest group then takes over the weakest member of the weakest group. Once the weakest group loses all its members, it is removed from the population, and its leader is transferred to the strongest group. This dynamic redistribution has two benefits. First, strong groups receive fresh genetic material, which reduces internal homogeneity. Second, weak groups are removed, so computational effort is not wasted on low-potential regions. In this way, diversity is preserved without sacrificing efficiency.

Omega participation in updating: In the standard GWO, only alpha, beta, and delta contribute to shaping the new position, while omega wolves simply follow. In CCGWO, however, Eq. 29 includes a contribution from the mean personal best of the omega wolves. Although this contribution is small (with a weight of 0.05), it ensures that all members of the group play a role in determining the search direction, rather than making the new position solely a weighted average of the leaders. In practical terms, even if the leaders of a group move close to one another, the omega contribution helps prevent a complete loss of diversity within that group.

To evaluate these claims quantitatively, a suitable metric is the average Euclidean distance of population members from the center of mass.

The pseudocode for solving the ELD problem using the CCGWO algorithm is given by:

Pseudocode 1: Pseudocode of solving the ELD problem using the CCGWO algorithm

1) Set to $Iter_{max}$ the maximum number of iterations, set parameters $Npop$ and Ngr. Initialize a random population of grey wolves $\vec{X}_m$ $(m = 1, 2, \dots, Npop)$.

2) Evaluate the fitness value of all search agents (i.e., compute the cost function considering the ELD constraints for each grey wolf).

3) Select NG of the strongest wolves as the initial alpha wolves $\vec{X}_\alpha$ and form the Ngr groups with them.

4) Distribute the rest of the initial population between Ngr groups.

5) Save the positions of all wolves.

6) For each group of grey wolves specify the positions of $\vec{X}_\beta$ and $\vec{X}_\delta$.

7) $t := 1$

8) While $t < Iter_{max}$ perform steps 9) to 17)

9) For each group of grey wolves, perform steps 10) to 15)

10) For each member of any group, perform steps 11) to 14)

11) Update parameters $\vec{a}$, $\vec{A}_j$, $\vec{C}_j$

12) Update the current position of all wolves.

13) Evaluate the fitness value of all search agents, i.e., compute the cost function considering the ELD constraints for each new grey wolf.

14) Update their best position.

15) Update the value of $\vec{X}_\alpha, \vec{X}_\beta, \vec{X}_\delta$.

16) Calculate the power of each group and perform colonial competition between the grey wolves' groups. Give the weakest member of the weakest group to the strongest group, and if a group has only an alpha, delete that group and give the alpha as a colony to the strongest group.

17) $t := t + 1$

18) Print the strongest alpha wolf among all groups as the best possible solution.

Simulation results of ELD problem

In this part of the study, the 6-, 15-, 110-, and 140-generator test power systems, the most common systems used in recent articles and can cover all the various aspects of the ELD problem, are selected for simulations. All simulations' algorithm parameters are chosen based on the original GWO algorithm suggested in [101]. The number of population members is set to 30, 30, 60, and 90, and they are used for 6-, 15-, 110-, and 140-generator test power systems, respectively. For each scenario, 25 independent runs are performed to obtain the best cost value, average cost value, maximum cost value, and standard deviation indices. The details of the studied networks, as well as the systems and software used, are briefly provided in **Appendix A**.1.

To evaluate the performance of the algorithms in the ELD problem, we considered four primary statistical metrics. The Best Cost represents the minimum objective function value obtained over 25 independent runs, directly reflecting the algorithm's ability to converge toward the global optimum. The Mean Cost, calculated as the average of all runs, provides a realistic assessment of the algorithm's general performance, stripping away any "luck" factor associated with a single successful run—a critical consideration given the stochastic nature of metaheuristic algorithms. Meanwhile, the Max Cost highlights the worst-case scenario encountered, serving as a measure of the algorithm's robustness against undesirable fluctuations and premature convergence to local optima. Finally, the Standard Deviation (Std.) quantifies the dispersion of results around the mean; this is a key indicator for assessing the stability and reliability of the algorithm in practical power system applications. By employing these four standard metrics, which are widely recognized in ELD literature, we aim to provide a comprehensive analysis of each algorithm's accuracy, stability, and overall robustness.

A. Case A: The 6-unit test system

The test system with 6 generator units and the total power demand of $P_D = 1{,}263$ MW is considered in this section [61]. The ELD problem limits include the ramp rate, POZ, and the system's real power losses P_L.

Tables I and II present the statistical results of solving the ELD problem in test system Case A and the best output power for each generator unit, respectively. Table I shows the statistical results in indicators of the best (Min. cost), the average (Mean cost), the maximum (Max. cost), and the standard deviation (Std.) of the objective function values for GWO, CGWO (cooperative multi-group version), CCGWO, and other algorithms used for comparison.

Table I shows that the proposed CCGWO algorithm gives the best results for this problem in terms of all statistical indicators among the other methods, which indicates the optimal performance and robustness of the method in Case A. Also, the convergence diagram of GWO and CCGWO algorithms for solving this problem is shown in Figure 1. The better convergence characteristic of CCGWO follows from Figure 1.

TABLE I

THE BEST STATISTICAL RESULTS OF VARIOUS ALGORITHMS AND THE PROPOSED CCGWO FOR THE 6-UNIT TEST SYSTEM.

Methods (*Npop/Iter*$_{max}$)	**Min. cost ($/h)**	**Mean cost ($/h)**	**Max. cost ($/h)**	**Std. ($/h)**
RDPSO [62]	15,442.7575	15,445.0245	15,455.2936	2.2828
HPSOWM [62]	15,442.8205	15,455.6220	15,502.6333	15.8867
aBBOmDE [23]	15,442.673	15,442.83	15,442.9930	-
BFO [26]	15,443.8164	15,446.95383	-	-
ESAHJ [89] (50/200)	15,442.80	-	-	-
ISSO [16] (20/100)	15,443.0749	15,443.077	15,443.12	-
BBO [26]	15,443.096	15,443.096	15,443.096	-
Jaya-SML [46] (30/2000)	15,445.1682	15,447.2910	15,450.6497	6.2214
GWO [9]	15,443.0	-	-	-
MCSA [58]	15,449.8995	15,449.8995	15,449.8995	1.6404e-11
KHA-IV [71] (50/100)	15,443.0752	15,443.1863	15,443.3265	-
GC-Jaya [45] (30/2000)	15,443.3002	15,454.0331	15,468.7704	9.62
IJaya [45] (30/2000)	15,443.0881	15446.3730	15448.6353	5.30
SOA [50]	15,443.2106	15,444.8813	-	-
HBB-BC [103] (100/100)	15,444.26	15,446.46	15,448.89	1.52
SOH-PSO [28] (-/120)	15,446.02	15,497.35	15,609.64	-
GWO (30/100)	15,443.1072	15,443.8303	15,447.2749	8.45
CGWO (30/100)	15,443.1567	15,443.7906	15,446.0315	5.39
CCGWO (30/100)	**15,442.6609**	**15,442.6675**	**15,442.6782**	**0.0037**

TABLE II

THE BEST FINAL SOLUTION OF THE PROPOSED CCGWO FOR THE 6-UNIT TEST SYSTEM (IN MW).

Variable	**CCGWO**	**IBSA** [104]	**NPSO-LRS** [105]
P_{g1}	**447.0176**	447.48	446.96
P_{g2}	**173.2144**	173.30	173.39
P_{g3}	**263.9508**	263.44	262.34
P_{g4}	**139.0735**	139.05	139.51
P_{g5}	**165.5371**	165.46	164.70
P_{g6}	**86.6218**	87.12	89.01
P_L	**12.4152**	12.95	12.93
Cost ($/h)	**15,442.6609**	15,448.98	15,450.00

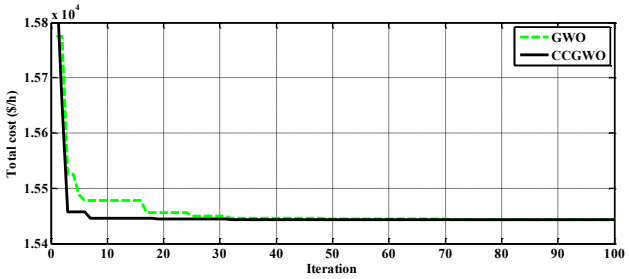


Figure 1. The convergence curve of the GWO and CCGWO algorithms for the 6-unit test system.

B. Case B: The 15-unit test system

Case B contains 15 thermal generator units with $P_D = 2{,}630$ MW. The other information about this case is presented in [61]. The ELD problem limits are the same as in case A. Two ELD problem scenarios based on those limits are simulated. Type-1 considers only the POZ constraint, and Type-2 runs with both POZ and ramp rate limit.

The statistical results of solving the ELD problem of Types 1 and 2 in test system Case B are presented in Tables III and IV, respectively. As can be seen from Table III, the proposed CCGWO algorithm gives the best results for Type-1 problem in this test system among the methods presented in the articles. The results in Table IV show that although the proposed method does not get the best result for the type 2 problem, it is ranked third in terms of the objective function value. It should be mentioned that the proposed method, in both types of problems mentioned in this test system, is completely superior to the original GWO and CGWO algorithms. Also, the best results obtained for Types 1 and 2 are presented in Table V. The convergence diagram of GWO and CCGWO algorithms in solving the Type-1 problem of test system Case B is shown in Figure 2, which shows the better convergence characteristic of CCGWO compared with GWO.

TABLE III
THE BEST STATISTICAL RESULTS OF VARIOUS ALGORITHMS FOR THE 15-UNIT TEST SYSTEM WITHOUT RAMP RATE LIMITS (TYPE 1)

Methods (*Npop/Iter$_{max}$*)	**Min. cost ($/h)**	**Mean cost ($/h)**	**Max. cost ($/h)**	**Std. ($/h)**
MICA [29] (100/600)	32,544.559	32,546.0017	32,548.2506	3.416
KHA-IV [78] (50/100)	32,547.3700	32,548.1348	32,548.9326	-
ESAHJ [102] (50/3000)	32,568.12	-	-	-
CIHBMO [55] (-/100)	32,548.58587	32,548.58587	32,548.58587	-
SSOA [70]	32,731.6903	32,845.5416	32,930.9734	-
CSA [23] (10/300)	32,544.9704	32,545.0068	32,546.6734	0.2386
CSO [25] (30/200)	32,588.9189	32,679.8775	32,796.7792	-
GWO (30/500)	32,564.3248	32,581.1020	32,869.5826	376.15
CGWO (30/500)	32,566.2614	32,578.6237	32,852.0954	264.38
CCGWO (30/500)	**32,543.7457**	**32,544.0920**	**32,544.7398**	**0.0854**

TABLE IV
THE BEST STATISTICAL RESULTS OF VARIOUS ALGORITHMS FOR THE 15-UNIT TEST SYSTEM WITH RAMP RATE LIMITS (TYPE 2)

Methods (*Npop/Iter$_{max}$*)	**Min. cost ($/h)**	**Mean cost ($/h)**	**Max. cost ($/h)**	**Std. ($/h)**
SSOA [70]	32,819.1554	32,902.1631	32,999.4008	-
BFO [30] (50/200)	32,784.5024	32,976.81	-	85.7743
NhFA-Rnp [32] (75/1000)	32,697.91	32,700.56	32,709.94	2.64
EMA [48] (-/1000)	32,704.4503	32,704.4504	32,704.4506	-
SOH-PSO [28] (500/125)	32,751.0	32,878.0	32,945.0	-
WCA [73]	32,704.44	32,704.50	32,704.51	4.513e-05
MsEBBO [45] (60/300)	32,692.3972	32,692.3973	32,692.3975	6.09e-005
IODPSO-L [71] (60/1000)	32,692.39	32,692.39	32,692.39	0.0
DSOS [65]	32,706.76107	32,706.89931	32,709.9665	0.45737
C-MIMO-CSO [106] (50/100)	32,701.21	32,701.21	32,701.22	0.0058
CCSO [42] (20/1000)	32,706.64	32,706.64	32,706.64	0.0007
mQOGWO [107]	32,692.23	32,692.40	32,692.60	1.09
GWO (30/500)	32,746.8011	32,895.0148	32,975.2751	816.43
CGWO (30/500)	32,752.6900	32,883.3025	32,953.1157	639.19
CCGWO (30/500)	**32,687.1325**	**32,689.0065**	**32,691.8215**	**0.6023**

TABLE V
THE BEST FINAL SOLUTIONS OF THE CCGWO, CIHBMO [55], AND CMICA/M6 [29] ALGORITHMS FOR THE 15-UNIT TEST SYSTEM FOR TYPE 1 AND FOR THE CCGWO, IJAYA [49], AND Artificial Electric Field Algorithm (AEFA) [108] ALGORITHMS FOR THE 15-UNIT TEST SYSTEM FOR TYPE 2 (IN MW)

Variable	TYPE 1			TYPE 2		
	CCGWO	CIHBMO [55]	CMICA/M6 [29]	**CCGWO**	IJaya [49]	AEFA [108]
P_{g1}	**454.7678**	454.9999	455	**454.9808**	455	455
P_{g2}	**455.0000**	454.9999	455	**380.0000**	380	380
P_{g3}	**130.0000**	130	130	**130.0000**	130	130
P_{g4}	**130.0000**	130	130	**130.0000**	130	130
P_{g5}	**231.5964**	234.2005	229.6902	**170.0000**	170	170
P_{g6}	**460.0000**	460	456.5507	**460.0000**	460	460
P_{g7}	**465.0000**	464.9999	465	**430.0000**	430	430
P_{g8}	**60.0203**	60	60	**67.6718**	71.8261	71.429057
P_{g9}	**25.1497**	25	25	**58.8697**	59.0248	58.596432
P_{g10}	**35.5565**	30.9939	41.1988	**159.9999**	160	160
P_{g11}	**75.0242**	76.7014	73.8171	**80.0000**	80	80
P_{g12}	**79.0903**	79.9999	80	**80.0000**	80	80
P_{g13}	**25.0454**	25	25	**25.0007**	25	25
P_{g14}	**15.0478**	15	15	**17.3986**	15	15
P_{g15}	**15.0361**	15	15	**15.0032**	15	15
P_L	**26.3345**	26.8959	26.2635	**28.9247**	30.8568	–
Cost ($/h)	**32,543.7457**	32,548.58587	32,544.5605	**32,687.1325**	32,706.6218	32,697.2819

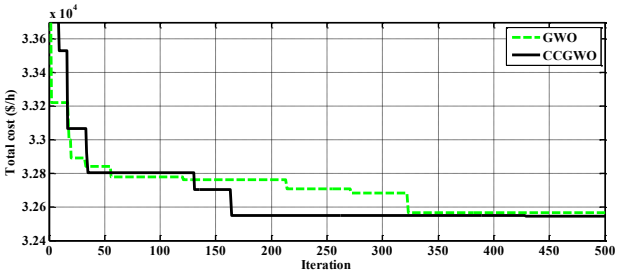


Figure 2. The convergence of GWO and CCGWO algorithms in solving type 1 problem of test system case B.

C. Case C: The 110-unit test system

Case C is a large-scale power system containing quadratic fuel cost function characteristics for 110 generators [87]. The total load demand is $P_D = 15{,}000$ MW.

The statistical results of solving the ELD problem in the case C test system are presented in Table VI, and the best result for this problem is presented in Table VII. It is clear from Table VI that the proposed CCGWO algorithm has the best performance and obtained better statistical indices compared with the other methods, which indicates the optimal performance and robustness of CCGWO. Figure 3 displays the convergence diagrams of the GWO and CCGWO algorithms in solving this problem, clearly showing the better convergence characteristics of CCGWO compared with GWO.

TABLE VI
THE BEST FINAL SOLUTION OF THE CCGWO AND ORCCRO [56] ALGORITHMS FOR THE 110-UNIT TEST SYSTEM (IN MW).

P_{g1}~	CCGWO	2.4001	2.4002	2.4002	2.4001	2.4001	4.0001	4.0002	4.0000	4.0000	64.3190
P_{g10}	ORCCRO	2.40	2.42	2.42	2.44	2.44	4.01	4.02	4.01	4.00	66.59
P_{g11}~	CCGWO	61.9304	36.3154	57.4634	25.0008	25.0010	25.0010	154.9993	154.9998	154.9999	154.9996
P_{g20}	ORCCRO	59.25	30.23	52.47	25.00	25.14	25.02	154.91	154.99	154.52	154.82
P_{g21}~	CCGWO	68.9002	68.9001	68.9001	349.9999	400.0000	399.9997	499.9979	499.9995	199.9985	99.9997
P_{g30}	ORCCRO	68.93	68.98	69.00	349.872545	400.00	400.00	500.00	500.00	200.00	99.54
P_{g31}~	CCGWO	10.0001	19.9993	79.7789	249.9992	359.9989	399.9999	39.9998	69.9964	99.9995	119.9762
P_{g40}	ORCCRO	10.00	19.99	79.72	249.79	359.98	400.00	39.41	69.73	99.99	119.70
P_{g41}~	CCGWO	157.2236	219.9977	439.9996	559.9993	659.9998	616.2407	5.4000	5.4003	8.4000	8.4002
P_{g50}	ORCCRO	157.90	219.57	439.98	559.96	660.00	604.13	5.44	5.40	8.4000	8.4000

P_{g51}~	CCGWO	8.4001	12.0000	12.0001	12.0001	12.0001	25.2000	25.2047	35.0005	35.0011	45.0001
P_{g60}	ORCCRO	8.64	12.06	12.00	12.00	12.00	25.24	25.93	35.07	35.08	45.11
P_{g61}~	CCGWO	45.0006	45.0010	184.9986	185.0000	184.9979	184.9988	70.0002	70.0000	70.0000	359.9973
P_{g70}	ORCCRO	45.10	45.18	184.96	184.22	184.64	185.00	70.01	70.01	70.01	359.99
P_{g71}~	CCGWO	399.9991	399.9993	104.7277	191.3450	89.9990	49.9999	160.0041	295.1338	175.4212	98.1541
P_{g80}	ORCCRO	399.96	399.87	96.85	201.25	89.99	49.99	161.01	284.23	183.91	107.53
P_{g81}~	CCGWO	10.0002	12.0120	20.1180	199.9951	324.8952	439.9986	14.3350	24.3775	83.0492	89.0896
P_{g90}	ORCCRO	10.07	12.05	21.95	199.22	324.52	439.98	32.33	22.30	86.67	92.52
P_{g91}~	CCGWO	57.6596	100.0000	439.9996	500.0000	599.9988	471.3491	3.6001	3.6001	4.4003	4.4000
P_{g100}	ORCCRO	59.67	98.59	439.98	499.99	599.95	462.47	3.63	3.67	4.41	4.41
P_{g101}~	CCGWO	10.0001	10.0003	20.0004	20.0001	40.0000	40.0002	50.0001	30.0002	40.0000	20.0002
P_{g110}	ORCCRO	10.00	10.05	20.11	20.00	40.01	40.01	50.01	30.03	40.06	20.02
	CCGWO	**Total power output**		**15,000.0 MW**		**Fuel total cost**			**197,988.3384 $/h**		
	ORCCRO			**15,000.0 MW**					**198,016.29 $/h**		

TABLE VII

THE BEST STATISTICAL RESULTS OF VARIOUS ALGORITHMS FOR THE 110-UNIT TEST SYSTEM WITH RAMP RATE LIMITS ($P_D = 15{,}000$ MW)

Methods (*Npop/Iter$_{max}$*)	**Min. cost ($/h)**	**Mean cost ($/h)**	**Max. cost ($/h)**	**Std. ($/h)**
CFLBO [5]	197,988.4112	197,988.4593	197,988.8342	0.008
OIWO [109] (20/250)	197,989.14	197,989.41	197,989.93	-
BBO [56] (50/400)	198,241.1660	198,413.4500	199,102.5900	-
DE/BBO [56] (50/400)	198,231.0600	198,326.6600	198,828.5700	-
ORCCRO [56] (50/400)	198,016.2900	198,016.3200	198,016.8900	-
SA [110], [111] (-/1000)	216,100.5475	216,365.7269	216,823.5408	-
SAB [110], [111] (-/1000)	206,921.9057	207,764.7398	208,197.0059	-
SAF [110], [111] (-/1000)	207,380.5164	207,813.3717	208,012.6248	-
IWO [112] (50/2000)	198,252.3594	198,621.3233	198,902.7697	138.47
GWO (60/3000)	198,084.6564	198,126.5799	198,843.3092	749.37
CGWO (60/3000)	198,125.6842	198,270.3342	198,885.8561	906.23
CCGWO (60/3000)	**197,988.3384**	**197,988.4648**	**197,988.7505**	**0.0025**

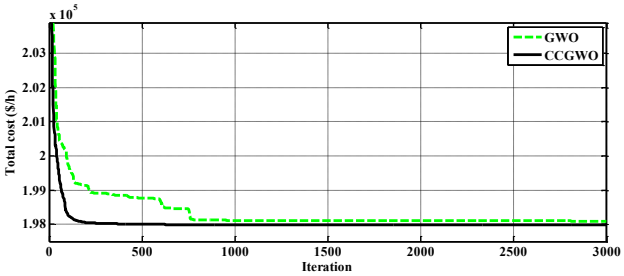


Figure 3. The convergence curve of the GWO and CCGWO algorithms for the 110-unit test system.

D. Case D: The 140-unit test system

Case D includes a real and comprehensive system containing all kinds of constraints mentioned in the paper for 140 generators. The total load demand is $P_D = 49{,}342$ MW. The other power system information is described in [113].

In general, as the size of an electrical system grows, its complexity increases non-linearly. So, optimal design for such systems is always challenging and requires a robust optimization algorithm.

The statistical results of solving the ELD problem in the large-scale test system Case D using the aforementioned methods are presented in Table VIII, and the best results from the proposed method are given in Table IX. According to Table VIII, the best optimal results have been obtained by the proposed CCGWO algorithm, which indicates the optimization power of this

method. Additionally, the statistical indices obtained from CCGWO show its robustness in achieving the optimal solution. It should be noted that the superior performance of CCGWO compared with GWO and CGWO indicates the efficiency of the modifications accomplished to achieve a more powerful and robust method. Figure 4 shows the convergence diagrams of GWO and CCGWO in solving Case D test system, again confirming the better convergence performance of CCGWO.

TABLE VIII

COMPARISON OF THE BEST RESULTS OBTAINED BY DIFFERENT ALGORITHMS FOR THE 140-UNIT TEST SYSTEM

Methods (*Npop/Iter$_{max}$*)	**Min. cost ($/h)**	**Max. cost ($/h)**	**Mean cost ($/h)**	**Std. ($/h)**
CCPSO [113] (30/10000)	1,657,962.73	1,657,962.73	1,657,962.73	0.0
SSA [87] (50/1000)	1,658,384.88	1,658,386.57	1,658,384.25	0.10
IL-SHADE [47] (-/6500)	1,657,962.7303	1,658,090.54	1,657,965.30	-
CCEDE [98] (100/4000)	1,657,962.7	1,657,965.18	1,657,963.05	1.1466
AGWO [63]	1,657,962.0	1,657,964.0	1,657,963.0	0.54820
C-GRASP–SaDE [64] (20/11250)	1,657,962.7268	1,658,583.5267	1,658,006.2712	-
ORCCRO [56] (50/1000)	1,659,654.83	1,659,823.97	1,659,725.96	0.16
HPSO-ICA [44] (200/250)	1,747,466.0	1,783,367.0	-	17,168.0
CQGSO [72]	1,657,962.7270	1,657,962.7760	1,657,962.7410	-
QOGWO [107]	1,657,962.73	1,657,969.23	1,657,975.73	4.03
MFPA [114]	1,657,962.6907	1,657,962.8243	1,657,962.7506	0.05557
GWO (90/5000)	1,678,808.5745	1,811,261.0029	1,702,107.3528	2.318e+04
CGWO (90/5000)	1,658,815.1062	1,820,149.7536	1,735,256.9919	3.406e+04
CCGWO (90/5000)	**1,657,960.0099**	**1,657,961.5846**	**1,657,960.8251**	**0.0847**

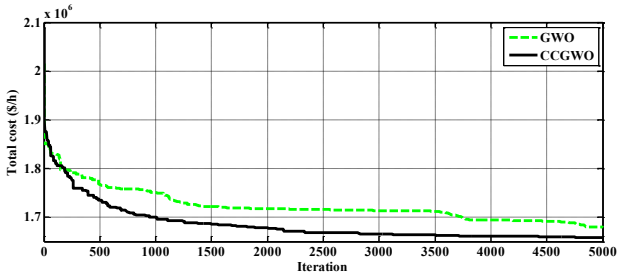


Figure 4. The convergence curve of GWO and CCGWO algorithms for the 140-unit system.

TABLE IX

THE BEST SOLUTIONS OBTAINED FOR THE 140-UNIT TEST SYSTEM (IN MW) BY CCGWO AND CCEDE [98].

P_{g1}~ P_{g10}	CCGWO	119.0000	164.0000	190.0000	190.0000	168.4991	190.0000	490.0000	490.0000	496.0000	496.0000
	CCEDE	118.9999	164.0	190.0	190.0	168.5398	190.0	490.0	490.0	496.0	496.0
P_{g11}~ P_{g20}	CCGWO	496.0	496.0	506.0	509.0	506.0	505.0	506.0	506.0	505.0	505.0
	CCEDE	496.0	496.0	506.0	509.0	506.0	505.0	506.0	506.0	505.0	505.0
P_{g21}~ P_{g30}	CCGWO	505.0	505.0	505.0	505.0	537.0	537.0	549.0	549.0	501.0	499.0
	CCEDE	505.0	505.0	505.0	505.0	537.0	537.0	549.0	549.0	501.0	499.0
P_{g31}~ P_{g40}	CCGWO	506.0	506.0	506.0	506.0	500.0	500.0	241.0	241.0	774.0	769.0
	CCEDE	506.0	506.0	506.0	506.0	500.0	500.0	241.0	241.0	774.0	769.0
P_{g41}~ P_{g50}	CCGWO	3.0	3.0	250.0	250.0	250.0	250.0	250.0	250.0	250.0	250.0
	CCEDE	3.0	3.0	250.0	250.0	250.0	250.0	250.0	250.0	250.0	250.0
P_{g51}~ P_{g60}	CCGWO	165.0	165.0	165.0	165.0	180.0	180.0	103.0	198.0	312.0	308.5084
	CCEDE	165.0	165.0	165.0	165.0	180.0	180.0	103.0	198.0	312.0	308.4685
P_{g61}~ P_{g70}	CCGWO	163.0	95.0	511.0	511.0	490.0	256.9930	490.0	490.0	130.0	339.4395
	CCEDE	163.0	95.0	511.0	511.0	490.0	256.9923	490.0	490.0	130.0	339.4395
P_{g71}~ P_{g80}	CCGWO	141.6076	388.3280	195.0004	200.0508	194.9316	257.1706	398.9960	330.0	531.0000	531.0000
	CCEDE	141.6082	388.3274	195.0007	200.0504	194.9317	257.1705	398.9961	330.0	531.0	531.0
P_{g81}~ P_{g90}	CCGWO	542.0	56.0	115.0	115.0	115.0	207.0	207.0	175.0	175.0	180.3000
	CCEDE	542.0	56.0	115.0	115.0	115.0	207.0	207.0	175.0	175.0	180.375
P_{g91}~ P_{g100}	CCGWO	175.0	575.4750	547.5000	836.8	837.5	682.0	720.0	718.0	720.0	964.0
	CCEDE	175.0	575.4	547.5	836.8	837.5	682.0	720.0	718.0	720.0	964.0

P_{g101}~ P_{g110}	CCGWO	958.0	947.9	934.0	935.0	876.5	880.9	873.7002	877.3998	871.71	864.7900
	CCEDE	958.0	947.9	934.0	935.0	876.5	880.9	873.7	877.4	871.7	864.8
P_{g111}~ P_{g120}	CCGWO	882.0	94.0	94.0	94.0	244.0	244.0	244.0	95.0	95.0	116.0
	CCEDE	882.0	94.0	94.0	94.0	244.0	244.0	244.0	95.0	95.0	116.0
P_{g121}~ P_{g130}	CCGWO	175.0	2.0	4.0	15.0	9.0	12.0	10.0	112.0	4.0	5.0
	CCEDE	175.0	2.0	4.0	15.0	9.0	12.0	10.0	112.0	4.0	5.0
P_{g131}~ P_{g140}	CCGWO	5.0	50.0	5.0	42.0	42.0	41.0	17.0	7.0	7.0	26.0
	CCEDE	5.0	50.0	5.0	42.0	42.0	41.0	17.0	7.0	7.0	26.0
	CCGWO	Total power output		49, 342.0 MW		Total fuel cost			1,657,960.0099$/h		
	CCEDE			49, 342.0 MW					1,657,962.7 $/h		

E. DISCUSSION

The experimental part of this study aimed to thoroughly evaluate the effectiveness of the proposed Colonial Competitive Grey Wolf Optimizer (CCGWO) algorithm by analyzing its performance across multiple key factors: solution quality, computational efficiency, and robustness. Through a series of experiments, we obtained valuable insights into the capabilities and strengths of CCGWO.

Superior Solution Quality:

The results consistently demonstrated that CCGWO outperforms existing techniques, including recent and previous methods, by achieving the best fuel cost across various test cases. We closely examined the algorithm's unique features to shed light on the reasons behind this superior solution quality. Incorporating socio-political evolution modeling and the enhanced operator for population update are key contributors to CCGWO's ability to explore the search space effectively. By combining these elements, CCGWO is adept at balancing exploration and exploitation, leading to the discovery of high-quality solutions. The socio-political evolution modeling facilitates a more informed exploration process, allowing the algorithm to identify promising regions for further exploitation. Moreover, the enhanced operator enhances the exploitation capabilities, enabling CCGWO to converge efficiently to optimal solutions. These factors collectively contribute to the algorithm's consistently finding superior solutions.

Comparison with Existing Techniques:

The comprehensive comparisons with various existing techniques, including hybrid and developed-based methods, further validate the effectiveness of CCGWO. The superior performance can be attributed to its ability to avoid getting trapped in local optima while efficiently exploring the search space. CCGWO's adaptive and balanced exploration-exploitation strategy allows it to efficiently navigate complex optimization landscapes, leading to improved global exploration. Additionally, its ability to consistently outperform other techniques across diverse problem instances demonstrates the algorithm's robustness.

Advantages over Conventional GWO:

A notable distinction between CCGWO and conventional GWO lies in the smoother convergence characteristics exhibited by CCGWO. This smoother convergence can be attributed to the influence of socio-political evolution modeling. Incorporating socio-political elements guides the search process more effectively, leading to more efficient convergence and superior solution quality. Furthermore, the lower Std. values further affirm CCGWO's robustness and consistent performance across different problem instances.

Hybrid Mechanism:

The hybrid mechanism employed in CCGWO plays a pivotal role in its overall efficacy. By synergistically combining socio-political evolution modeling with the enhanced population update operator, the algorithm achieves a more adaptive navigation of the search space. This balance between intensification and diversification enables CCGWO to effectively explore and exploit the search space, resulting in the identification of global optima even in the presence of a large number of local solutions. Moreover, the simplicity of implementation further enhances the practicality and accessibility of CCGWO for various optimization tasks.

In conclusion, the deep analysis of the experimental results provides valuable insights into the strengths and advantages of the proposed CCGWO algorithm. The algorithm's unique features, including socio-political evolution modeling and the enhanced operator, contribute to its superior solution quality, computational efficiency, and robustness. These findings offer inspiration and open new avenues for applying CCGWO to address complex optimization challenges in real-world applications. The remarkable performance and hybrid approach of CCGWO have the potential to contribute significantly to the field of optimization research, enabling faster convergence and more reliable solutions for diverse optimization problems.

Convergence Behavior Analysis

The convergence behavior of a metaheuristic algorithm reflects the dynamic interplay between two fundamental processes: exploration and exploitation. An algorithm capable of balancing these processes can effectively escape local optima while rapidly converging toward the global optimum. In this section, we analyze the convergence characteristics of the CCGWO in comparison with the standard GWO and the CGWO, based on the convergence curves plotted for the 6-unit (Fig. 1), 15-unit (Type I) (Fig. 2), 110-unit (Type I) (Fig. 3), and 140-unit (Fig. 4) test systems.

Convergence Speed: A key observation from all convergence curves is that the CCGWO exhibits a steeper downward slope during the initial iterations compared to both the GWO and CGWO. In other words, the CCGWO reaches specific cost levels in fewer iterations than its counterparts. For instance, in the 110-unit system (Fig. 3), the CCGWO reaches a cost approximately 1.5 times its final value within the first 250 iterations, whereas the standard GWO remains at a cost level very more its final value at the same stage. This rapid initial acceleration is a direct result of the competitive mechanism, which guides the groups toward promising search regions from the earliest stages of the optimization process.

Convergence Quality (Final Value): Beyond speed, the CCGWO also yields superior final solution quality. In all four test systems, the CCGWO convergence curve stabilizes at a lower cost level. This performance gap is more pronounced in larger systems (110 and 140 units) than in smaller ones (6 and 15 units). This phenomenon is attributed to the fact that as the search space dimension increases, the number of local optima proliferates, often trapping the standard GWO. Conversely, the multi-group structure of the CCGWO preserves population diversity throughout the execution, allowing it to bypass these traps, while the competitive mechanism prevents aimless dispersion of the groups.

Convergence Stability: Fluctuations in the convergence curve—representing cost variations across iterations—serve as a proxy for search stability. The CCGWO curves, particularly in the second half of the execution, exhibit significantly fewer fluctuations compared to the GWO. This is notably evident in the 140-unit system (Fig. 4), where the GWO curve displays relatively large oscillations even in late iterations; in contrast, the CCGWO achieves a smooth and stable trajectory after approximately half the maximum number of iterations. This stability is entirely consistent with the Std. values reported in the results tables, where the CCGWO consistently achieves the lowest values among all compared algorithms.

Mechanism Analysis: The tripartite superiority (speed, quality, and stability) of the CCGWO stems from the synergy between its two primary components. During the early stages, the multi-group structure divides the population into independent sub-groups, enabling parallel exploration of a vast search space. Simultaneously, the competitive mechanism selects the optimal wolves across all groups at the end of each iteration. As the process progresses, the groups gradually focus on promising regions—marking a transition from exploration to exploitation—while the competitive mechanism leverages selection pressure to ensure targeted and rapid convergence. Consequently, the CCGWO neither converges prematurely nor lingers in erratic states; instead, it follows a balanced, swift, and robust convergence path.

Statistical Analysis:

To ensure that the observed superiority of the proposed algorithm is not due to random effects, a statistical analysis was conducted based on the results of 25 independent runs.

As shown in Tables I–VIII, CCGWO not only achieves the lowest cost in all test systems, but also exhibits the smallest standard deviation among all compared algorithms. For instance, in the 6-, 15-, 110-, and 140-unit systems, the Std. of CCGWO are 0.0037, 0.6023, 0.0025, and 0.0847, respectively, whereas the closest competitor algorithms RDPSO, mQOGWO, CFLBO, and AGWO reports a value of 2.2828, 1.09, 0.008, and 0.5482, respectively. This difference indicates a higher level of stability and consistency for CCGWO.

To further examine the statistical significance of these differences, the non-parametric Wilcoxon signed-rank test was applied at a significance level of alpha = 0.05 between the results of CCGWO and the best competing algorithm for each system. The obtained p-values for the 6-, 15-, 110-, and 140-unit systems were 0.00085, 0.0032, 0.0096, and 0.00047, respectively, respectively. Since all values are below 0.05, the null hypothesis (i.e., no significant difference between the algorithms) is rejected, confirming that the superiority of CCGWO is statistically significant.

In addition, the Friedman mean rank was computed for all algorithms. CCGWO obtained the best average rank (1.23), followed by algorithms GWO and others in the subsequent positions. These results further support the robust and consistently superior performance of CCGWO compared with the other methods.

Enhancing Optimization Performance:

CCGWO's dominance is attributed to factors like as population division and colonial rivalry, which lead to faster convergence and escape from local optima. This structure enables a more diverse search process while keeping the algorithm from being stuck in unsatisfactory responses.

Second, the new update technique in CCGWO takes advantage of each wolf's best personal position, including data from all members rather than just the top three (alpha, beta, and delta) as in the traditional GWO. The incorporation of lesser wolves' experiences improves exploration capabilities and helps to conduct a more complete search of the solution area. Third, adaptive weighting of alpha, beta, and delta influences, as determined by experimental testing, ensures that the algorithm can dynamically balance exploration and exploitation in response to the issue landscape. This adaptability is essential for successfully handling a diverse set of optimization difficulties.

Finally, the colonial rivalry mechanism assures that weaker groups are eventually eliminated while stronger groups absorb their members, resulting in a long-term gain in colony power and solution quality. This competitive dynamic increases the algorithm's robustness and effectiveness.

Together, these traits lead to the CCGWO's superior performance in a wide range of optimization circumstances, allowing it to continuously beat competing techniques.

Conclusions

Nowadays, the ELD problems in electrical power networks are very important. On the other hand, the extraordinary complexity of such problems has always directed designers to propose robust and fast-responding algorithms. This paper presents a powerful and suitable algorithm, called Colonial Competitive Grey Wolf Optimizer (CCGWO), for the ELD problem. The proposed algorithm is the developed and improved version of the basic GWO algorithm. The proposed algorithm is implemented on different test systems with different ELD functions, and the results are compared with the most recent algorithms based on four statistical indices. The results confirmed the robustness and effectiveness of the proposed algorithm for complex engineering problems such as ELD. Consequently, the proposed algorithm can be used and developed for various engineering problems in future studies.

Funding information

No funding

Competing interest

The author declares that they have no competing interests.

Data Availability

All data and information used in this study have been obtained from previously published sources, and the corresponding references have been provided throughout the manuscript. No new datasets were generated during this study. All datasets, benchmark cases, and relevant materials used for the experiments are clearly cited and referenced in the manuscript.

Declaration of Generative AI and AI-assisted technologies in the writing process

The authors declare that they did not use AI-assisted technologies to create this article.

Authors Contributions:

Taybeh Salehnia: Formal analysis, Writing , Software implementation, Simulation setup, and Performance evaluation, Conceptualization, Methodology design, and Final manuscript review.
Roqia Rateb: Formal analysis, Experimental design, Data curation, Supervision, Validation, Methodology design, Writing , and Final manuscript review.

Maytham N Meqdad: Formal analysis, Experimental design, Data curation, Supervision, Validation, Methodology design, Writing .

All authors have read and approved the final version of the manuscript.

Appendix A.1. The complete details of the simulation environment, test systems, parameters, and evaluation procedure

To ensure that the results of this study are reliable and reproducible, this section provides the complete details of the simulation environment, test systems, algorithm parameters, and evaluation procedure.

Test Systems:

The evaluation was conducted on four standard power system test cases. These include the 6‑unit system with a load demand of 1263 MW, the 15‑unit system with a load of 2630 MW considering transmission losses and valve‑point effects, the 110‑unit system with a load demand of 15000 MW, and the 140‑unit system representing the Iranian power grid with a load demand of 49,342 MW over 24 time periods. The complete datasets of these systems, including cost coefficients, generation limits, ramp rates, and prohibited operating zones, are provided in the appendix of the manuscript.

Algorithm Parameters:

The parameter settings of CCGWO for each system were described in the previous section. For the competing algorithm the standard GWO, the parameters were set exactly according to the values reported in their respective reference papers. Using consistent parameter settings across all experiments ensures a fair comparison.

Constraint Handling:

The problem constraints (power balance, generation limits, ramp rate limits, and prohibited operating zones) were handled using a penalty function approach. Any infeasible solution is penalized in the objective function according to the magnitude of its constraint violation. The same penalty coefficients were applied for all algorithms.

Number of Runs and Evaluation Metrics:

For each system and each algorithm, 25 independent runs were performed using different initial populations. From these runs, four statistical measures were recorded: the best, mean, worst, and standard deviation of the objective function value.

Hardware and Software Environment:

All simulations were carried out on a computer equipped with an Intel Core i7‑10750H processor, 16 GB of RAM, running Windows 10 and MATLAB R2021a.

Stopping Criterion:

The only stopping condition was reaching the predefined maximum number of iterations ($Iter_{max}$). No early‑stopping condition was applied so that all algorithms had equal opportunity to explore the search space.

Appendix A.2. The Validation of the Algorithm on CEC 2017 Test Functions

To evaluate the performance of the CCGWO algorithm in a standard and unbiased benchmark environment, we implemented it on the well-known CEC 2017 test suite [102]. All tests were conducted in 30 dimensions. This choice was deliberate; as dimensionality increases, problem complexity grows nonlinearly, making the performance gap between robust and weaker algorithms more apparent. To mitigate the impact of randomness and ensure a fair statistical judgment, each function was executed for 30 independent runs. The parameter settings for the competing algorithms are listed in Table A.2.1, and the simulation results for the D=30 case are presented in Table A.2.1 [103], [104], [105], [106].

What do the results indicate? Overall, CCGWO outperformed its competitors in nearly all key functions, ranging from F1 and F3 to F30. A particularly notable aspect is the behavior of CCGWO on multimodal and composition functions. Imagine a search space landscape filled with numerous valleys and peaks, where each valley represents a local optimum. Many algorithms fall into the first valley they encounter and fail to escape. However, through its colonial competitive mechanism—where weaker groups fade away and stronger members are reorganized—CCGWO effectively “scouts” different valleys, significantly increasing the probability of locating the global optimum.

In summary, the results of this benchmark demonstrate that CCGWO is not merely a slightly modified version of GWO; rather, it is a reliable and robust algorithm that maintains its effectiveness even under challenging conditions.

Table A.2.1. Configuring parameters in the algorithms [107]

Algorithm	Parameter settings with NFEs=3.00E+05
Gravitational search algorithm (GSA) [115]	$\alpha = 20$, population = 100, $G_0=100$
Sine cosine algorithm (SCA) [115]	$r_1 = a - Iter \times (a / Itermax)$, $r_2 = 2\pi \times rand()$, $r_3 = 2 \times rand()$, $r_4 = rand()$,a = 2, population= 60
Butterfly optimization algorithm (BOA) [116]	c = 0.01, p = 0.8, $\alpha = 0.1$, population= 60
Bat algorithm (BA) [117]	the pulse rate (r), the parameters updating γ and α are set to 0.9, population = 50
Elephant herding optimization (EHO) [118]	$\beta = 0.1$, clan = 5, $\alpha = 0.5$, population= 60
Harris Hawks optimization (HHO) [119]	C= [0, 2], C= [0, 2], E = [-2,2], population= 60

Table A.2.2. Optimal results of the optimization algorithms on CEC 2017.

Function	GSA	EHO	SCA	BA	HHO	BOA	GWO	CCGWO
	Mean Std.	Mean Std.	Mean Std.	Mean Std.	Mean Std.	Mean Std.	Mean Std.	Mean Std.
F1	1.90E+3 1.03E+3	2.68E+10 5.87E+9	1.25E+10 2.94E+9	3.37E+11 3.48E+9	1.04E+7 1.87E+6	4.47E+10 6.73E+9	1.72E+3 8.91E+8	**5.15E+2** 3.02E+2
F3	8.27E+4 4.33E+3	7.23E+4 9.76E+3	3.47E+4 7.52E+3	2.85E+9 4.62E+8	4.85E+3 1.70E+3	6.48E+4 9.30E+3	2.82E+4 9.70E+3	**2.25E-01** 9.89E-02
F4	1.42E+2 1.59E+1	4.60E+3 3.91E+2	1.05E+3 3.75E+2	3.54E+4 1.36E+3	1.25E+2 2.36E+1	1.94E+4 3.77E+3	1.70E+2 4.98E+1	**7.13E+1** 6.69E+0
F5	2.26E+2 2.01E+1	3.34E+2 1.80E+1	2.73E+2 2.24E+1	5.26E+3 2.13E+1	2.30E+2 2.98E+1	3.71E+2 2.33E+1	9.20E+1 2.63E+1	**8.34E+1** 4.85E+1
F6	5.00E+1 2.75E+0	7.10E+1 6.00E+0	4.90E+1 5.20E+0	6.85E+2 6.55E+0	6.34E+1 5.40E+0	7.00E+1 9.54E+0	4.00E+0 2.33E+0	**1.72E+0** 1.05E+0
F7	8.70E+1 1.19E+1	7.30E+2 4.43E+1	4.40E+2 4.15E+1	1.80E+3 1.63E+2	5.30E+2 8.35E+1	6.10E+2 5.58E+1	1.35E+2 4.95E+1	**5.82E+1** 5.67E+0
F8	1.51E+2 1.31E+1	3.00E+2 2.15E+1	2.50E+2 1.62E+1	1.65E+4 3.98E+1	1.69E+2 2.33E+1	3.00E+2 1.85E+1	8.10E+1 1.25E+1	**4.28E+1** 3.94E+1
F9	2.03E+3 3.92E+2	8.48E+3 1.20E+3	4.81E+3 1.43E+3	1.01E+4 1.69E+3	5.86E+3 5.95E+2	8.45E+3 1.03E+3	2.80E+2 1.39E+2	**2.05E+2** 1.16E+2
F10	3.87E+3 4.34E+2	7.15E+3 3.14E+2	7.04E+3 3.42E+2	9.88E+4 7.65E+3	4.39E+3 6.57E+2	7.49E+3 2.75E+2	2.73E+3 5.49E+2	**7.95E+2** 3.46E+2
F11	3.50E+2 8.92E+1	1.03E+3 1.46E+2	1.05E+3 2.65E+2	3.46E+4 3.37E+3	**1.52E+2** 3.99E+1	4.51E+3 1.83E+3	4.10E+2 4.42E+2	1.78E+2 6.29E+1
F12	1.03E+7 1.93E+7	3.17E+9 5.20E+8	1.13E+9 2.52E+8	3.95E+9 9.45E+7	9.24E+6 5.95E+6	1.05E+10 3.41E+9	3.31E+7 3.81E+7	**5.61E+5** 2.03E+5
F13	**2.97E+4** 6.45E+3	1.06E+9 2.71E+8	3.87E+8 1.24E+8	4.17E+9 6.17E+8	3.14E+5 1.17E+5	6.52E+9 3.65E+9	6.63E+6 2.33E+7	5.17E+4 6.64E+4
F14	4.73E+5 1.31E+5	1.79E+5 7.54E+4	1.42E+5 8.76E+4	1.35E+7 1.65E+6	5.19E+4 1.03E+5	9.60E+5 1.84E+6	7.96E+4 1.76E+5	**2.65E+4** 8.47E+4
F15	1.02E+4 1.93E+3	2.60E+7 1.04E+7	1.19E+7 1.07E+7	1.73E+8 6.11E+7	6.89E+4 6.59E+4	1.35E+8 1.80E+8	2.43E+5 5.82E+5	**9.23E+3** 4.32E+4
F16	1.58E+3 2.84E+2	2.57E+3 1.94E+2	2.01E+3 3.47E+2	4.46E+4 2.96E+2	1.60E+3 3.65E+2	4.58E+3 1.38E+3	**7.20E+2** 2.39E+2	8.42E+2 4.60E+2
F17	1.20E+3 1.70E+2	8.90E+2 1.65E+2	7.10E+2 1.88E+2	3.07E+4 1.82E+2	9.54E+2 2.84E+2	6.17E+3 5.38E+3	2.30E+2 1.16E+2	**1.81E+2** 9.58E+1
F18	3.18E+5 1.76E+5	1.38E+6 6.64E+5	3.92E+6 2.66E+6	2.26E+9 1.98E+7	8.29E+5 7.16E+5	1.08E+7 1.50E+7	7.73E+5 1.40E+6	**2.59E+5** 2.94E+4
F19	**1.23E+4** 5.13E+3	6.23E+7 2.53E+7	2.13E+7 1.10E+7	1.66E+9 6.24E+7	2.41E+5 1.77E+5	2.64E+8 2.69E+8	2.04E+5 3.88E+5	5.26E+4 5.72E+3
F20	1.03E+3 2.36E+2	5.80E+2 7.75E+1	6.30E+2 1.01E+2	3.96E+3 1.36E+2	7.37E+2 2.12E+2	8.40E+2 1.04E+2	3.30E+2 1.66E+2	**2.79E+2** 1.35E+2
F21	4.60E+2 1.95E+1	5.00E+2 1.60E+1	4.50E+2 1.50E+1	2.87E+3 2.55E+1	4.60E+2 5.25E+1	3.90E+2 1.51E+2	2.70E+2 1.85E+1	**6.84E+1** 7.95E+0
F22	4.19E+3 1.69E+3	2.99E+3 3.14E+2	5.98E+3 2.36E+3	5.33E+4 1.70E+3	4.23E+3 1.77E+3	2.20E+3 7.00E+2	2.27E+3 1.45E+3	**1.83E+3** 8.60E+2
F23	1.26E+3 1.23E+2	9.90E+2 5.15E+1	6.80E+2 2.64E+1	3.37E+4 4.95E+2	8.07E+2 1.05E+2	9.00E+2 1.23E+2	4.30E+2 3.04E+1	**4.04E+2** 2.83E+1
F24	8.90E+2 5.57E+1	1.08E+3 6.37E+1	7.60E+2 2.46E+1	5.77E+4 5.19E+5	1.05E+3 1.47E+2	1.37E+3 1.67E+2	5.00E+2 4.70E+1	**4.29E+2** 9.18E+0
F25	4.30E+2 1.22E+1	2.22E+3 5.33E+2	6.70E+2 7.39E+1	4.95E+5 4.86E+7	**4.10E+2** 2.02E+1	3.15E+3 4.64E+2	4.60E+2 2.69E+1	4.35E+2 6.81E+1
F26	4.26E+3 8.95E+2	6.00E+3 5.87E+2	4.33E+3 3.06E+2	7.85E+4 6.76E+2	3.75E+3 2.12E+3	7.80E+3 1.17E+3	1.83E+3 2.45E+2	**9.27E+2** 8.06E+1

F27	1.97E+3 3.21E+2	1.23E+3 9.38E+1	6.90E+2 2.60E+1	3.65E+4 6.75E+1	6.31E+2 6.78E+1	1.06E+3 1.35E+2	**5.30E+2** 1.78E+1	6.25E+2 1.69E+2
F28	5.10E+2 4.94E+1	2.29E+3 3.00E+2	1.03E+3 1.63E+2	5.64E+4 4.33E+2	4.52E+2 2.65E+1	5.05E+3 5.08E+2	5.30E+2 4.83E+1	**4.03E+2** 1.10E+1
F29	1.81E+3 2.10E+2	2.30E+3 2.14E+2	1.69E+3 2.52E+2	5.46E+4 3.57E+2	1.37E+3 2.96E+2	6.47E+3 2.74E+3	8.10E+2 1.26E+2	**7.24E+2** 8.86E+1
F30	1.67E+5 1.24E+5	8.97E+7 3.04E+7	7.06E+7 3.47E+7	1.65E+9 6.31E+7	1.65E+6 8.27E+5	5.74E+8 3.63E+8	3.90E+6 3.10E+6	**5.95E+4** 9.08E+3

REFERENCES

[1] M. Ghasemi, M. Zare, S. K. Mohammadi, and S. Mirjalili, "Applications of whale migration algorithm in optimal power flow problems of power systems," in *Handbook of Whale Optimization Algorithm*, Elsevier, 2024, pp. 347–364.

[2] M. Ghasemi, E. Akbari, I. Faraji Davoudkhani, A. Rahimnejad, M. B. Asadpoor, and S. A. Gadsden, "Application of Coulomb's and Franklin's laws algorithm to solve large-scale optimal reactive power dispatch problems," *Soft Comput.*, vol. 26, no. 24, pp. 13899–13923, 2022.

[3] X. Wu, W. Cao, D. Wang, M. Ding, L. Yu, and Y. Nakanishi, "Demand response model based on improved Pareto optimum considering seasonal electricity prices for Dongfushan Island," *Renew. Energy*, vol. 164, pp. 926–936, 2021.

[4] A. Wan *et al.*, "An integrated TCN-ECA-BiLSTM and TD3 framework for high-accuracy forecasting and optimal load allocation in combined heat and power systems," *Energy*, p. 140670, 2026.

[5] V. P. Sakthivel, M. Suman, and P. D. Sathya, "Physics inspired metaheuristic approach for solving economic load dispatch in large scale power systems," in *Power Engineering POWER COIN'20 proceedings of international Conference at Sona College of Technology, Salem, India*, 2020, pp. 12–13.

[6] A. Srivastava and D. K. Das, "A new aggrandized class topper optimization algorithm to solve economic load dispatch problem in a power system," *IEEE Trans. Cybern.*, 2020.

[7] M. Ghasemi, S. Ghavidel, M. M. Ghanbarian, M. Gharibzadeh, and A. Azizi Vahed, "Multi-objective optimal power flow considering the cost, emission, voltage deviation and power losses using multi-objective modified imperialist competitive algorithm," *Energy*, vol. 78, pp. 276–289, Dec. 2014, doi: 10.1016/j.energy.2014.10.007.

[8] Y. Yang, G. Yan, G. Mu, and Z. Chen, "Hierarchical bidding strategy for heterogeneous P2H loads and wind power: state-driven aggregation and switching time scheduling," *Energy*, p. 137766, 2025.

[9] G. Xu and Z. Yang, "Risk-aware low carbon optimization of hydrogen-oriented multi-energy system with linear approximation of bi-product fuel cell," *Energy*, p. 137581, 2025.

[10] G. Chen, J. Qian, Z. Zhang, and S. Li, "Application of modified pigeon-inspired optimization algorithm and constraint-objective sorting rule on multi-objective optimal power flow problem," *Appl. Soft Comput.*, vol. 92, p. 106321, Jul. 2020, doi: 10.1016/j.asoc.2020.106321.

[11] G. R. Venkatakrishnan, R. Rengaraj, and S. Salivahanan, "Grey Wolf optimizer to real power dispatch with non-linear constraints," *C. - Comput. Model. Eng. Sci.*, vol. 115, no. 1, pp. 25–45, 2018, doi: 10.3970/cmes.2018.115.025.

[12] R. P. Parouha and P. Verma, "An innovative hybrid algorithm to solve nonconvex economic load dispatch problem with or without valve point effects," *Int. Trans. Electr. Energy Syst.*, vol. 31, no. 1, p. e12682, doi: 10.1002/2050-7038.12682.

[13] M. Mohammadian, A. Lorestani, and M. M. Ardehali, "Optimization of single and multi-areas economic dispatch problems based on evolutionary particle swarm optimization algorithm," *Energy*, vol. 161, pp. 710–724, 2018.

[14] P. Wang, Y. Wang, and J. Jin, "A DBSCAN-enhanced niching differential evolution algorithm for solving nonlinear equations," *Inf. Sci. (Ny).*, p. 122996, 2025.

[15] Y. Tian, Z. Zhu, X. Zhao, X. Chen, W. Huang, and X. Zhang, "A Dynamic and Heterogeneous Representation for Topology Optimization Using Evolutionary Algorithms [Research Frontier]," *IEEE Comput. Intell. Mag.*, vol. 20, no. 4, pp. 71–82, 2025.

[16] Y. Xu, X. Wang, and Z. Wang, "Quantum mapping algorithm for structural non-probabilistic reliability optimization," *Struct. Multidiscip. Optim.*, vol. 68, no. 7, p. 138, 2025.

[17] T. Ji *et al.*, "Mode-Constrained Two-Stage Current Injection Optimization for Voltage Unbalance Mitigation in IBR-Rich Distribution Network," *IEEE J. Emerg. Sel. Top. Power Electron.*, 2025.

[18] H. Jia *et al.*, "A Novel Fault Location Method for Unbalanced Active Distribution Networks With Sparse Hybrid Measurements," *IEEE Trans. Smart Grid*, 2025.

[19] Z. Li *et al.*, "Receding horizon D-optimal input design for identification of wireless power transfer systems," *IEEE J. Emerg. Sel. Top. Power Electron.*, vol. 11, no. 3, pp. 3597–3606, 2023.

[20] L. C. Kien, T. T. Nguyen, C. T. Hien, and M. Q. Duong, "A Novel Social Spider Optimization Algorithm for Large-Scale Economic Load Dispatch Problem," *Energies*, vol. 12, no. 6, p. 1075, 2019, doi: 10.3390/en12061075.

[21] M. F. Tabassum *et al.*, "Evolutionary simplex adaptive Hooke-Jeeves algorithm for economic load dispatch problem considering valve point loading effects," *Ain Shams Eng. J.*, vol. 12, no. 1, pp. 1001–1015, doi: 10.1016/j.asej.2020.04.006.

[22] H. Lu, P. Sriyanyong, Y. H. Song, and T. Dillon, "Experimental study of a new hybrid PSO with mutation for economic dispatch with non-smooth cost function," *Int. J. Electr. Power Energy Syst.*, vol. 32, no. 9, pp. 921–935, 2010, doi: 10.1016/j.ijepes.2010.03.001.

[23] D. N. Vo, P. Schegner, and W. Ongsakul, "Cuckoo search algorithm for non-convex economic dispatch," *IET Gener. Transm. Distrib.*, vol. 7, no. 6, pp. 645–654, 2013, doi: 10.1049/iet-gtd.2012.0142.

[24] A. Goudarzi, Y. Li, and J. Xiang, "A hybrid non-linear time-varying double-weighted particle swarm optimization for solving non-convex combined environmental economic dispatch problem," *Appl. Soft Comput.*, vol. 86, p. 105894, 2020.

[25] A. I. Selvakumar and K. Thanushkodi, "Optimization using civilized swarm: Solution to economic dispatch with multiple minima," *Electr. Power Syst. Res.*, vol. 79, no. 1, pp. 8–16, 2009, doi: 10.1016/j.epsr.2008.05.001.

[26] A. Bhattacharya and P. K. Chattopadhyay, "Biogeography-Based Optimization for Different Economic Load Dispatch Problems," *IEEE Trans. Power Syst.*, vol. 25, no. 2, pp. 1064–1077, 2010, doi: 10.1109/tpwrs.2009.2034525.

[27] M. R. Lohokare, B. K. Panigrahi, S. S. Pattnaik, S. Devi, and A. Mohapatra, "Neighborhood Search-Driven Accelerated Biogeography-Based Optimization for Optimal Load Dispatch," *IEEE Trans. Syst. Man, Cybern. Part C (Applications Rev.*, vol. 42, no. 5, pp. 641–652, Sep. 2012, doi: 10.1109/TSMCC.2012.2190401.

[28] K. T. Chaturvedi, M. Pandit, and L. Srivastava, "Self-Organizing Hierarchical Particle Swarm Optimization for Nonconvex Economic Dispatch," *IEEE Trans. Power Syst.*, vol. 23, no. 3, pp. 1079–1087, Aug. 2008, doi: 10.1109/TPWRS.2008.926455.

[29] M. Ghasemi, "A chaotic modified algorithm for economic dispatch problems with generator constraints," *Amer. J. Electr. Comput. Eng.*, vol. 1,

no. 2, pp. 61–71, 2017.
[30] B. K. Panigrahi and V. Ravikumar Pandi, “Bacterial foraging optimisation: Nelder–Mead hybrid algorithm for economic load dispatch,” *IET Gener. Transm. Distrib.*, vol. 2, no. 4, p. 556, 2008, doi: 10.1049/iet-gtd:20070422.
[31] Y. Labbi and D. Ben Attous, “A Hybrid Big Bang–Big Crunch optimization algorithm for solving the different economic load dispatch problems,” *Int. J. Syst. Assur. Eng. Manag.*, vol. 8, no. 2, pp. 275–286, 2016, doi: 10.1007/s13198-016-0432-4.
[32] K. Kapelinski, J. P. J. Neto, and E. M. dos Santos, “Firefly Algorithm with non-homogeneous population: A case study in economic load dispatch problem,” *J. Oper. Res. Soc.*, pp. 1–16, 2020, doi: 10.1080/01605682.2019.1700184.
[33] X. He, Y. Rao, and J. Huang, “A novel algorithm for economic load dispatch of power systems,” *Neurocomputing*, vol. 171, pp. 1454–1461, 2016, doi: 10.1016/j.neucom.2015.07.107.
[34] N. Amjady and H. Nasiri-Rad, “Economic dispatch using an efficient real-coded genetic algorithm,” *IET Gener. Transm. Distrib.*, vol. 3, no. 3, pp. 266–278, 2009, doi: 10.1049/iet-gtd:20080469.
[35] G. Xiong and D. Shi, “Orthogonal learning competitive swarm optimizer for economic dispatch problems,” *Appl. Soft Comput.*, vol. 66, pp. 134–148, 2018, doi: 10.1016/j.asoc.2018.02.019.
[36] V. K. Jadoun, N. Gupta, K. R. Niazi, and A. Swarnkar, “Modulated particle swarm optimization for economic emission dispatch,” *Int. J. Electr. Power Energy Syst.*, vol. 73, no. 1, pp. 80–88, Dec. 2015, doi: 10.1016/j.ijepes.2015.04.004.
[37] X. Li, A. Li, and Z. Lu, “A granular computing method for economic dispatch problems with valve-point effects,” *IEEE Access*, vol. 7, pp. 78260–78273, 2019.
[38] W. T. Elsayed and E. F. El-Saadany, “A Fully Decentralized Approach for Solving the Economic Dispatch Problem,” *IEEE Trans. Power Syst.*, vol. 30, no. 4, pp. 2179–2189, 2015, doi: 10.1109/tpwrs.2014.2360369.
[39] F. Mohammadi and H. Abdi, “A modified crow search algorithm (MCSA) for solving economic load dispatch problem,” *Appl. Soft Comput.*, vol. 71, pp. 51–65, 2018.
[40] X. Li, H. Zhang, and Z. Lu, “A differential evolution algorithm based on multi-population for economic dispatch problems with valve-point effects,” *IEEE Access*, vol. 7, pp. 95585–95609, 2019.
[41] C.-L. Chiang, “Improved Genetic Algorithm for Power Economic Dispatch of Units With Valve-Point Effects and Multiple Fuels,” *IEEE Trans. Power Syst.*, vol. 20, no. 4, pp. 1690–1699, 2005, doi: 10.1109/tpwrs.2005.857924.
[42] J. Yu, C.-H. Kim, and S.-B. Rhee, “Clustering cuckoo search optimization for economic load dispatch problem,” *Neural Comput. Appl.*, vol. 32, no. 22, pp. 16951–16969, 2020, doi: 10.1007/s00521-020-05036-w.
[43] A. Kavousi-Fard and A. Khosravi, “An intelligent θ-Modified Bat Algorithm to solve the non-convex economic dispatch problem considering practical constraints,” *Int. J. Electr. Power Energy Syst.*, vol. 82, pp. 189–196, 2016, doi: 10.1016/j.ijepes.2016.03.017.
[44] J. Chen and H. Imani Marrani, “An Efficient New Hybrid ICA-PSO Approach for Solving Large Scale Non-convex Multi Area Economic Dispatch Problems,” *J. Electr. Eng. Technol.*, vol. 15, no. 3, pp. 1127–1145, 2020, doi: 10.1007/s42835-020-00416-7.
[45] G. Xiong, D. Shi, and X. Duan, “Multi-strategy ensemble biogeography-based optimization for economic dispatch problems,” *Appl. Energy*, vol. 111, pp. 801–811, 2013, doi: 10.1016/j.apenergy.2013.04.095.
[46] A. Hazra, S. Das, A. Laddha, and M. Basu, “Economic Power Generation Strategy for Wind Integrated Large Power Network Using Heat Transfer Search Algorithm,” *J. Inst. Eng. Ser. B*, pp. 1–7, 2020.
[47] M. Hamdi, L. Idomghar, M. Chaoui, and A. Kachouri, “An improved adaptive differential evolution optimizer for non-convex Economic Dispatch Problems,” *Appl. Soft Comput.*, vol. 85, p. 105868, 2019, doi: 10.1016/j.asoc.2019.105868.
[48] N. Ghorbani and E. Babaei, “Exchange market algorithm for economic load dispatch,” *Int. J. Electr. Power Energy Syst.*, vol. 75, pp. 19–27, 2016, doi: 10.1016/j.ijepes.2015.08.013.
[49] C. Chen, D. Zou, and C. Li, “Improved Jaya Algorithm for Economic Dispatch Considering Valve-Point Effect and Multi-Fuel Options,” *IEEE Access*, vol. 8, pp. 84981–84995, 2020, doi: 10.1109/access.2020.2992616.
[50] J. Yu, C.-H. Kim, A. Wadood, T. Khurshaid, and S.-B. Rhee, “Jaya Algorithm With Self-Adaptive Multi-Population and Lévy Flights for Solving Economic Load Dispatch Problems,” *IEEE Access*, vol. 7, pp. 21372–21384, 2019, doi: 10.1109/access.2019.2899043.
[51] K. Vaisakh and A. S. Reddy, “MSFLA/GHS/SFLA-GHS/SDE algorithms for economic dispatch problem considering multiple fuels and valve point loadings,” *Appl. Soft Comput.*, vol. 13, no. 11, pp. 4281–4291, 2013, doi: 10.1016/j.asoc.2013.07.001.
[52] E. Naderi, A. Azizivahed, H. Narimani, M. Fathi, and M. R. Narimani, “A comprehensive study of practical economic dispatch problems by a new hybrid evolutionary algorithm,” *Appl. Soft Comput.*, vol. 61, pp. 1186–1206, 2017.
[53] A. Rajagopalan, V. Sengoden, and R. Govindasamy, “Solving economic load dispatch problems using chaotic self-adaptive differential harmony search algorithm,” *Int. Trans. Electr. Energy Syst.*, vol. 25, no. 5, pp. 845–858, 2014, doi: 10.1002/etep.1877.
[54] N. Karimi and K. Khandani, “Social optimization algorithm with application to economic dispatch problem,” *Int. Trans. Electr. Energy Syst.*, vol. 30, no. 11, 2020, doi: 10.1002/2050-7038.12593.
[55] T. Niknam, H. D. Mojarrad, H. Z. Meymand, and B. B. Firouzi, “A new honey bee mating optimization algorithm for non-smooth economic dispatch,” *Energy*, vol. 36, no. 2, pp. 896–908, 2011, doi: 10.1016/j.energy.2010.12.021.
[56] K. Bhattacharjee, A. Bhattacharya, and S. H. nee Dey, “Oppositional real coded chemical reaction optimization for different economic dispatch problems,” *Int. J. Electr. Power \& Energy Syst.*, vol. 55, pp. 378–391, 2014.
[57] L. Li, Y. Yang, M.-L. Tseng, C.-H. Wang, and M. K. Lim, “A novel method to solve sustainable economic power loading dispatch problem,” *Ind. Manag. Data Syst.*, vol. 118, no. 4, pp. 806–827, 2018, doi: 10.1108/imds-04-2017-0145.
[58] A. Beirami, V. Vahidinasab, M. Shafie-khah, and J. P. S. Catalão, “Multiobjective ray optimization algorithm as a solution strategy for solving non-convex problems: A power generation scheduling case study,” *Int. J. Electr. Power Energy Syst.*, vol. 119, p. 105967, 2020, doi: 10.1016/j.ijepes.2020.105967.
[59] T. Jayabarathi, T. Raghunathan, B. R. Adarsh, and P. N. Suganthan, “Economic dispatch using hybrid grey wolf optimizer,” *Energy*, vol. 111, pp. 630–641, 2016.
[60] M. Kumar and J. S. Dhillon, “A conglomerated ion-motion and crisscross search optimizer for electric power load dispatch,” *Appl. Soft Comput.*, vol. 83, p. 105641, 2019, doi: 10.1016/j.asoc.2019.105641.
[61] M. Ghasemi, S. Ghavidel, J. Aghaei, E. Akbari, and L. Li, “CFA optimizer: A new and powerful algorithm inspired by Franklin’s and Coulomb’s laws theory for solving the economic load dispatch problems,” *Int. Trans. Electr. Energy Syst.*, vol. 28, no. 5, p. e2536, May 2018, doi: 10.1002/etep.2536.
[62] J. Zhao, S. Liu, M. Zhou, X. Guo, and L. Qi, “Modified cuckoo search algorithm to solve economic power dispatch optimization problems,” *IEEE/CAA J. Autom. Sin.*, vol. 5, no. 4, pp. 794–806, 2018.
[63] D. Singh and J. S. Dhillon, “Ameliorated grey wolf optimization for economic load dispatch problem,” *Energy*, vol. 169, pp. 398–419, 2019.
[64] J. X. V. Neto, G. Reynoso-Meza, T. H. Ruppel, V. C. Mariani, and L. dos S. Coelho, “Solving non-smooth economic dispatch by a new combination of continuous GRASP algorithm and differential evolution,” *Int. J. Electr. Power Energy Syst.*, vol. 84, pp. 13–24, 2017, doi:

10.1016/j.ijepes.2016.04.012.
[65] B. Vedik, P. Naveen, and C. K. Shiva, "A novel disruption based symbiotic organisms search to solve economic dispatch," *Evol. Intell.*, 2020, doi: 10.1007/s12065-020-00506-5.
[66] J. Sun, V. Palade, X.-J. Wu, W. Fang, and Z. Wang, "Solving the power economic dispatch problem with generator constraints by random drift particle swarm optimization," *IEEE Trans. Ind. Informatics*, vol. 10, no. 1, pp. 222–232, 2013.
[67] G. Binetti, A. Davoudi, D. Naso, B. Turchiano, and F. L. Lewis, "A Distributed Auction-Based Algorithm for the Nonconvex Economic Dispatch Problem," *IEEE Trans. Ind. Informatics*, vol. 10, no. 2, pp. 1124–1132, 2014, doi: 10.1109/tii.2013.2287807.
[68] T. T. Nguyen and D. N. Vo, "The application of one rank cuckoo search algorithm for solving economic load dispatch problems," *Appl. Soft Comput.*, vol. 37, pp. 763–773, 2015, doi: 10.1016/j.asoc.2015.09.010.
[69] T. P. Van, V. Snasel, and T. T. Nguyen, "Antlion optimization algorithm for optimal non-smooth economic load dispatch," *Int. J. Electr. Comput. Eng.*, vol. 10, no. 2, p. 1187, 2020, doi: 10.11591/ijece.v10i2.pp1187-1199.
[70] R. Azizipanah-Abarghooee, T. Niknam, M. Gharibzadeh, and F. Golestaneh, "Robust, fast and optimal solution of practical economic dispatch by a new enhanced gradient-based simplified swarm optimisation algorithm," *IET Gener. Transm. Distrib.*, vol. 7, no. 6, pp. 620–635, 2013, doi: 10.1049/iet-gtd.2012.0616.
[71] Q. Qin, S. Cheng, X. Chu, X. Lei, and Y. Shi, "Solving non-convex/non-smooth economic load dispatch problems via an enhanced particle swarm optimization," *Appl. Soft Comput.*, vol. 59, pp. 229–242, 2017, doi: 10.1016/j.asoc.2017.05.034.
[72] M. Moradi-Dalvand, B. Mohammadi-Ivatloo, A. Najafi, and A. Rabiee, "Continuous quick group search optimizer for solving non-convex economic dispatch problems," *Electr. Power Syst. Res.*, vol. 93, pp. 93–105, 2012, doi: 10.1016/j.epsr.2012.07.009.
[73] M. A. Elhameed and A. A. El-Fergany, "Water cycle algorithm-based economic dispatcher for sequential and simultaneous objectives including practical constraints," *Appl. Soft Comput.*, vol. 58, pp. 145–154, 2017.
[74] H. Shayeghi and A. Ghasemi, "A modified artificial bee colony based on chaos theory for solving non-convex emission/economic dispatch," *Energy Convers. Manag.*, vol. 79, pp. 344–354, 2014, doi: 10.1016/j.enconman.2013.12.028.
[75] T. Salehnia et al. (2021). Multilevel image thresholding using GOA, WOA and MFO for image segmentation. in Proceedings of the 8th International Conference on New Strategies in Engineering. Information Science and Technology in the Next Century.
[76] S. Raziani et al.,"Selecting of the best features for the knn classification method by Harris Hawk algorithm," in Conference: 8th International Conference on New Solutions in Engineering, Information Science and Technology of the Century ahead, Dubai: the 8th International Conference on New Strategies in Engineering, Information Science and Technology in the Next Century, 2021.
[77] T. Salehnia et al., "Chapter 45 - A MTIS method using a combined of whale and moth-flame optimization algorithms," Handbook of Whale Optimization Algorithm, Variants, Hybrids, Improvements, and Applications, PP: 625-651, 2024.
[78] B. Mandal, P. K. Roy, and S. Mandal, "Economic load dispatch using krill herd algorithm," *Int. J. Electr. power \& energy Syst.*, vol. 57, pp. 1–10, 2014.
[79] Wang, P., Wang, Y., & Jin, J. (2026). A DBSCAN-enhanced niching differential evolution algorithm for solving nonlinear equations. Information Sciences, 734, 122996. doi: https://doi.org/10.1016/j.ins.2025.122996
[80] T. Salehnia, M. Ahmadi, L. Abualigah et al. , "MEO: Martial Eagle Optimizer Algorithm," PREPRINT (Version 1), Research Square, 2024, [https://doi.org/10.21203/rs.3.rs-3908195/v1]
[81] F. MiarNaeimi, T. salehnia, N. Safaeian Hamzehkolaei et al., "PMW: The Portuguese Man o' War optimization algorithm," PREPRINT (Version 1), Research Square, 2024, [https://doi.org/10.21203/rs.3.rs-3930990/v1]
[82] M. Ghasemi, P. Trojovsk\`y, E. Trojovská, and M. Zare, "Gaussian bare-bones Levy circulatory system-based optimization for power flow in the presence of renewable units," *Eng. Sci. Technol. an Int. J.*, vol. 47, p. 101551, 2023.
[83] Bai Bing, Chen Jing, Bai Fan, Nie Qingke, Jia Xiangxin. Corrosion effect of acid/alkali on cementitious red mud-fly ash materials containing heavy metal residues. Environmental Technology & Innovation, 2024, 33: 103485
[84] Bai B, Wu H, Nie Q, Liu J, Jia X. Granular thermodynamic migration model suitable for high-alkalinity red mud filtrates and test verification. International Journal for Numerical and Analytical Methods in Geomechanics, 2025, 49: 1530-1543
[85] Bai B, Wu H, Zhou R, Wu N, Zhang B. A granular thermodynamic framework-based coupled multiphase-substance flow model considering temperature driving effect. Journal of Rock Mechanics and Geotechnical Engineering, 2025, 17: 5816-5828
[86] Bai Bing, Zhang Bixia, Ji Yanjie, Zong Yongchen. A thermodynamic multi-field model for unsaturated sulfate-saline soils considering crystallization process. Computers and Geotechnics, 2025, 184: 107251
[87] K. Bhattacharjee and N. Patel, "A comparative study of Economic Load Dispatch with complex non-linear constraints using Salp Swarm Algorithm," *Sci. Iran.*, 2020.
[88] Bai Bing, Xu Tao, Nie Qingke, Li Pengpeng. Temperature-driven migration of heavy metal Pb2+ along with moisture movement in unsaturated soils. International Journal of Heat and Mass Transfer, 2020,153: 119573
[89] Bai Bing, Bai Fan, Nie Qingke, Jia Xiangxin. A high-strength red mud–fly ash geopolymer and the implications of curing temperature. Powder Technology, 2023, 416: 118242
[90] Bai Bing, Bai Fan, Li Xianke, Nie Qingke, Jia Xiangxin. The remediation efficiency of heavy metal pollutants in water by industrial red mud particle waste. Environmental Technology & Innovation, 2022, 28: 102944
[91] M. Ghasemi, I. F. Davoudkhani, E. Akbari, A. Rahimnejad, S. Ghavidel, and L. Li, "A novel and effective optimization algorithm for global optimization and its engineering applications: Turbulent Flow of Water-based Optimization (TFWO)," *Eng. Appl. Artif. Intell.*, vol. 92, p. 103666, 2020.
[92] B. R. Adarsh, T. Raghunathan, T. Jayabarathi, and X.-S. Yang, "Economic dispatch using chaotic bat algorithm," *Energy*, vol. 96, pp. 666–675, 2016.
[93] J. C. S. Chavez, A. Zamora-Mendez, M. R. A. Paternina, J. F. Y. Heredia, and R. Cardenas-Javier, "A hybrid optimization framework for the non-convex economic dispatch problem via meta-heuristic algorithms," *Electr. Power Syst. Res.*, vol. 177, p. 105999, 2019.
[94] D. Aydın and S. Özyön, "Solution to non-convex economic dispatch problem with valve point effects by incremental artificial bee colony with local search," *Appl. Soft Comput.*, vol. 13, no. 5, pp. 2456–2466, May 2013, doi: 10.1016/j.asoc.2012.12.002.
[95] M. Basu, "Artificial bee colony optimization for multi-area economic dispatch," *Int. J. Electr. Power \& Energy Syst.*, vol. 49, pp. 181–187, 2013.
[96] M. Kheshti, X. Kang, J. Li, P. Regulski, and V. Terzija, "Lightning flash algorithm for solving non-convex combined emission economic dispatch with generator constraints," *IET Gener. Transm. \& Distrib.*, vol. 12, no. 1, pp. 104–116, 2018.
[97] H. M. Dubey, B. K. Panigrahi, and M. Pandit, "Bio-inspired optimisation for economic load dispatch: a review," *Int. J. Bio-Inspired Comput.*, vol. 6, no. 1, pp. 7–21, 2014.
[98] M. Ghasemi, M. Taghizadeh, S. Ghavidel, and A. Abbasian, "Colonial competitive differential evolution: An experimental study for optimal economic load dispatch," *Appl. Soft Comput.*, vol. 40, pp. 342–363, 2016, doi: 10.1016/j.asoc.2015.11.033.
[99] M. Lin, Q. Li, F. Wang, and D. Chen, "An Improved Beetle Antennae Search Algorithm and Its Application on Economic Load Distribution of

Power System," *IEEE Access*, vol. 8, pp. 99624–99632, 2020, doi: 10.1109/access.2020.2997687.
[100] M. Ghasemi, J. Aghaei, E. Akbari, S. Ghavidel, and L. Li, "A differential evolution particle swarm optimizer for various types of multi-area economic dispatch problems," *Energy*, vol. 107, no. 1, pp. 182–195, Jul. 2016, doi: 10.1016/j.energy.2016.04.002.
[101] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, 2014.
[102] M. Farhan Tabassum, M. Saeed, N. Ahmad Chaudhry, J. Ali, M. Farman, and S. Akram, "Evolutionary simplex adaptive Hooke-Jeeves algorithm for economic load dispatch problem considering valve point loading effects," *Ain Shams Eng. J.*, 2020, doi: 10.1016/j.asej.2020.04.006.
[103] Y. Labbi and D. Ben Attous, "A Hybrid Big Bang--Big Crunch optimization algorithm for solving the different economic load dispatch problems," *Int. J. Syst. Assur. Eng. Manag.*, vol. 8, no. 2, pp. 275–286, 2017.
[104] C. Fu, S. Zhang, and K.-H. Chao, "Energy management of a power system for economic load dispatch using the artificial intelligent algorithm," *Electronics*, vol. 9, no. 1, p. 108, 2020.
[105] M. Gholamghasemi, E. Akbari, M. B. Asadpoor, and M. Ghasemi, "A new solution to the non-convex economic load dispatch problems using phasor particle swarm optimization," *Appl. Soft Comput. J.*, vol. 79, 2019, doi: 10.1016/j.asoc.2019.03.038.
[106] M. Kumar and J. S. Dhillon, "A conglomerated ion-motion and crisscross search optimizer for electric power load dispatch," *Appl. Soft Comput.*, vol. 83, p. 105641, 2019, doi: 10.1016/j.asoc.2019.105641.
[107] S. M. Dubey, H. M. Dubey, and S. R. Salkuti, "Modified quasi-opposition-based grey wolf optimization for mathematical and electrical benchmark problems," *Energies*, vol. 15, no. 15, p. 5704, 2022.
[108] K. Shah, J. Soni, and K. Bhattacharjee, "Artificial Electric Field Algorithm Applied to the Economic Load Dispatch Problem With Valve Point Loading Effect: AEFA Applied to ELD With VPLE," *Int. J. Swarm Intell. Res.*, vol. 14, no. 1, pp. 1–23, 2023.
[109] A. K. Barisal and R. C. Prusty, "Large scale economic dispatch of power systems using oppositional invasive weed optimization," *Appl. Soft Comput.*, vol. 29, pp. 122–137, Apr. 2015, doi: 10.1016/J.ASOC.2014.12.014.
[110] G. S. S. Babu, D. B. Das, and C. Patvardhan, "Simulated annealing variants for the solution of economic load dispatch," *JOURNAL-INSTITUTION Eng. INDIA PART EL Electr. Eng. Div.*, pp. 222–229, 2002.
[111] K. K. Vishwakarma and H. M. Dubey, "Simulated annealing based optimization for solving large scale economic load dispatch problems," *Int. J. Eng. Res. \& Technol.*, vol. 1, no. 3, pp. 1–8, 2012.
[112] Z. Zheng, L. Jun-qing, and S. Hong-yan, "A hybrid invasive weed optimization algorithm for the economic load dispatch problem in power systems," *Math. Biosci. Eng.*, vol. 16, no. 4, p. 2775, 2019.
[113] J.-B. Park, Y.-W. Jeong, J.-R. Shin, and K. Y. Lee, "An improved particle swarm optimization for nonconvex economic dispatch problems," *IEEE Trans. power Syst.*, vol. 25, no. 1, pp. 156–166, 2009.
[114] H. M. Dubey, M. Pandit, and B. K. Panigrahi, "A biologically inspired modified flower pollination algorithm for solving economic dispatch problems in modern power systems," *Cognit. Comput.*, vol. 7, pp. 594–608, 2015.
[115] S. Mirjalili, "SCA: a sine cosine algorithm for solving optimization problems," *Knowledge-based Syst.*, vol. 96, pp. 120–133, 2016.
[116] S. Arora and S. Singh, "Butterfly optimization algorithm: a novel approach for global optimization," *Soft Comput.*, vol. 23, no. 3, pp. 715–734, 2019.
[117] X.-S. Yang and A. H. Gandomi, "Bat algorithm: a novel approach for global engineering optimization," *Eng. Comput.*, 2012.
[118] G.-G. Wang, S. Deb, and L. dos S. Coelho, "Elephant herding optimization," in *2015 3rd international symposium on computational and business intelligence (ISCBI)*, 2015, pp. 1–5.
[119] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, "Harris hawks optimization: Algorithm and applications," *Futur. Gener. Comput. Syst.*, vol. 97, pp. 849–872, 2019.