energies

MDPI

Article

Modified Cuckoo Search Algorithm: A Novel Method to Minimize the Fuel Cost

Thang Trung Nguyen [1], Dieu Ngoc Vo [2], Nguyen Vu Quynh [3] and Le Van Dai [4,5,*]

1 Power System Optimization Research Group, Faculty of Electrical and Electronics Engineering, Ton Duc Thang University, Ho Chi Minh City 700000, Vietnam; nguyentrungthang@tdt.edu.vn
2 Department of Power Systems, Ho Chi Minh City University of Technology, Ho Chi Minh City 700000, Vietnam; vndieu@gmail.com
3 Department of Electrical Engineering, Lac Hong University, Bien Hoa 810000, Vietnam; vuquynh@lhu.edu.vn
4 Institute of Research and Development, Duy Tan University, Danang 550000, Vietnam
5 Office of Science Research and Development, Lac Hong University, Bien Hoa 810000, Vietnam
* Correspondence: levandai@duytan.edu.vn; Tel.: +84-901-672-689

Received: 11 April 2018; Accepted: 18 May 2018; Published: 23 May 2018

check for updates

Abstract: Economic load dispatch (ELD) is an important optimization problem for operating and controlling modern power systems, and if ELD is effectively executed, power systems work stably and economically. The main objective of this paper is to develop a novel method to solve the ELD with the purpose of minimizing the total fuel cost of all available generating units while requirements are to satisfy all constraints regarding thermal units, generators, and transmission power networks. The proposed high performance cuckoo search algorithm (HPCSA) is developed from the efficient technique for the second new solution generation of conventional cuckoo search algorithm (CCSA), called adaptive mutation technique. This proposed technique diversifies the local search ability based on a new comparison criterion. The HPCSA is verified on difference systems under special conditions, namely the 10-unit system with multi fuels, 15-unit system considering prohibited operating zones, and three IEEE systems with 30, 57, and 118 buses considering transmission power network constraints. The specific evaluation of the HPCSA is compared to that of Lagrange optimization-based methods (LMS), neural network-based methods (NNMS), CCSA, and other popular methods such as Particle swarm optimization (PSO) variants, Differential evolution (DE) variants, Genetic Algorithm (GA) variants, and state-of-the-art methods. In comparison with CCSA, the proposed method is always more effective and more robust since the proposed method can find most solutions with better quality and faster convergence speed. In comparison with LMS and NNMS, the proposed method can also find solutions with approximate or equal quality. In comparison with popular methods and state-of-the-art methods, the proposed method has more potential since it can reach faster convergence to valid solutions with approximate or better quality. Consequently, it can be concluded that the proposed HPCSA is an effective optimization tool for dealing with ELD problems.

Keywords: cuckoo search algorithm; valve point loading effects; prohibited operating zone; transmission network constraints; IEEE networks

1. Introduction

Over the past decades, a high number of researchers have been devoted to solving optimization problems in engineering by applying conventional optimization algorithms or proposing improved algorithms. Even though there are some wider application areas where these works are applied, this study narrows down to the economic load dispatch (ELD) problem, which is to minimize the total electricity generation fuel cost of all thermal generating units and to satisfy all constraints of the units and other constraints related to transmission power networks [1,2]. For the considered problems, we consider five systems, in which the first system, namely the 10-unit system, considers multiple types of fuel; the second system, the 15-unit system, considers single fuel, prohibited zones, and spinning reserve for power systems; and the three remaining systems, IEEE 30, 57, and 118 buses, power networks and consider single fuel and all constraints.

So far, a large number of methods have been successfully applied for dealing with the five cases of the problem in which methods applied to the case of multi-fuel options are conventional Hopfield neural network (HNN) [3], hierarchical approach (HA) [4], adaptive Hopfield neural network (AHNN) [5], improved Lagrangian neural network (ILNN) [6], hybrid real-coded genetic algorithm (HRCGA) [7], differential evolution (DE) [8], modified evolutionary programming (MEP) [9], artificial immune system algorithm (AIS) [10] and hybrid differential evolution and dynamic programming (HDEDP) [11], and cuckoo search algorithm (CSA) [12]. Among these methods, ones based on neural network and numerical method have the same disadvantages, such as the hard task of tuning control parameters and stopping application for systems with non-differentiable functions. On the contrary, the remaining methods, DE, HRCGA, AISA, and CSA, can overcome such drawbacks, but they cope with other restrictions much depending on randomization and taking much time for tuning control parameters. Among methods belonging to neural networks, IALHN can be considered the most powerful method while CSA can be the most promising meta-heuristic method in the second group.

For the second system, with consideration of prohibited operating zones (POZ) constraints, several methods as CSA [12], the combination of decomposition method and Lagrange relaxation (DLR) [13], lambda iteration method (LIM) [14], particle swarm optimization (PSO) [15], improved quantum-inspired evolutionary algorithm (IQIEA) [16], and improved augmented Lagrange-Hopfield network (IALHN) [17] have been successfully applied with promising results, but most of these methods have not been evaluated in terms of convergence speed because iterations and execution time have not been reported. It is clear that the systems considering POZ constraints have attracted both conventional algorithms and recent meta-heuristic algorithms. Among the mentioned methods, DLR and LIM are the first two methods applied for handling such complicated constraints, and they have obtained optimal solutions with higher objective function than most other remaining methods excluding the comparison of LIM with PSO.

For IEEE 30, 57, and 118 buses power networks, complicated constraints of transmission power networks such as power and voltage limitations of generators, limitations of transformer tap, limitations of capacitor banks, capacity of transmission lines, and active and reactive power balance are taken into consideration. On the other hand, for the cases of considering the constraints associated with transmission power networks, the ELD problem can be also called optimal power flow (OPF) problem. For the complicated OPF problem, the three most popular IEEE systems with 30, 57 and 118 buses have been employed to test performance of optimization methods in terms of the ability to handle all constraints, quality of solutions, and processing speed. Most methods are the family of meta-heuristic algorithms in which conventional methods, modified methods, combination of two different methods, and hybrid methods have been developed widely. In fact, there have been a huge number of applied methods such as the integration of improved genetic algorithm and effective decoupled quadratic load flow (IGA-EDQLF) [18], hybrid IGA with incremental power flow model (HIGA) [19], HIGA with boundary method (HIGA-BM) [20], differential evolution [21,22], conventional PSO [23], Evolving ant direction particle swarm optimization (EADPSO) [24], PSO with Pseudo-Gradient and constriction factor (PG-CF-PSO) [25], Biogeography-based optimization algorithm (BBOAA) [26] and adaptive real-coded biogeography-based optimization algorithm (ARCBBOA) [27], teaching–learning-based optimization algorithm (TLBO) [28], improved TLBO (ITLBO) [29], gravitational search algorithm (GSA) [30], Artificial bee colony algorithm (ABCA) [31], Grey wolf optimizer (GWO) [32], modified electromagnetism-like mechanism algorithm (MELMA) [33], modified Colliding Bodies Optimization algorithm (MCBOA) [34], moth swarm algorithm (MSA) [35], improved imperialist competitive algorithm (IICA) [36], cuckoo optimization algorithm (COA) [37], Gaussian bare-bones imperialist competitive algorithm (GBBICA) [38], and mathematical programming algorithm (MPA) [39]. In [18–20], different variants of GA have been developed in which GA has been improved first and then combined with another method for handling constraints of OPF problem. In fact, Decoupled Quadratic Load Flow has been used in [18] for dealing with OPF problem while IGA has acted as an optimization tool for searching optimal solutions. Hybrid IGA has been applied in both [19,20] while incremental power flow model has been employed in [19], but boundary method has been used in [20]. Conventional PSO and two other improved versions have been suggested, respectively, for each OPF problem in [23–25]. In [19], five velocity-updating formulas have been proposed for EADPSO while ant colony search (ACS) has acted as operator for choosing the most appropriate model for each solution. Contrary, EADPSO and PG-CF-PSO have determined more effective direction for updating velocity by using pseudo-gradient theory and used constriction factor (CF) for focusing on potential search zone. The final comparison results have revealed that EADPSO has become more efficient than conventional PSO but less effective than PG-CF-PSO in terms of solution quality and solution searching speed. The authors in [27,29] have made a big effort in improving the performance of improved versions of BBOA and TLBO. However, the obtained results compared to BBOA and TLBO could not show any superiority of ARCBBOA and ITLBO over BBOA and TLBO. Among remaining methods, MSA is a new method applied to the problem and the comparison results show its strong search ability and stand out over other methods including most above-mentioned methods.

In order to solve the above-mentioned complicated problem, this paper proposes a method to modify the conventional cuckoo search algorithm, namely, the high-performance cuckoo search algorithm (HPCSA). In this paper, the proposed HPCSA is first developed by carrying an adaptive mutation technique with two modifications of the original mutation of conventional cuckoo search algorithm (CCSA) [40]. The first proposes two more equations for updating new solutions, adding to the original mutation model. However, only one out of the three equations needs to be determined for using the adaptive mutation technique for each considered solution depending on the solution's fitness function value. Thus, the second is proposed to establish a decision of using the most appropriate equation for each solution by comparing the fitness function index of each solution compared to the fitness of the best solution and the average fitness index of all solutions compared to the best solution. The second modification is used for the purpose of supporting the first one effectively in its function, so that it can produce high quality solutions. Through the adaptive mutation technique, the proposed method can diversify its search due to exploiting local search and global search in between small and large zones.

The proposed method and CCSA are implemented based on the numerical results through the tests on other systems with different types of objective functions and adifferent set of constraints to demonstrate the effectiveness and robustness of the proposed technique. In addition, the proposed method is also compared to other existing methods, and then, its efficiency is analyzed and concluded. The main contributions to power system optimization field are as follows:

(i) Point out drawbacks of conventional Cuckoo search algorithm clearly and propose improvements on conventional Cuckoo search algorithm effectively
(ii) Present a clear description for handling constraints, namely selection of decision variables and calculation of dependent variables.
(iii) Investigate performance of the proposed method by testing on different systems with different constraints ranging from small-scale systems to large-scale systems, from simple constraint set to complicated constraint set related to thermal generating units and transmission power networks.

This paper is organized as follows: The introduction is presented in Section 1. Section 2 analyzes the economic load dispatch problem formulation. The classical cuckoo search algorithm is recalled, and the proposed method is developed in Section 3. The implementation of proposed method for solving load dispatch problems is introduced in Section 4. The study cases and discussion of the results are given in Section 5 and Appendix A. Finally, the conclusions are stated in Section 6.

2. Analysis for Economic Load Dispatch

2.1. Objective Function

Minimizing the total cost of electricity generation, the objective function of the ELD problem is considered as follows.

$$\text{Min } F = \sum_{i=1}^{N} F_i(P_i) \tag{1}$$

where $F_i(P_i)$ the ith fuel cost function, and can be represented in quadratic form as follows [2]:

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

Considering the valve-point loading effects of the generating units, this fuel cost function has non-convex form, as shown in Equation (3). For better comparison of the complex between the quadratic form without valve point loading effects and the non-convex form with the valve effects, Figure 1 is constructed. As seen from the figure, non-convex form is a challenge for optimization tools.

$$F_i(P_i) = a_i + b_i P_i + c_i P_i^2 + |e_i \times \sin(f_i \times (P_{i,\min} - P_i))| \tag{3}$$

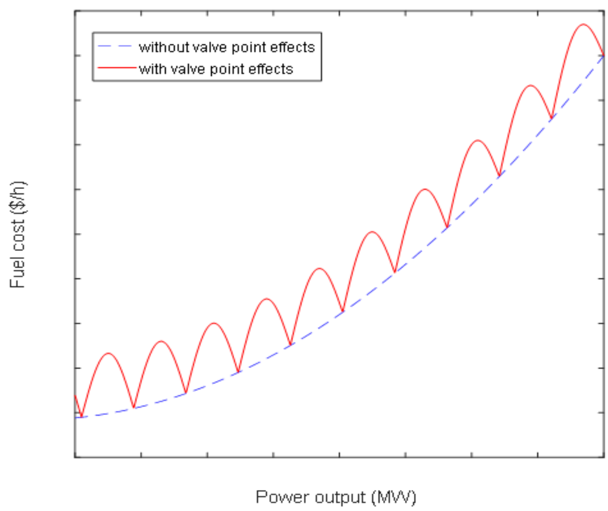

Figure 1. Fuel cost function for the case of single fuel option.

The fuel cost function for the generating units, which are supplied with multiple fuel options, is mathematically formulated described as follows [21]:

$$F_i(P_i) = \begin{cases} a_{i1} + b_{i1} P_i + c_{i1} P_i^2, \text{ fuel } 1, \; P_{i,\min} \leq P_i \leq P_{i1,\max} \\ a_{i2} + b_{i2} P_i + c_{i2} P_i^2, \text{ fuel } 2, \; P_{i2,\min} \leq P_i \leq P_{i2,\max} \\ \vdots \\ a_{ij} + b_{ij} P_i + c_{ij} P_i^2, \text{ fuel } j, \; P_{ij,\min} \leq P_i \leq P_{ij,\max} \end{cases} \tag{4}$$

The fuel cost function can combine multiple fuel (MF) options and valve point effects (VPF), as depicted in Figure 2 and expressed by [15]:

$$F_i(P_i) = \begin{cases} a_{i1} + b_{i1}P_i + c_{i1}P_i^2 + |e_{i1} \times \sin(f_{i1} \times (P_{i1,\min} - P_i))|, \text{ for fuel } 1, \ P_{i,\min} \le P_i \le P_{i1,\max} \\ a_{i2} + b_{i2}P_i + c_{i2}P_i^2 + |e_{i2} \times \sin(f_{i2} \times (P_{i2,\min} - P_i))|, \text{ for fuel } 2, \ P_{i2,\min} \le P_i \le P_{i2,\max} \ , \ j = 1, \ldots, m_i \\ \vdots \\ a_{ij} + b_{ij}P_i + c_{ij}P_i^2 + |e_{ij} \times \sin(f_{ij} \times (P_{ij,\min} - P_i))|, \text{ for fuel } j, \ P_{ij,\min} \le P_i \le P_{ij,\max} \end{cases} \quad (5)$$

where e_{ij} and f_{ij} are fuel cost coefficients for fuel type jth of unit ith reflecting valve-point effects and m_i is the number of fuel types of the thermal unit ith.

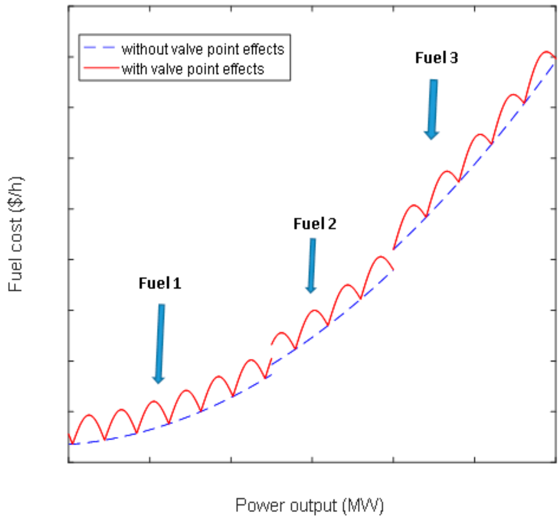

Figure 2. Fuel cost function for the case of multi-fuel options.

2.2. Constraints

Thermal Generating Unit

Operating Generator constraints: Active and reactive power and working voltage of each generator must satisfy the following inequalities.

$$P_{i,\min} \le P_i \le P_{i,\max}; \ i = 1, \ldots, N \quad (6)$$

$$Q_{i,\min} \le Q_i \le Q_{i,\max}; \ i = 1, \ldots, N \quad (7)$$

$$V_{i,\min} \le V_i \le V_{i,\max}; \ i = 1, \ldots, N \quad (8)$$

where $Q_{i,\min}$ and $Q_{i,\max}$ are the lower and upper reactive power output of generator ith, respectively; $V_{i,\min}$ and $V_{i,\max}$ are the allowed minimum and maximum voltage of generator ith, respectively; Q_i and V_i are the reactive power output and working voltage of generator ith, respectively.

Prohibited Operating Zone constraints: Prohibited operating zones (POZ) exist in the input–output curve of each generator due to the steam valve operation or vibration in its shaft bearing. Therefore, the operating region of a generating unit with POZ will be broken into several isolated feasible sub-regions. The mathematical model of POZ is given in the following formula:

$$P_i \in \begin{cases} P_{i,\min} \le P_i \le P_{i1}^{lower} \\ P_{ik}^{upper} \le P_i \le P_{ik+1}^{lower} \ ; \ k = 1, \ldots, NPOZ_i \\ P_{i\mathrm{NPOZ_i}}^{upper} \le P_i \le P_{i,\max} \end{cases} \quad (9)$$

where $NPOZ_i$ is the number of prohibited zones of unit i; and P_{ik}^{upper} and P_{ik}^{lower} are upper and lower bounds for prohibited zone kth of unit ith, respectively; and kth is the POZ number of thermal unit ith.

Spinning reserve constraint: To enhance the stabilization operation of power systems, active power spinning reserve is required and expressed as follows:

$$\sum_{i=1}^{N} SRP_i \geq SRP \tag{10}$$

where SRP is the total active power that the power systems require for spinning reserve; SRP_i is the active power spinning reserve of thermal generating unit ith and calculated as follows:

$$SRP_i = \min\{P_{i,\max} - P_i, SRP_{i,\max}\};\ i = 1, \ldots, N \tag{11}$$

where $SRP_{i,\max}$ is the maximum active power that thermal generating unit ith can contribute to spinning reserve of the power system.

Real power balance constraints: The total real power output of generating units satisfies total load demand plus system power losses

$$\sum_{i=1}^{N} P_i = P_D + P_L \tag{12}$$

and the total power loss is calculated using Kron's formula as follows:

$$P_L = \sum_{i=1}^{N}\sum_{j=1}^{N} P_i B_{ij} P_j + \sum_{i=1}^{N} B_{0i} P_i + B_{00} \tag{13}$$

In addition, the active power balance constraints can be expressed with respect to terms in transmission lines as follows

$$P_{Gi} - P_{di} = V_i \sum_{j=1}^{N_b} V_j \left[G_{ij}\cos(\delta_i - \delta_j) + B_{ij}\sin(\delta_i - \delta_j)\right];\ i = 1, \ldots, N_b \tag{14}$$

where G_{ij} and B_{ij} are the conductance and the susceptance of transmission line ijth connecting bus ith and bus jth.

Reactive power balance constraints: Similar to the constraint model in Equation (14), reactive power balance can be expressed as follows:

$$Q_{Gi} + Q_{ci} - Q_{di} = V_i \sum_{j=1}^{N_b} V_j \left[G_{ij}\sin(\delta_i - \delta_j) - B_{ij}\cos(\delta_i - \delta_j)\right];\ i = 1, \ldots, N_b \tag{15}$$

in which, the presence of capacitor banks is a result of improving voltage and reduction of power losses; Q_{di} is reactive power requirement of load at bus ith; and Q_{ci} is the reactive power generation of capacitor banks installed at bus ith and is constrained by the following inequality:

$$Q_{ci,\min} \leq Q_{ci} \leq Q_{ci,\max};\ i = 1, \ldots, N_c \tag{16}$$

where $Q_{ci,\min}$ and $Q_{ci,\max}$ are the minimum and maximum reactive power generation of capacitor banks at bus i, respectively, and N_c is the number of buses where capacitors are installed.

Transformer tap constraints: The secondary voltage of a transformer corresponds to transformer tap location. Thus, the tap of the transformer should be set to the predetermined range shown in Equation (17):

$$T_{i,\min} \leq T_i \leq T_{i,\max};\ i = 1, \ldots, N_t \tag{17}$$

where $T_{i,\min}$ and $T_{i,\max}$ are the minimum and maximum transformer tap locations at bus ith respectively; T_i is the current tap location at bus ith; and N_t is the total number of buses where transformers are located.

Load bus voltage constraints: The voltage is one of the most important power quality criteria. Consequently, a stability voltage within the following range must supply the load as follows:

$$V_{loadi,\min} \leq V_{loadi} \leq V_{loadi,\max};\ i = 1, \ldots, N_{load} \tag{18}$$

where $V_{loadi,\min}$ and $V_{loadi,\max}$ are the minimum and maximum voltages at bus ith that loads at this bus can work, respectively.

Conductor capacity constraints: The capacity of conductor with respect to the maximum current is also represented as the maximum apparent in power system. The capacity must always satisfy the following rule.

$$S_{condi} \leq S_{condi,\max};\ i = 1, \ldots, N_{cond} \tag{19}$$

where N_{cond} represents the total number of conductors (transmission lines); $S_{condi,\max}$ is the capacity of conductor ith; and S_{condi} is the apparent power flow in conductor ith calculated by:

$$S_{condi} = \max\{|S_{nm}|, |S_{mn}|\} \tag{20}$$

where S_{nm} and S_{mn} are the apparent power flow from buses nth to mth and from buses mth back to nth, respectively.

3. Approaches

3.1. Classical Cuckoo Search Algorithm

Classical cuckoo search algorithm (CCSA) is composed of two generations for producing new solutions and two times for selection of promising solutions [40]. The first generation is carried out by using Lévy Flights stage and the second generation is done by using mutation operation called discarding identified eggs. The whole search process of CCSA is summarized in the four following steps:

Step 1: Lévy Flights for the first generation is calculated by the following inequality

$$S_{condi} \leq S_{condi,\max};\ i = 1, \ldots, N_{cond} \tag{21}$$

where $\alpha > 0$ is the step size ranging from 0 to 1; Lévy (β) is calculated as in Ref. [40]; X_s is solution s that is stored from initialization or from the end of the loop procedure and $s = 1, \ldots, N_{nest}$ (where N_{nest} is the number of nests or the number of solutions in the population).

Step 2: Selection for keeping promising solutions: There are two solutions X_s and Y_s at each nest s. Thus, only one solution is kept and another must be discarded by using the formula below.

$$Z_s = \begin{cases} Y_s & if\ FF(Y_s) < FF(X_s) \\ X_s & else \end{cases} \tag{22}$$

Step 3: Mutation operation for the second generation: The second generation of new solutions is carried out here for improving the solution quality.

$$U_s = \begin{cases} Z_s + \text{rand}(Z_{r1} - Z_{r2})\ if\ \text{rand}_{Us} < PP \\ Z_s \qquad\qquad otherwise \end{cases} \tag{23}$$

where Z_{r1} and Z_{r2} are two randomly picked solutions from the current population, $rand_{Us}$ is randomly produced within the range from 0 to 1 for solution U_s, and PP is a predetermined probability for producing new solutions.

Step 4: Selection for keeping promising solutions: At the end of each iteration, selection operation is repeated for retaining promising solutions.

$$X_s = \begin{cases} U_s & if\, FF(U_s) < FF(Z_s) \\ Z_s & else \end{cases} \tag{24}$$

The best solution among the solution set $X = [X_1, \ldots, X_s, \ldots, X_{Nnest}]$ is determined by choosing the solution with the lowest fitness function.

3.2. The Proposed Approach

In this section, the proposed HPCSA is developed by pointing out disadvantages of CCSA, and then solutions for overcoming the disadvantage are proposed. In the mutation technique shown in Equation (23), CCSA updates new solutions for current solution U_s by using a jumping step, which is the difference between two randomly selected solutions Z_{r1} and Z_{r2}. Clearly, the use of the three considered solutions can lead to several limitations such as low performance of local search and easily trapping into local optimum. Besides, the mutation of CCSA is always carried out by using the difference of only two random solutions over the search process with I_{max} iterations. Consequently, the mutation with Equation (23) reduces the diversity of the local search and global search of CCSA. To overcome the drawback, we suggest using the three following mutation modes for the current population.

$$\text{rand}/1: \quad U_s = Z_s + \text{rand}(Z_{r1} - Z_{r2}) \tag{25}$$

$$\text{rand}/2: \quad U_s = Z_s + \text{rand}(Z_{r1} - Z_{r2} + Z_{r3} - Z_{r4}) \tag{26}$$

$$\text{rand}/3: \quad U_s = Z_s + \text{rand}(Z_{best} - Z_s + Z_{r1} - Z_{r2}) \tag{27}$$

Among such mutation modes, rand/1 can narrow the search space while rand/2 and rand/3 can reach to larger search zone. The three modes can diversify the search strategy; however, the target is only reached as use of them is appropriate for considered solutions. Here, we classify the three different equations into two groups, small zone search group with rand/1 and large zone search group with rand/2 and rand/3. The condition of using either rand/1 or rand/2 and rand/3 is dependent on the comparison of Δ_s and Δ_{mean}, which are shown in Equations (28) and (29).

$$\Delta_{Z_s} = \left| \frac{FF(Z_s)}{FF(Z_{best})} - 1 \right| \tag{28}$$

$$\Delta_{mean} = \left| \frac{\sum_{s=1}^{N_{nest}} FF(Z_s)}{N_{nest} \times FF(Z_{best})} - 1 \right| \tag{29}$$

where Z_{best} is the best solution among the solution set Z, in which $Z = [Z_1, Z_2, \ldots, Z_{Nnest})$; $FF(Z_{best})$ is the fitness function value of the so-far best solution Z_{best}. Δ_{Zs} is the fitness index of solution Z_s compared to the so-far best solution Z_{best}; Δ_{mean} is the average fitness index of all solutions compared to the best solution.

For the case of $\Delta_{Zs} > \Delta_{mean}$, it means that solution Z_s is still far away the current so-far best solution, thus it needs to be search around the current solution Z_s with a small zone nearby such solution Z_s. On the contrary, for the case that Δ_{Zs} is equal or less than Δ_{mean}, larger jumping step will be applied because current solution is too close to the current best solution. For the latter case, rand/2 or rand/3 is used depending on a random condition of comparing between random number and 0.5. The adaptive mutation is described in detail in Algorithm 1.

Algorithm 1: The proposed mutation technique

```
If        Δ_Zs > Δ_mean % using rand/1
              U_s = Z_s + rand(Z_r1 − Z_r2) % rand/1
    else
              if    rand_s > 0.5 % the second condition
              U_s = Z_s + rand(Z_r1 − Z_r2 + Z_r3 − Z_r4) % rand/2
          else

              U_s = Z_s + rand(Z_best − Z_s + Z_r1 − Z_r2) % rand/3
          end
end
```

4. Implementation

The proposed HPCSA method is implemented for solving ELD problem as follows.

4.1. Selection of Control Variables for Each Solution

As mentioned in Section 3.1, solution X_s is the initial solution produced at the beginning of the implementation of the proposed HPCSA method. Consequently, control variables should be determined and added in each solution X_s while solution Y_s, Z_s, and U_s have the same control variables as X_s. In the paper, the ELD problem is constructed by using two different sets of constraints of which the first set of constraints neglecting all transmission power networks is comprised, and the second set of constraints that consider the transmission power network. The first set of constraints is composed of Equations (6) and (12), while the second set of constraints consists of all constraints. For the case considering the first constraint set, the control variables are all active power output of $(N-1)$ thermal generating units while the control variables are P_i (where i = 2, ..., N), V_i (where i = 1, ..., N), T_i (where i = 1, ..., N_t), Q_{ci} (where i = 1, ..., N_c).

4.2. Processing of Constraints

The equality constraints: For the ELD problem neglecting all constraints in transmission lines, the equality constraint in Equation (6) is exactly met by using the following equation

$$P_1 = P_D + P_L - \sum_{i=2}^{N} P_i \tag{30}$$

where P_L is calculated by using Equation (7) and the violation of P_1 is penalized in fitness function. The detail of calculating P_1 can be referred in Ref. [12]. For the case of considering all constraints in transmission lines, all control variables are added in the OPF program, Mathpower for running. Then, all dependent variables, such as P_1 (active power output of generator at slack bus), Q_i (where i = 1, . . . , N), V_{loadi} (where i = 1, . . . , N_{load}), and $S_{branchi}$ (where i = 1, . . . , N_{branch}) are obtained. Such obtained dependent variables are verified and penalized in fitness function in the case that they are violated.

The inequality constraints: All control variables are checked and repaired. If X_s is higher than the maximum values of control variables, X_s will be assigned to maximum values, $X_{\max}$ and on the contrary, X_s will be set to $X_{\min}$ if it is lower than the minimum values. For the two cases of considering constraints, $X_{\min}$ and $X_{\max}$ are defined as follows

$$X_{\min} = \{P_{2,\min}, \ldots, P_{N,\min}, V_{1,\min}, \ldots, V_{N,\min}, T_{1,\min}, \ldots, T_{N_t,\min}, Q_{c1,\min}, \ldots, Q_{cN_c,\min}\} \tag{31}$$

$$X_{\max} = \{P_{2,\max}, \ldots, P_{N,\max}, V_{1,\max}, \ldots, V_{N,\max}, T_{1,\max}, \ldots, T_{N_t,\max}, Q_{c1,\max}, \ldots, Q_{cN_c,\max}\} \tag{32}$$

$$X_{\min} = \{P_{2,\min}, \ldots, P_{N,\min}\} \tag{33}$$

$$X_{\max} = \{P_{2,\max}, \ldots, P_{N,\max}\} \quad (34)$$

4.3. Construction of Fitness Function

The Fitness function can reflect the objective function quality and the violation level of dependent variables. For the cases of neglecting and considering transmission line constraints, the fitness function is as follows

$$FF_s = \sum_{i=1}^{N} F_i(P_{i,s}) + K_{PF} \times \left(P_{1,s} - P_{1,s}^{\lim}\right)^2 + K_{PF} \times \max\left(0, SRP - \sum_{i=1}^{N} SRP_i\right)^2 \quad (35)$$

$$FF_s = \left(\begin{array}{l} \sum\limits_{i=1}^{N} F_i(P_{i,s}) + K_{PF} \sum\limits_{i=1}^{N} \left(P_{1,s} - P_{1,s}^{\lim}\right)^2 + K_{PF} \sum\limits_{i=1}^{N} \left(Q_{i,s} - Q_{i,s}^{\lim}\right)^2 + K_{PF} \sum\limits_{i=1}^{N_{load}} \left(V_{loadi,s} - V_{loadi,s}^{\lim}\right)^2 \\ + K_{PF} \sum\limits_{i=1}^{N_{cond}} \left(S_{condi,s} - S_{condi,s}^{\lim}\right)^2 + K_{PF} \times \max\left(0, SRP - \sum\limits_{i=1}^{N} SRP_{i,s}\right)^2 \end{array} \right) \quad (36)$$

where K_{PF} is the penalty factor, and the limits related to dependent variables are determined by:

$$P_{1,s}^{\lim} = \begin{cases} P_{1,\max} & if\ P_{1,s} > P_{1,\max} \\ P_{1,\min} & if\ P_{1,s} < P_{1,\min} \\ P_{1,s} & else \end{cases} \quad (37)$$

$$Q_{i,s}^{\lim} = \begin{cases} Q_{i,\max} & if\ Q_{i,s} > Q_{i,\max} \\ Q_{i,\min} & if\ Q_{i,s} < Q_{i,\min} \\ Q_{i,s} & else \end{cases} \quad (38)$$

$$V_{loadi,s}^{\lim} = \begin{cases} V_{loadi,\max} & if\ V_{loadi,s} > V_{loadi,\max} \\ V_{loadi,\min} & if\ V_{loadi,s} < V_{loadi,\min} \\ V_{loadi,s} & else \end{cases} \quad (39)$$

$$S_{condi,s}^{\lim} = \begin{cases} S_{condi,\max} & if\ S_{condi,s} > S_{condi,\max} \\ S_{condi,s} & otherwise \end{cases} \quad (40)$$

4.4. The Proposed Algorithm

The completely iterative algorithm for implementation of the HPCSA for solving the ELD problem is described in detail below.

Step 1: Select parameters for the HPCSA including three CCSA parameters such as number of nests N_{nest}, probability of a host bird to discover an alien egg in its nest PP, and maximum number of iterations I_{max}.

Step 2: Select control variables for each solution by using Section 4.1.

Produce an initial population randomly so that $X_{\min} \leq X_s \leq X_{\max}$ is always satisfied.

Step 3: Handle equality constraints by using Section "*The equality constraints*".

Calculate fitness function by using either Equation (35) or (36).
Choose the best solution with the lowest fitness function value.
Set current iteration to 1, I_{cur} = 1.

Step 4: Produce the first new solution by using Equation (21).

Dealing with inequality constraints by using Section "*The inequality constraints*".

Step 5: Calculate fitness function by using either Equation (35) or (36).

Perform section operation by using Equation (22).

Step 6: Produce the second new solutions by using proposed Algorithm 1.

Dealing with inequality constraints by using Section "*The inequality constraints*".

Step 7: Calculate fitness function by using either Equation (35) or (36).

Perform section operation by using Equation (24).

Step 8: Determine the best solution with the lowest fitness function value.

Step 9: Determine the condition to stop the search process: If $I_{cur} < I_{max}$, $I_{cur} = I_{cur} + 1$ and back to Step 4. Otherwise, stop the search process and accept an optimal solution.

5. Case Study and Discussion of the Results

In order to verify the effectiveness, robustness, and convergence speed of the impact of the proposed method, five main power systems, the 10-unit power system, 15-unit power system, IEEE-30 bus power system, IEEE-57 bus power system and IEEE-118 bus power system, are employed. The detail of the five main systems in addition to the selections of population N_{nest} and the maximum number of iterations I_{max} are summarized in Table 1. The proposed algorithm is to run 50 independent trials for case 1 and case 2, 100 independent trials for case 3 and case 4, and 200 independent trials for case 5 on a 2.4 GHz PC with 4 GB of RAM.

In addition to the two control parameters, *PP* is also tuned from 0.1 to 1 with a change of 0.1 for determining the best optimal solution. In order to demonstrate the effectiveness and robustness of the proposed adaptive mutation technique, we also run CCSA for all considered study cases above. For implementation of CCSA, control parameter setting is also carried out similarly as the proposed method.

Table 1. Description of five main employed power systems and the selections of control parameters.

Name	Description	Fuel Cost Function	Constraints	Selection of	Selection of
		Eq.	Eqs.	N_{nest}	I_{max}
Case 1	10-unit power system	(4)	(6), (12)	10	100
Case 2	15-unit power system	(2)	(6), (9) ÷ (12)	10	120
Case 3	The IEEE-30 bus power system	-	-	-	-
Sub-case 3.1	Single fuel with quadratic function	(2)	(6) ÷ (8), (14) ÷ (20)	10	100
Sub-case 3.2	Single fuel with VPLE and POZ constraints	(4)	(6) ÷ (8), (14) ÷ (20)	10	100
Sub-case 3.3	Multi fuels without VPLE	(3)	(6) ÷ (9), (14) ÷ (20)	10	100
Case 4	The IEEE-57 bus power system	(2)	(6) ÷ (8), (14) ÷ (20)	15	200
Case 5	The IEEE-57 bus power system	(2)	(6) ÷ (8), (14) ÷ (20)	20	250

5.1. Case 1: The 10-Unit System with Multi Fuels and Four Load Cases

In this section, we investigate the performance of the proposed method on a system with 10 units using multi fuel options for four sub-cases in which sub-case 1.1 considers load of 2400 MW, sub-case 1.2 considers load of 2500 MW, sub-case 1.3 considers load of 2600 MW, and sub-case 1.4 considers load of 2700 MW. The data of the system is taken from Ref. [3].

In addition to the implementation of CCSA, we also implement other popular methods such as Particle swarm optimization (PSO), Firefly algorithm (FA), and Flower pollination algorithm (FPA) for solving four subcases of case 1 for comparison. For running the additional methods, we set the population of all methods to 10 whilst the maximum number of iterations is set to 200. The setting aims to balance the number of new solution evaluations between CCSA, the proposed HPCSA, and other methods. Besides, other control parameters of these methods are also set to different values in determined ranges. For instance, two acceleration factors c_1 and c_2 of PSO are set to different values within 0 and 2.05 with a step size of 0.2, and the probability of FPA has been set to ten values from 0.1 to 1 with a step size of 0.1, while updated step size factor α of FA has been set to 4 values consisting of 0.25, 0.5, 0.75 and 1. The results obtained by these methods and the proposed method are reported in Table 2. In comparison with CCSA, FA, PSO and FPA, it can be seen that the proposed method obtains the lowest minimum cost, and it also obtains the lowest average cost, the lowest highest cost, and the lowest standard deviation, excluding comparison with FPA for subcase 1.3. The indications can confirm the superiority of the proposed method over other ones in terms of the global optimal solution search ability, the stable search ability, and fast convergence to the global solutions.

For more evidence to demonstrate the fast convergence to global optimum of the proposed method over CCSA, one of the best convergence characteristics of CCSA and the proposed method are depicted in Figure 3 for sub-case 1.1 and Figure 4 for sub-case 1.2. The observations from the curves show that the proposed method is always faster than CCSA once the drop of fitness function from the proposed method is significant at the first iterations, and the drop is nearly not clear at the last iterations while CCSA gets not much improvement at the first iterations, and the improvement is still seen at the last iteration. On the other hand, the best fitness function of each run among 50 runs for sub-cases 1.1 and 1.2 are also taken into account in Figures 5 and 6. The figures indicate that most optimal solutions found by the proposed method have approximately equal quality, and the deviation between the worst solution and the best solution is very small, while the fluctuation of CCSA's solutions is very high. Clearly, the stabilization of optimal solution search ability of the proposed method is superior to that of CCSA. Here, we can see three advantages of the proposed method over CCSA, such as better quality solutions, faster convergence, and more stable search ability. Consequently, it obviously results in a conclusion that the proposed method is much better than CCSA for the system with thermal units using multi-fuel options.

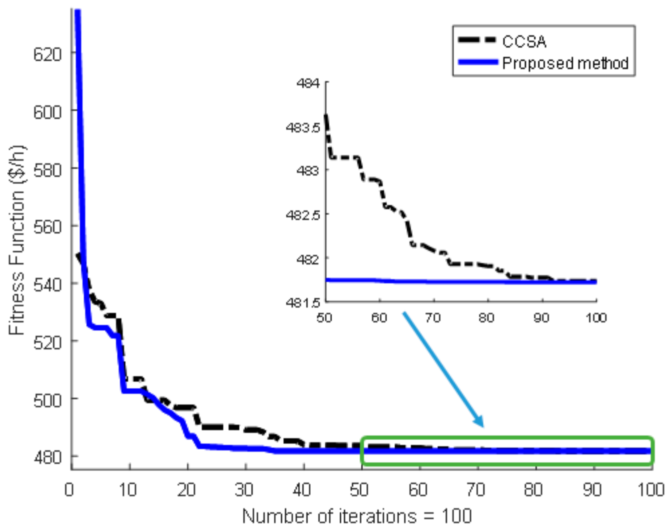

Figure 3. Fitness function curves iteration obtained by the CCSA and proposed method for sub-case 1.1.

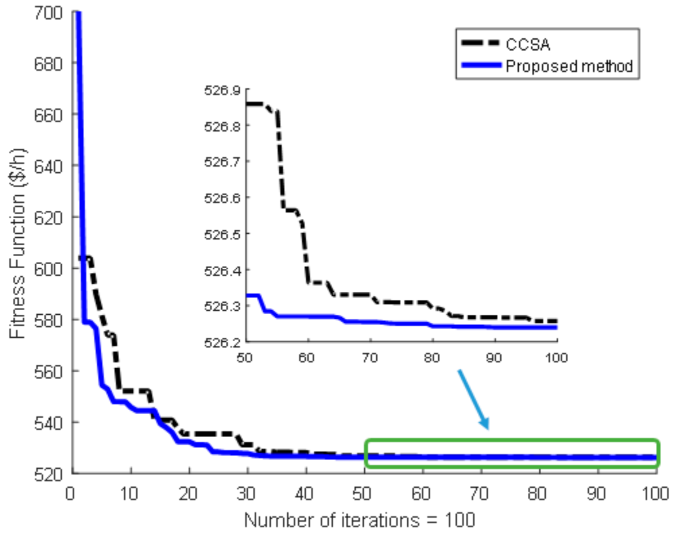

Figure 4. Fitness function curves iteration obtained by the conventional cuckoo search algorithm (CCSA) and proposed method for sub-case 1.2.

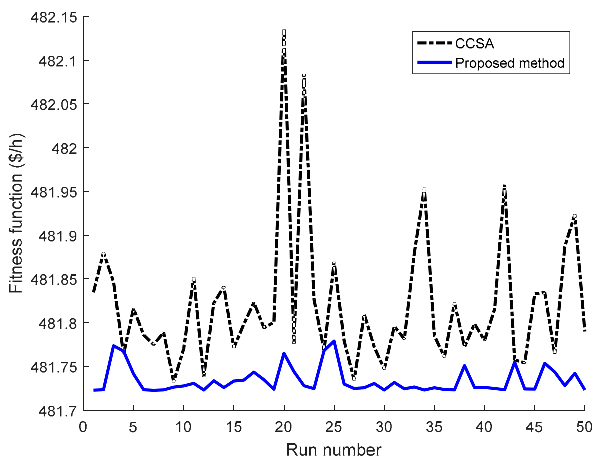

Figure 5. Fitness function of 50 runs obtained by the CCSA and proposed method in case 1.1.

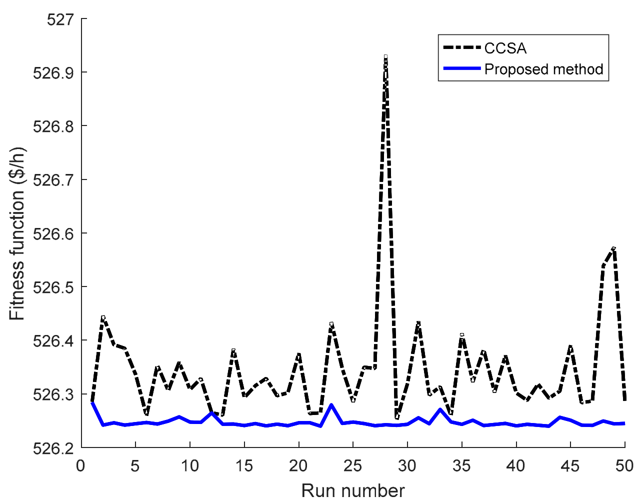

Figure 6. Fitness function of 50 runs obtained by the CCSA and proposed method in case 1.2.

Table 2. The results obtained by Firefly algorithm (FA), particle swarm optimization (PSO), flower pollination algorithm (FPA), CCSA and the proposed high performance cuckoo search algorithm (HPCSA) for subcases of case 1.

Subcase	Method	Best Cost ($/h)	Mean Cost ($/h)	Worst Cost ($/h)	Std. Dev. ($/h)	N_{FES}
1.1	FA	508.742	546.0893	618.8079	198.76584	2000
	PSO	481.7629	487.8131	502.9694	74.71908	2000
	FPA	481.7253	483.9445	490.5896	2.5693	2000
	CCSA	481.727	481.8197	482.136	0.0784	2000
	HPCSA	481.7227	481.734	481.779	0.0148	2000
1.2	FA	536.7378	580.7542	632.082	172.4974	2000
	PSO	526.2895	531.8344	556.5096	33.95347	2000
	FPA	526.2414	526.2926	526.4906	0.0512	2000
	CCSA	526.2522	526.4374	529.0576	0.595	2000
	HPCSA	526.2392	526.2471	526.2843	0.0093	2000
1.3	FA	583.4841	625.3787	659.5854	169.96985	2000
	PSO	574.5034	581.1681	609.9326	49.3633	2000
	FPA	574.3898	574.516	574.9305	0.158	2000
	CCSA	574.41	574.5077	575.8832	0.2109	2000
	HPCSA	574.3813	574.6089	575.4695	0.1862	2000
1.4	FA	639.1301	672.879	726.1476	7.72754	2000
	PSO	623.8252	630.7629	668.3191	43.03717	2000
	FPA	623.812	624.2896	626.5917	0.8572	2000
	CCSA	623.8343	624.3534	635.1689	1.5145	2000
	HPCSA	623.8096	624.1516	626.1913	0.3987	2000

For further investigation of the effectiveness of the proposed method, comparisons with other methods are tabulated in Table 3. In addition to the best costs, the number of fitness evaluations (N_{FES}), which is equal to ($N_{nest} \times I_{max}$) for one generation-based methods and ($2 \times N_{nest} \times I_{max}$) for two generations-based methods, is also reported in the table. As known, the high population size and/or the high number of iterations can lead to a significant improvement of results. Consequently, a method with lower cost and lower N_{FES} or the same N_{FES} is a more effective method than other compared ones. Based on the comparison criterion, the performance of the proposed method is evaluated for the test case. As observed from the cost for sub-cases, the best cost from the proposed method is less than or approximately equal to that from other ones excluding the cost from AHNN [5] for sub-cases 1.1 and 1.2. Note that power generated by the method [5] is slightly less than the load demand. For comparison, the proposed method has used smaller N_{FES} than all methods whose N_{FES} have been reported. The proposed method has used N_{FES} of 2000 while other methods have used from 3000 to 12,000 in which AISA [10] with 3000, HDEDP [11] with 4000, RCGA [7] and HRCGA [7] with 8000, and DE [8] with 12,000. Clearly, the convergence speed to global optimum of the proposed method is much faster than other ones. There is no evaluation executed on the comparison of the proposed method with others such as HNN [3], HA [4], AHNN [5], ILNN [6], and MEP [9] because HNN [3], HA [4], AHNN [5] and ILNN [6] are not the family of meta-heuristic algorithms, and no information was reported for MEP in [9]. The analysis of the results indicates that the proposed method can yield approximate or better solutions than others while N_{FES} of the proposed method is much lower than that from others. As a result, it can be concluded that the proposed method is more effective than other methods for the system.

Table 3. The result comparison about cost ($/h) between methods in case 1.

Method	Sub-Case 1.1	Sub-Case 1.2	Sub-Case 1.3	Sub-Case 1.4	N_{FES}
HNN [3]	487.780	526.130	574.260	626.120	-
HRCGA [7]	481.7226	526.2388	574.3808	623.8092	8000
RCGA [7]	481.7233	526.2393	574.3966	623.8094	8000
DE [8]	481.723	526.239	574.381	623.809	12,000
HA [4]	488.500	526.700	574.030	625.180	-
AHNN [5]	481.72	526.230	574.370	626.240	-
ILNN [6]	481.740	526.270	574.410	623.880	-
MEP [9]	481.779	526.304	574.473	623.851	-
AISA [10]	481.723	526.24	574.381	623.809	3000
HDEDP [11]	481.723	526.239	574.381	623.809	4000
Proposed method	481.7227	526.239	574.3812	623.8096	2000

5.2. Case 2: The 15-Unit System Considering Prohibited Operating Zones Constraint

The test system consists of 15 units with four ones such as units 2, 5, 6 and 12 constrained by POZ condition. The system load demand is 2650 MW and the spinning reserve of the system is 200 MW. The data of the system is from [9]. The obtained results by CCSA and proposed methods are summarized in Table 4 while the best fitness convergence and the 50 runs are depicted in Figures 7 and 8. The numerical table can evaluate the best optimal solutions exactly while the figures can support the exact evaluation of the stabilization of the 50 runs. The best cost from the proposed method is $32,544.9704, but the cost from CCSA is $32,544.9834. The average cost from the proposed method is also less than that from CCSA. Figure 7 confirms the faster search of the proposed method compared to CCSA while Figure 8 shows a high fluctuation of CCSA, and most runs of CCSA show much higher cost than those from the proposed method. Thus, the proposed method is more effective than CCSA for the system with POZ constraints.

In Table 5, the best cost, average cost, maximum cost and N_{FES} from the proposed method are compared to those from other methods such as DM [13], LIM [14], QIEA [16], IQIEA [16] and IALHN [17]. The comparisons of best cost show that the proposed method can converge to a more effective optimal solution with less best cost than DM [13] and LIM [14] while the proposed method also provides less mean cost and less maximum cost than QIEA and IQIEA. Moreover, the value of N_{FES} from the proposed method, 2000, is also smaller than from QIEA and IQIEA, which is 5000. There was no N_{FES} reported for other ones.

Table 4. The obtained results for the CCSA and proposed method in case 2.

Method	CCSA	Proposed Method
Best cost ($/h)	32,544.9834	32,544.9704
Mean cost ($/h)	32,548.1099	32,547.3881
Worst cost ($/h)	32,559.0233	32,563.9819
Std. dev. ($/h)	2.9503	3.5606
CPU time (s)	0.1513	0.1678

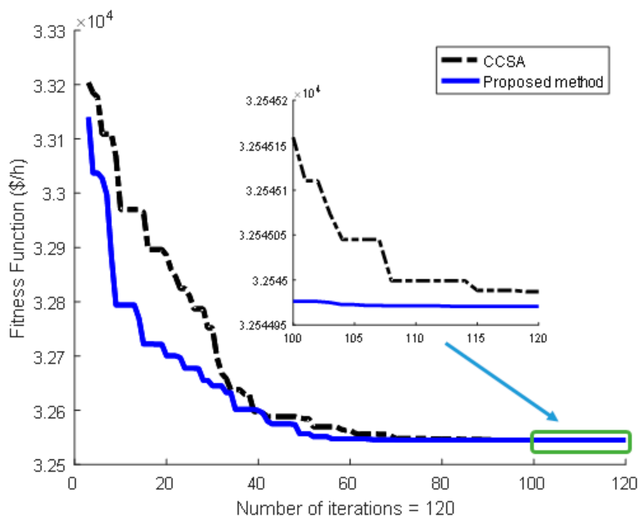

Figure 7. Fitness function curves iteration obtained by the CCSA and proposed method for case 2.

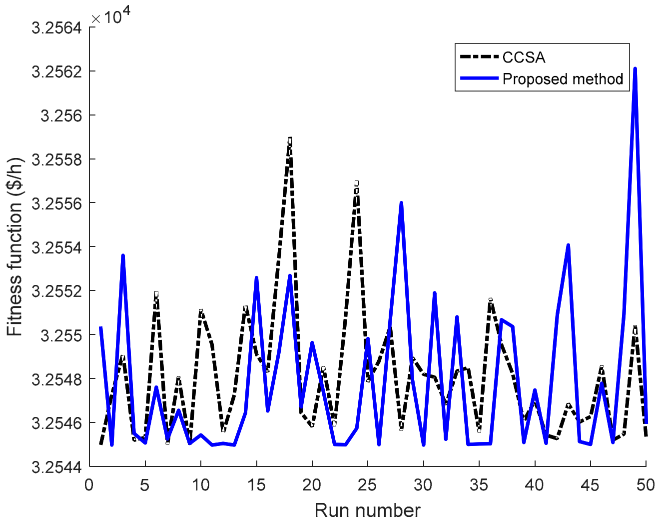

Figure 8. Fitness function of 50 runs obtained by the CCSA and proposed method in case 2.

Table 5. The result comparison about cost ($/h) between methods in case 1.

Method	Total Cost ($/h)	Mean Cost ($)	Maximum Cost ($)	N_{FES}
DM [13]	32,549.80	-	-	-
LIM [14]	32,544.99	-	-	-
QIEA [16]	32,548.48	32,806.89	32,679.54	5000
IQIEA [16]	32,544.97	32,699.56	32,575.35	5000
IALHN [17]	32,544.97	-	-	-
Proposed method	32,544.9704	32,547.3881	32,563.9819	2400

5.3. Case 3: IEEE-30 Bus Power System

The test system is composed of 30 buses consisting of generation 6 buses and 24 load buses, 41 branches, 4 transformers and 9 switchable capacitor banks. The control variables for the system are P_i (i = 1, . . . , 5), V_i (i = 1, . . . , 6), Q_{ci} (i = 1, . . . , 9) and T_i (i = 1, . . . , 4). For the system, there are three sub-cases with three types of fuel cost function where sub-case 3.1 considers single fuel with quadratic form, sub-case 3.2 considers single fuel with nonconvex form and POZ constraints, and sub-case 3.3 considers multi-fuels with piecewise form. The data of the fuel cost functions for these sub-cases are taken from [20,22,41], respectively. The main data belonging to transmission lines of the systems is taken from [25,42].

Figure 9 illustrates the fitness function values of 100 successful runs obtained by CCSA and the proposed method for sub-case 3.1. As seen from the figure, most fitness function values of the proposed method are lower than those of CCSA, and the fluctuation of CCSA is high while the deviation zone of the proposed method is much narrower. In addition, the best costs, mean cost, standard deviation and N_{FES} from CCSA, and the proposed method for the three sub-cases are also given in Table 6 for subcase 3.1, in Table 7 for subcase 3.2 and in Table 8 for subcase 3.3 for comparisons with those from other methods. Observations from such sub-cases show that the proposed method yields better minimum cost than CCSA and most methods excluding MCBOA [34] for sub-cases 3.1, 3.2, and 3.3, HIGA-BM [20], and EADPSO [24] for sub-case 3.3; however, MCBOA has used much high value N_{FES} with 25,000 (for sub-cases 3.1 and 3.2) and 45,000 (for sub-case 3.3) while that of HIGA-BM [20] and EADPSO [24] are 12,000 and 12,500, respectively, but the value from the proposed method is only 2000. Besides, as we check the optimal solution reported in [24], the solution is valid but the exact cost is $956.2325, which is much higher than the reported number of $629.4692. For sub-case 3.2, there is an important note that HIGA-BM [20] has only reported active power of generators for optimal solution, leading to a restriction for checking validation, and the effectiveness of the proposed method cannot be evaluated. The comparison of N_{FES} to the proposed method and other methods indicates that the proposed method is much faster than them because the proposed method used only N_{FES} of 2000 while others have used from N_{FES} of 4560 (HIGA [19]) to 45,000 (MCBOA [34]). For comparisons of mean cost and standard deviation, the three subcases have the same result that those values of other methods are better than those of the proposed method, but the deviation is not insignificant. The results are because of the use of a much higher number of fitness evaluations of other methods.

As a result, it can be concluded that the proposed method is very efficient for solving the system with different cases of fuel cost function. The key variables corresponding to the best fitness function yielded by the proposed method for case 3 are given in Appendix A.

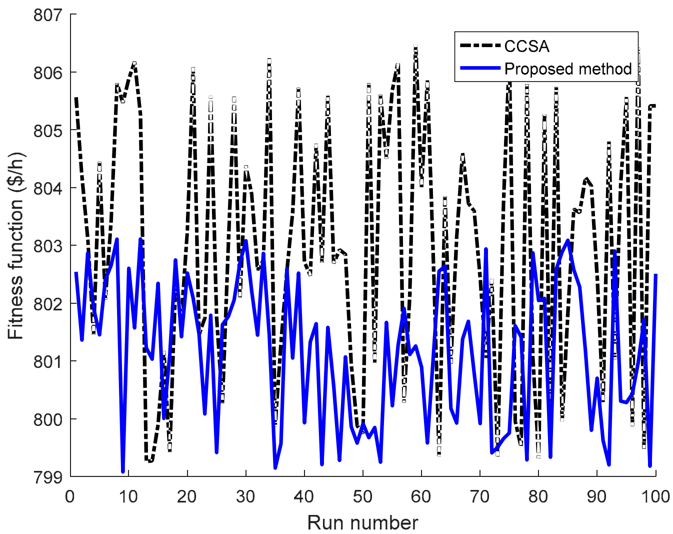

Figure 9. Fitness function of 100 runs obtained by the CCSA and proposed method in sub-case 3.1.

Table 6. The result comparison for subcase 3.1.

Method	Min. Cost ($/h)	Mean Cost ($/h)	Std. Dev. ($/h)	N_{FES}
IGA-EDQLF [18]	799.56	-	-	6000
HIGA [19]	799.56	799.6497	0.0406	4560
HIGA-BM [20]	800.0435	800.122	0.0385	12,000
DE [21]	801.23	801.282	0.0663	-
DE [22]	799.2891	-	-	25,000
PSO [23]	800.41	-	-	-
EADPSO [24]	800.2276	800.2625	0.0303	12,500
BBOA [26]	799.1116	799.1985	-	10,000 (15,000)
ARCBBOA [27]	800.5159	800.6412	-	10,000
TLBO [28]	800.7257	-	-	25,000
ABCA [31]	800.6600	800.8715	-	-
GWO [32]	799.5585	-	-	-
MELMA [33]	799.1821	-	-	-
MCBOA [34]	799.0353	-	-	25,000 (45,000)
MSA [35]	800.5099	-	-	-
CCSA	799.2487	803.0061	2.2011	2000
Proposed method	799.0751	801.1872	1.2327	2000

Table 7. The result comparison for subcase 3.2.

Method	Min. Cost ($/h)	Mean Cost ($/h)	Std. Dev. ($/h)	N_{FES}
HIGA-BM [20]	826.6962	-	-	12,000
MCBOA [34]	830.4531	-	-	25,000 (45,000)
CCSA	831.2097	839.179	7.33	2000
Proposed method	830.7992	835.620	5.95	2000

Table 8. The result comparison for subcase 3.3.

Method	Min. Cost ($/h)	Mean Cost ($/h)	Std. Dev. ($/h)	N_{FES}
DE [22]	650.8224	-	-	25,000
PSO [23]	647.69	647.73	-	-
EADPSO [24]	629.4692	629.6470	0.1159	12,500
BBOA [26]	647.7437	647.7645	-	10,000 (15,000)
ABCA [31]	649.0855	654.0784	-	-
MELMA [33]	649.6309	-	-	-
MCBOA [34]	645.1668	-	-	25,000 (45,000)
MSA [35]	646.8364	646.8603	-	-
CCSA	646.4081	651.882	4.52	2000
Proposed method	646.0569	648.493	1.76	2000

5.4. Case 4: IEEE-57 Bus Power System

In this section, IEEE 57-bus system is employed as a test study to verify the effectiveness and robustness of the proposed method. The system has 80 branches, 57 buses with 7 generator buses and 50 load buses, 15 transformers, and 3 switchable capacitor banks. The main data of the systems is taken from [25,42]. For solving such system, the control variables for the system are P_i (i = 2, . . . , 7), V_i (i = 1, . . . , 7), Q_{ci} (i = 1, 3), and T_i (i = 1, . . . , 15). Similar to other reported tables, the best cost, mean cost, and standard deviation together with the value of N_{FES} from the proposed method, CCSA, and other compared methods are summarized in Table 9 for evaluation. In the table, the best costs yielded by the proposed method and CCSA are $41,669.8269 and $41,694.5162, respectively, and the comparison between the two numbers indicates that the optimal solution from the proposed method can provide a lesser cost of $24.69. Again, the fitness function of 100 independent runs obtained by CCSA and the proposed method depicted in Figure 10 illustrates the search ability superiority of the proposed method over CCSA for 100 considered runs. It is clear that the proposed method can find a higher number of good optimal solutions and a fewer number of bad optimal solutions than CCSA because the number of blue points of the proposed method, which is below the number of black points of CCSA, are higher while the number of black points of CCSA, which is more than the blue points, are higher. Thus, the proposed method is more powerful and stronger than CCSA for searching optimal solutions.

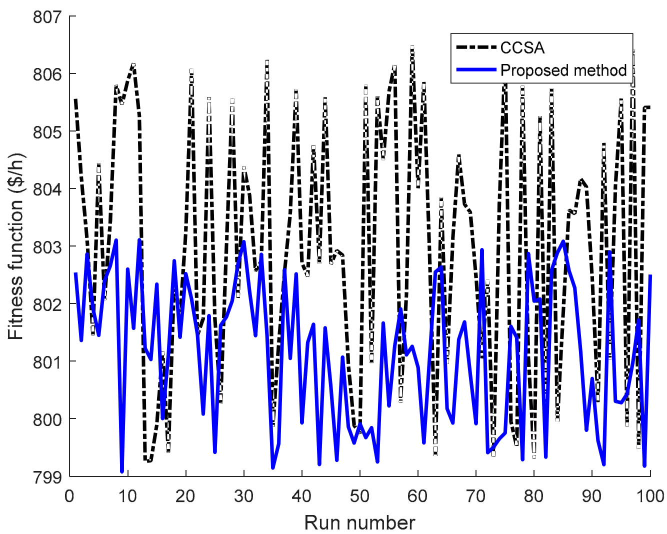

Figure 10. Fitness function of 100 runs obtained by the CCSA and proposed method in case 4.

Table 9. The result comparison for case 4.

Method	Min. Cost ($/h)	Mean Cost ($/h)	Std. Dev. ($/h)	N_{FES}
EADPSO [24]	41,697.54	41,707.69	3.9157	7500
ARCBBOA [27]	41,686	41,718	-	50,000
GSA [30]	41,695.8717	-	-	-
ABCA [31]	41,693.9589	-	-	14,000
PSO [25]	42,109.7231	44,688.4203	1786.3245	5000
PG-CF-PSO [25]	41,688.5004	42,032.7064	551.9334	5000
ITLBO [29]	41,638.3822	-	-	-
IICA [36]	41,738.4352	-	-	110,000
COA [37]	41,901.9977	42,176.3511	610.17	-
GBBICA [38]	41,715.7101	-	-	110,000
CCSA	41,694.5162	42,079.3565	106.18	6000
Proposed method	41,669.8269	41,887.5785	76.19	6000

For comparisons with other methods, there is a standout lower cost from ITLBO [29] of $41,638.3822. However, the validation of the reported solution of ITLBO cannot be carried out because ITLBO has reported only active power outputs of generators for decision variables while other remaining decision variables such as capacitor banks' reactive power output, transformers' tap setting and generators' voltage have been omitted. For other comparisons with the second best method, ARCBBOA [27] and the worst method, PSO [25], the optimal solution yielded by the proposed method can provide a cost decreased by $16.17 and $439.89, respectively. Moreover, the comparison of N_{FES} can reflect fast search ability of the proposed method compared to most methods excluding methods in [25] since the proposed method has used N_{FES} of 6000 while that used by other methods is from 14,000 to 110,000. These methods have used 8000 to 105,000 fitness evaluations higher than the proposed method, but the proposed method has used only 1000 fitness evaluations higher than methods in [25]. Most methods have tended to use high value of N_{FES} for improving their performance. For instance, ARCBBOA could provide the second lowest cost, but it has employed a very high N_{FES} of 50,000, and ABCA [31] has owned the four best costs, but its N_{FES} is still high, up to 14,000. For comparison with mean cost and standard deviation, the proposed method reaches smaller values than most methods, excluding EADPSO [24] and ARCBBOA [27]. However, the two methods have used a higher number of fitness evaluations, namely 7500 for EADPSO [24] and 50,000 for ARCBBOA [27] while that of the proposed method is only 6000. Overall, it can lead to a conclusion that the proposed method is efficient for the system. The key variables corresponding to the best fitness function yielded by the proposed method for case 4 are given in Appendix A.

5.5. Case 5: IEEE-118 Bus Power System

In this section, the proposed method is run on the IEEE-118 bus power system with 54 generator buses, 64 load buses, 186 branches, 9 transformers, and 14 capacitor banks. For the largest system, the number of control variables is also the largest with respect to 130 variables such as active power output of 53 generator excluding generator at slack bus 69, voltage of 54 generators, tap value of 9 transformers, and reactive generation of 14 capacitor banks. The whole data of the system is taken from [25,42].

The best cost, mean cost, standard deviation and the value of N_{FES} from the proposed method, CCSA, and other existing methods are tabulated in Table 10 while the fitness function of 200 runs achieved by CCSA and the proposed method are plotted in Figure 11. Comparison with CCSA indicates that the proposed method can provide an optimal solution with less cost than that of CCSA by $254.10, which is equivalent to a reduction of 0.2%. Figure 11 sees that both CCSA and the proposed method have a high fluctuation among the runs; however, the fluctuation level of CCSA is much higher. Besides, the number of blue points below black points is high but the number of blue points above black points is small while most higher points belong to black points of CCSA. Clearly, the proposed method is more powerful than CCSA in searching for an optimal solution for the system. For comparison with other methods, the proposed method still shows its potential search ability, as its optimal solution leads to less cost than most methods excluding MPA [39]; however, there was no optimal solution reported for the result, leading to a failure of verifying the validation. For cost improvement, the proposed method can improve 3.63%, 10.51%, 6.72%, 2.17%, and 0.195% compared to MCBOA [33], PSO [25], PG-CF-PSO [25], COA [37], and CCSA, respectively. Furthermore, mean cost and standard deviation of the proposed method are also less than those from all methods. Comparison of N_{FES} indicates that the proposed method has used the same fitness evaluations of 10,000 as most methods except MCBOA [33] use 22,500 fitness evaluations. In summary, the proposed method can obtain the best optimal solution and the best stabilization of search ability among all compared methods while its fitness evaluations are equal to or less than that of other ones. Consequently, it can be concluded that the proposed method is the most effective method for case 5. The key variables corresponding to the best fitness function yielded by the proposed method for case 5 are given in Appendix A.

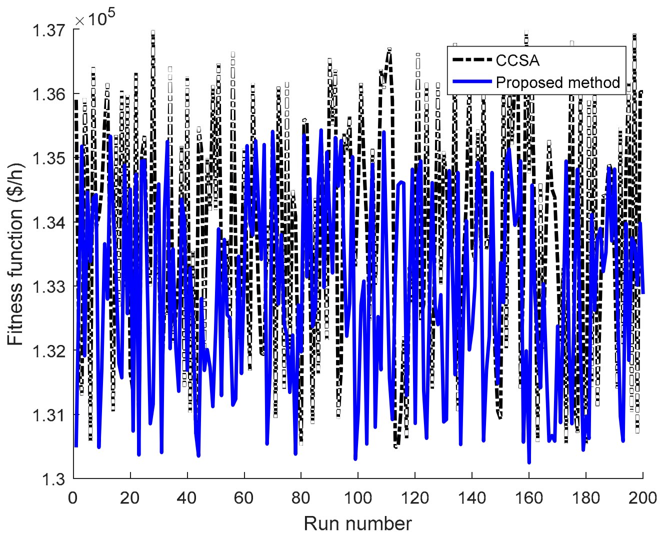

Figure 11. Fitness function of 100 runs obtained by the CCSA and proposed method in case 5.

Table 10. The result comparison for case 5.

Method	Min. Cost ($/h)	Mean Cost ($/h)	Std. Dev.	N_{FES}
MCBOA [34]	135,121.570	-	-	22,500
PSO [25]	145,520.0109	158,596.1725	9454.4231	10,000
PG-CF-PSO [25]	139,604.1326	152,204.2608	6344.7031	10,000
COA [37]	133,110.4316	138,260.4028	4580.9556	-
MPA [39]	130,114.429	-	-	-
CCSA	130,477.3573	132,396.6865	938.431	10,000
Proposed method	130,223.2910	131,873.220	844.366	10,000

5.6. Discussion of Results

In this paper, we propose a high performance cuckoo search algorithm to take advantage of conventional cuckoo search algorithms such as small number of control parameters and easily tuning such control parameters and high possibility of convergence to global optimal solutions. Besides, HPCSA also overcomes disadvantages that CCSA has been facing such as high number of fitness evaluations, low stabilization of searching global optimal solutions, and high standard deviation. In each iteration, CCSA consists of two new solution generations via global search and via local search. The proposed method aims at local search and improves the quality of new solutions obtained by such local search. Thus, the implementation process of such local search of the proposed method is more complicated than that of CCSA, but there is no more additional control parameter needing adjustment.

In comparison with other methods consisting of CCSA and other popular methods, the performance of the proposed HPCSA method has been reflected via the main comparison of the best cost and the number of fitness evaluations. Besides, mean cost and standard deviation have also been added for some cases. On the other hand, *t*-test reflected by *p*-value can give evidence of the improvement level of the proposed method over other ones. However, *p* values of Welch's *t*-test for comparison between the proposed and another are obtained only when enough information consisting of mean cost, standard deviation cost, and the number of runs are reported. Furthermore, the *p*-values can reflect the accurate improvement level of the proposed method over another if the number of runs and the fitness evaluations of the proposed method and compared methods are the same. The mean values and the standard deviation of two methods cannot be compared unless the number of runs of the two methods is equal and the number of fitness evaluations of the two methods is the same. A high number of runs can lead to a more accurate value of mean cost while a high number of fitness evaluations can result in better minimum cost, better mean cost, and better standard deviation cost [43]. In the paper, we have compared the results of the proposed method with more than twenty methods while the number of runs and the fitness evaluation of these compared methods are completely different. Thus, we could not run the proposed method with the same information as each compared method. As a result, we calculate *p*-values for cases with sufficient conditions. For other cases, we focus on the best cost and the number of fitness evaluations as priority comparison criterion and then mean cost and standard deviation are compared for more accurate evaluation. In Table 11, *p*-values of Welch's *t*-test for comparison of the proposed method and other methods for four subcases of case 1 are given. For evaluation of the *p*-values, significance level $t_\alpha = 0.05$ is considered, and calculated *p*-values can be either less than 0.05 or higher than 0.05. If the *p*-value of compared method is much smaller than 0.05, the improvement of the proposed method is highly significant. On the contrary, if *p*-values are much higher than 0.05, the improvement of the proposed method over a compared method is insignificant. As seen from *p*-values in the table, it can be pointed out that most numbers are smaller than 0.05 excluding the *p*-value of FPA for subcase 1.3, which is approximately 0.3. The *p*-value means that there is insignificant improvement here for the proposed method over FPA. In order to explain the *p*-value, mean cost and standard deviation of FPA are compared to those of the proposed method. These values of FPA are 574.516 and 0.158, respectively, while those of the proposed method are 574.6089 and 0.1862. Clearly, FPA reaches better mean cost and standard deviation cost than the proposed method. However, the best cost of the proposed method is still better than that of FPA, namely 574.3898 of FPA and 574.3813 of the proposed method. For another *p*-value such as <0.0001 of FA for subcase 1.1, it shows that the mean and standard deviation of FA are much higher than those of the proposed method. Namely, those of FA are 546.0893 and 198.7658, respectively, and those of the proposed method are 481.734 and 0.0148, respectively. Clearly, if the proposed method reaches much better mean and standard deviation than another method, the *p*-value is much lower than 0.05. On the contrary, if the *p*-value is much higher than 0.05, the proposed method reaches higher mean and standard deviation cost. As a result, it can lead to a conclusion that the best cost and the number of fitness evaluations are the priority comparison criteria for giving the performance conclusion of compared methods while mean cost and standard deviation cost or *p*-values are the secondary comparison criteria for giving the improvement level of compared methods.

Table 11. *p* values of Welch's *t*-test for comparison of the proposed method and others for case 1.

Subcase	Method	No. Runs	Mean Cost ($/h)	Std. Dev. ($/h)	*t*	df	*p*-Value
1.1	HPCSA	50	481.734	0.0148			
	FA	50	546.0893	198.7658	2.289432	0.062013	<0.0001
	PSO	50	487.8131	74.71908	0.57515	0.439068	~0.02
	FPA	50	483.9445	2.5693	5.630264	433.3116	~0.0001
	CCSA	50	481.8197	0.0784	2.88331	2,860,880	~0.0001
1.2	HPCSA	50	526.2471	0.0093			
	FA	50	580.7542	172.4974	2.234373	0.823381	~0.001
	PSO	50	531.8344	33.95347	3.163606	15.12519	~0.01
	FPA	50	526.2926	0.0512	6.182693	964,386.6	<0.02
	CCSA	50	526.4374	0.595	2.261277	6922.106	<0.02

Table 11. *Cont.*

Subcase	Method	No. Runs	Mean Cost ($/h)	Std. Dev. ($/h)	*t*	df	*p*-Value
1.3	HPCSA	50	574.6089	0.1862			
	FA	50	625.3787	169.9699	2.721194	0.848053	<0.05
	PSO	50	581.1681	49.3633	4.939567	10.05458	<0.001
	FPA	50	574.516	0.158	1.689999	80,046.64	~0.3
	CCSA	50	574.5077	0.2109	2.543561	60,972.02	~0.01
1.4	HPCSA	50	624.1516	0.3987			
	FA	50	672.879	7.72754	44.52862	41.13723	<0.001
	PSO	50	630.7629	43.03717	1.086198	1.322866	<0.001
	FPA	50	624.2896	0.8572	1.032179	3874.28	<0.01
	CCSA	50	624.3534	1.5145	0.911143	1136.704	<0.001

For further investigation of the performance of the proposed method, we continued to increase the number of iterations for CCSA when applied to four subcases of case 1. Table 12 reports the result of CCSA when setting I_{max} to 100, 120 and 140 while the result of the proposed method is obtained by accepting I_{max} = 100. Subcase 1.1 indicates that the best cost, mean cost, and standard deviation of CCSA can be improved when I_{max} is increased. Namely, the best costs are 481.727, 481.7235, and 481.7229 while mean cost and standard deviation are 481.8197 and 0.0784, 481.7473 and 0.0237, and 481.7283 and 0.0047, respectively, corresponding to I_{max} = 100, 120 and 140. In comparison with the best cost of the proposed method, the best cost of CCSA at I_{max} = 140 is still slightly higher but in comparison with mean cost and standard deviation of the proposed method, those of CCSA at I_{max} = 140 are lower. However, mean cost and standard deviation of CCSA at I_{max} = 120 are still higher than those of the proposed method at I_{max} = 100. The analysis of obtained results for subcases 1.2, 1.3 and 1.4 are also nearly similar to subcase 1.1.

In summary, it can be concluded that the proposed method can converge to the better optimal solutions, own a more stable search ability, and reach faster convergence with smaller number of fitness evaluations than CCSA.

Table 12. Result comparisons between CCSA and HPCSA for case 1 with different I_{max} of CCSA.

Sub-Case	Method	I_{max}	Best Cost ($/h)	Mean Cost ($/h)	Worst Cost ($/h)	Std. Dev. ($/h)
1.1	CCSA	100	481.727	481.8197	482.136	0.0784
		120	481.7235	481.7473	481.8741	0.0237
		140	481.7229	481.7283	481.7459	0.0047
	HPCSA	100	481.7227	481.734	481.779	0.0148
1.2	CCSA	100	526.2522	526.4374	529.0576	0.595
		120	526.241	526.2611	526.3261	0.0195
		140	526.2395	526.2445	526.2723	0.0053
	HPCSA	100	526.2392	526.2471	526.2843	0.0093
1.3	CCSA	100	574.41	574.5077	575.8832	0.2109
		120	574.384	574.4682	574.7961	0.1054
		140	574.3822	574.4407	574.752	0.0987
	HPCSA	100	574.3813	574.6089	575.4695	0.1862
1.4	CCSA	100	623.8343	624.3534	635.1689	1.5145
		120	623.8185	623.9837	626.1494	0.3997
		140	623.8105	623.9264	626.2978	0.4359
	HPCSA	100	623.8096	624.1516	626.1913	0.3987

6. Conclusions

In this paper, high quality optimization solutions of the considered ELD problem have been found by implementing a high performance cuckoo search algorithm, which was an improved version of the conventional cuckoo search algorithm. The proposed method has applied a new technique for newly updating solutions and obtained much better results than those of CCSA method. The main advantages of the proposed method over CCSA method can be summarized as follows:

(i) Find better optimal solutions with lower number of iterations.
(ii) Own more stable search ability. Most solutions found by the proposed method over a number of runs are approximate and close to the best solutions.

However, when employing the proposed method for dealing with all study cases of the considered ELD problem, several difficulties have not been avoided, such as

(i) Optimal values of predetermined probability have to be tuned within the range from 0 to 1 with a step of 0.1 while the number of nests and the number of iterations are selected by experiment. For small scale systems and simple constraints like case 1 and case 2, optimal solutions are found easily and successfully, but for large scale systems and complicated constraints like cases 3, 4 and 5, finding out optimal solutions is not an easy task.
(ii) For different systems with different constraints, the selection of control variables and the method of calculating all remaining dependent variables as well as the construction of fitness function are very difficult. Appropriate selections can result in high success rate, valid solutions, and high quality solutions, but wrong selections can lead to opposite results.

On the other hand, the performance of the proposed method has been also investigated via comparing with other existing methods of five study cases with different objective function forms and different constraints, especially all constraints of transmission power networks. The result comparisons have indicated that the proposed method has been superior to conventional methods, popular meta-heuristic methods such as PSO, GA, DE, GA and other state-of-the-art methods. As a result, it can lead to a conclusion that the proposed method is an effective optimization tool for searching solutions of the ELD problem with complicated constraints regarding thermal units and transmission power networks.

In the paper, we have applied CCSA and the proposed HPCSA for minimizing electricity generation fuel cost of a set of available thermal generating units for the case of neglecting and considering all constraints of a real power system with the presence of all electricity components. However, the considered ELD problem will become more practical and more valuable if renewable energies such as wind power plants and solar power plants are regarded as main electricity sources together with the thermal units. Currently, the capacity of wind power plants and solar power plants can be up to thousands of megawatts. Besides, solar energy is also stored for use at night, and wind speed is also predicted relatively accuratelyy. Thus, exact mathematical formulation for wind power plants and solar power plants and the implementation of the proposed method for the solutions of the new ELD problem are our future work.

Author Contributions: T.T.N. and D.N.V. have coded the implementation of the proposed method for solving the OPF problem in Matlab. L.V.D. and N.V.Q. have corrected and improved the paper quality and they have been in charge of other duties.

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

Nomenclature

a_i, b_i, c_i, e_i, f_i	Coefficients of cost function for thermal unit i
a_{ij}, b_{ij}, c_{ij}, e_{ij}, f_{ij}	Coefficients of cost function for unit i corresponding to fuel type j fuel cost coefficients for fuel type j of unit i reflecting valve-point effects
B_{ij}, B_{0i}, B_{00}	Coefficients of power loss matrix
N	Number of available thermal units
P_i	Active power generation of thermal unit i
$P_{i,\max}$	Maximum active power generation of unit i
$P_{i,\min}$	Minimum active power generation of unit i
$P_{ij,\min}$	Minimum active power generation for unit i corresponding to fuel type j
P_D	Active power requirement of all loads
P_L	Total active power losses in all transmission lines
$Zbest$	The so-far best solution among all Z considered solutions
$FF(X_s)$, $F(Y_s)$	Fitness function value of solution X_s, Y_s
$FF(U_s)$, $F(Z_s)$	Fitness function value of solution U_s, Z_s

Appendix

Table A1. Optimal solutions obtained by the proposed method for case 3.

Variable	Subcase 3.1	Subcase 3.2	Subcase 3.3
P_1	176.6652	139.9700	198.8593
P_2	48.5631	54.9999	44.4769
P_5	21.8754	24.3305	18.3710
P_8	20.7430	34.9832	10.0051
P_{11}	11.9756	20.8761	10.0194
P_{13}	12.1570	14.7303	12.0073
V_1	1.1000	1.0907	1.1000
V_2	1.0875	1.0776	1.0790
V_5	1.0634	1.0490	1.0500
V_8	1.0704	1.0574	1.0628
V_{11}	1.1000	1.1000	1.0990
V_{13}	1.0993	1.0833	1.1000
Q_{c1}	0.000	4.8000	4.7000
Q_{c2}	1.7000	0.0000	5.0000
Q_{c3}	5.0000	3.9000	4.1000
Q_{c4}	3.8000	3.9000	0.1000
Q_{c5}	2.6000	5.0000	4.0000
Q_{c6}	5.0000	5.0000	1.7000
Q_{c7}	3.6000	3.7000	3.4000
Q_{c8}	5.0000	5.0000	5.0000
Q_{c9}	2.1000	0.4000	5.0000
T_{11}	1.0500	0.9700	1.0200
T_{12}	0.9400	1.1000	0.9100
T_{15}	1.0000	1.0200	0.9800
T_{36}	0.9800	0.9800	1.0100

Table A2. Optimal solutions obtained by the proposed method for the IEEE-57 bus power system.

Variable	Value	Variable	Value
P_1	141.0787	T_1	0.9000
P_2	100.0000	T_2	1.0600
P_3	44.9664	T_3	1.0000
P_6	63.1274	T_4	0.9600
P_8	460.9205	T_5	0.9900
P_9	99.0144	T_6	1.0100
P_{12}	356.6532	T_7	0.9800
V_1	1.0870	T_8	0.9600
V_2	1.0848	T_9	0.9000
V_3	1.0760	T_{10}	0.9700
V_6	1.0886	T_{11}	1.0300
V_8	1.0973	T_{12}	1.0000
V_9	1.0710	T_{13}	0.9500
V_{12}	1.0709	T_{14}	0.9900
Q_{c1}	10.0000	T_{15}	0.9400
Q_{c2}	5.9000	T_{16}	0.9100
Q_{c3}	6.3000	T_{17}	0.9800

Table A3. Optimal solution obtained by NISSO for the IEEE-118 bus power system.

Variable	Value	Variable	Value	Variable	Value
P_1 (MW)	60.7336	P_{103} (MW)	2.7807	V_{76} (pu)	1.0253
P_4 (MW)	1.0572	P_{104} (MW)	0.519	V_{77} (pu)	1.0293
P_6 (MW)	15.1078	P_{105} (MW)	28.7272	V_{80} (pu)	1.0408
P_8 (MW)	5.8423	P_{107} (MW)	18.9129	V_{85} (pu)	1.0278
P_{10} (MW)	384.509	P_{110} (MW)	35.9693	V_{87} (pu)	0.9903
P_{12} (MW)	72.4772	P_{111} (MW)	31.0835	V_{89} (pu)	1.044
P_{15} (MW)	0.9952	P_{112} (MW)	2.0914	V_{90} (pu)	1.0186
P_{18} (MW)	18.7331	P_{113} (MW)	0	V_{91} (pu)	1.037
P_{19} (MW)	17.5849	P_{116} (MW)	1.0049	V_{92} (pu)	1.0323
P_{24} (MW)	0.0291	V_1 (pu)	1.0218	V_{99} (pu)	1.0374
P_{25} (MW)	193.2565	V_4 (pu)	1.0252	V_{100} (pu)	1.0344
P_{26} (MW)	262.5185	V_6 (pu)	1.0574	V_{103} (pu)	1.031
P_{27} (MW)	40.796	V_8 (pu)	1.0781	V_{104} (pu)	1.003
P_{31} (MW)	7.8957	V_{10} (pu)	1.011	V_{105} (pu)	0.9992
P_{32} (MW)	17.9297	V_{12} (pu)	1.0178	V_{107} (pu)	1.0116
P_{34} (MW)	2.6833	V_{15} (pu)	1.0245	V_{110} (pu)	1.0301
P_{36} (MW)	4.5349	V_{18} (pu)	1.0145	V_{111} (pu)	1.0383
P_{40} (MW)	26.7448	V_{19} (pu)	1.0421	V_{112} (pu)	1.028
P_{42} (MW)	66.5657	V_{24} (pu)	1.1	V_{113} (pu)	1.0262
P_{46} (MW)	20.7819	V_{25} (pu)	1.0706	V_{116} (pu)	1.0422
P_{49} (MW)	185.6132	V_{26} (pu)	1.0288	Q_{c5} (MVAr)	−39.5
P_{54} (MW)	46.0838	V_{27} (pu)	0.9869	Q_{c34} (MVAr)	1.5
P_{55} (MW)	46.1449	V_{31} (pu)	1.0142	Q_{c37} (MVAr)	−2.3
P_{56} (MW)	21.4044	V_{32} (pu)	1.0233	Q_{c44} (MVAr)	8
P_{59} (MW)	138.1946	V_{34} (pu)	1.0164	Q_{c45} (MVAr)	1.5
P_{61} (MW)	152.5369	V_{36} (pu)	1.0343	Q_{c46} (MVAr)	0.1
P_{62} (MW)	8.2426	V_{40} (pu)	1.0564	Q_{c48} (MVAr)	0
P_{65} (MW)	355.0914	V_{42} (pu)	1.0213	Q_{c74} (MVAr)	7.9
P_{66} (MW)	273.8651	V_{46} (pu)	1.0283	Q_{c79} (MVAr)	6.9
P_{70} (MW)	10.1045	V_{49} (pu)	1.0371	Q_{c82} (MVAr)	20
P_{72} (MW)	0.9537	V_{54} (pu)	1.0359	Q_{c83} (MVAr)	3.4
P_{73} (MW)	25.3063	V_{55} (pu)	1.0345	Q_{c105} (MVAr)	15.1
P_{74} (MW)	3.6106	V_{56} (pu)	1.0625	Q_{c107} (MVAr)	5.2
P_{76} (MW)	36.7909	V_{59} (pu)	1.0696	Q_{c110} (MVAr)	3.3
P_{77} (MW)	5.674	V_{61} (pu)	1.0464	T_8 (pu)	1.03
P_{80} (MW)	429.3572	V_{62} (pu)	1.0683	T_{32} (pu)	1.09
P_{85} (MW)	0.6222	V_{65} (pu)	1.0519	T_{36} (pu)	0.99
P_{87} (MW)	2.016	V_{66} (pu)	1.047	T_{51} (pu)	1.09
P_{89} (MW)	487.0964	V_{69} (pu)	1.0338	T_{93} (pu)	0.98
P_{90} (MW)	0.3816	V_{70} (pu)	1.0387	T_{95} (pu)	1.01
P_{91} (MW)	4.2073	V_{72} (pu)	1.0467	T_{102} (pu)	1.03
P_{92} (MW)	0.9548	V_{73} (pu)	1.0263	T_{107} (pu)	0.9
P_{99} (MW)	7.1701	V_{74} (pu)	2.7807	T_{127} (pu)	0.97
P_{100} (MW)	231.1577				

References

1. Sinha, N.; Chakrabarti, R.; Chattopadhyay, P.K. Evolutionary Programming Techniques for Economic Load Dispatch. *IEEE Trans. Evol. Comput.* **2003**, *7*, 83–94. [CrossRef]
2. Wood, A.J.; Wollenberg, B.F. *Power Generation Operation and Control*, 2nd ed.; Wiley: New York, NY, USA, 1996.
3. Park, J.H.; Kim, Y.S.; Eom, I.K.; Lee, K.Y. Economic load dispatch for piecewise quadratic cost function using Hopfield neural network. *IEEE Trans. Power Syst.* **1993**, *8*, 1030–1038. [CrossRef]
4. Lin, C.E.; Viviani, G.L. Hierarchical economic dispatch for piecewise quadratic cost functions. *IEEE Trans. Power Appar. Syst.* **1984**, *6*, 1170–1175. [CrossRef]
5. Lee, K.Y.; Sode-Yome, A.; Park, J.H. Adaptive Hopfield neural networks for economic load dispatch. *IEEE Trans. Power Syst.* **1998**, *13*, 519–526. [CrossRef]

6. Lee, S.C.; Kim, Y.H. An enhanced Lagrangian neural network for the ELD problems with piecewise quadratic cost functions and nonlinear constraints. *Electr. Power Syst. Res.* **2002**, *60*, 167–177. [CrossRef]
7. Baskar, S.; Subbaraj, P.; Rao, M.V.C. Hybrid real coded genetic algorithm solution to economic dispatch problem. *Comput. Electr. Eng.* **2003**, *2*, 407–419. [CrossRef]
8. Noman, N.; Iba, H. Differential evolution for economic load dispatch problems. *Electr. Power Syst. Res.* **2008**, *78*, 1322–1331. [CrossRef]
9. Park, Y.M.; Won, J.R.; Park, J.B. A new approach to economic load dispatch based on improved evolutionary programming. *Eng. Intell. Syst. Electr. Eng. Commun.* **1998**, *6*, 103–110.
10. Panigrahi, B.K.; Yadav, S.R.; Agrawal, S.; Tiwari, M.K. A clonal algorithm to solve economic load dispatch. *Electr. Power Syst. Res.* **2007**, *77*, 1381–1389. [CrossRef]
11. Balamurugan, R.; Subramanian, S. Hybrid integer coded differential evolution–dynamic programming approach for economic load dispatch with multiple fuel options. *Energy Convers. Manag.* **2008**, *49*, 608–614. [CrossRef]
12. Vo, D.N.; Schegner, P.; Ongsakul, W. Cuckoo search algorithm for non-convex economic dispatch. *IET Gener. Transm. Distrib.* **2013**, *7*, 645–654. [CrossRef]
13. Lee, F.N.; Breipohl, A.M. Reserve constrained economic dispatch with prohibited operating zones. *IEEE Trans. Power Syst.* **1993**, *8*, 246–254. [CrossRef]
14. Fan, J.Y.; McDonald, J.D. A practical approach to real time economic dispatch considering unit's prohibited operating zones. *IEEE Trans. Power Syst.* **1994**, *9*, 1737–1743. [CrossRef]
15. Gaing, Z.L. Particle swarm optimization to solving the economic dispatch considering the generator constraints. *IEEE Trans. Power Syst.* **2003**, *18*, 1187–1195. [CrossRef]
16. Neto, J.X.V.; de Andrade Bernert, D.L.; dos Santos Coelho, L. Improved quantum-inspired evolutionary algorithm with diversity information applied to economic dispatch problem with prohibited operating zones. *Energy Convers. Manag.* **2011**, *52*, 8–14. [CrossRef]
17. Vo, D.N.; Ongsakul, W. Enhanced augmented Lagrange Hopfield network for constrained economic dispatch with prohibited operating zones. *Eng. Intell. Syst.* **2009**, *17*, 199–208.
18. Kumari, M.S.; Maheswarapu, S. Enhanced genetic algorithm based computation technique for multi-objective optimal power flow solution. *Int. J. Electr. Power Energy Syst.* **2010**, *32*, 736–742. [CrossRef]
19. Reddy, S.S.; Bijwe, P.R.; Abhyankar, A.R. Faster evolutionary algorithm based optimal power flow using incremental variables. *Int. J. Electr. Power Energy Syst.* **2014**, *54*, 198–210. [CrossRef]
20. Reddy, S.S.; Bijwe, P.R. Efficiency improvements in meta-heuristic algorithms to solve the optimal power flow problem. *Int. J. Electr. Power Energy Syst.* **2016**, *82*, 288–302. [CrossRef]
21. El-Fergany, A.A.; Hasanien, H.M. Single and multi-objective optimal power flow using grey wolf optimizer and differential evolution algorithms. *Electr. Power Compon. Syst.* **2015**, *43*, 1548–1559. [CrossRef]
22. El Ela, A.A.; Abido, M.A.; Spea, S.R. Optimal power flow using differential evolution algorithm. *Electr. Power Syst. Res.* **2010**, *80*, 878–885. [CrossRef]
23. Abido, M.A. Optimal power flow using particle swarm optimization. *Int. J. Electr. Power Energy Syst.* **2002**, *24*, 563–571. [CrossRef]
24. Vaisakh, K.; Srinivas, L.R.; Meah, K. Genetic evolving ant direction particle swarm optimization algorithm for optimal power flow with non-smooth cost functions and statistical analysis. *Appl. Soft Comput.* **2013**, *13*, 4579–4593. [CrossRef]
25. Vo, D.N.; Schegner, P. An Improved Particle Swarm Optimization for Optimal Power Flow. In *Meta-Heuristics Optimization Algorithms in Engineering, Business, Economics, and Finance*; IGI Global: Hershey, PA, USA, 2013; pp. 1–40.
26. Bhattacharya, A.; Chattopadhyay, P.K. Application of biogeography-based optimization to solve different optimal power flow problems. *IET Gener. Transm. Distrib.* **2011**, *5*, 70–80. [CrossRef]
27. Kumar, A.R.; Premalatha, L. Optimal power flow for a deregulated power system using adaptive real coded biogeography-based optimization. *Int. J. Electr. Power Energy Syst.* **2015**, *73*, 393–399. [CrossRef]
28. Nayak, M.R.; Nayak, C.K.; Rout, P.K. Application of multi-objective teaching learning based optimization algorithm to optimal power flow problem. *Procedia Technol.* **2012**, *6*, 255–264. [CrossRef]
29. Shabanpour-Haghighi, A.; Seifi, A.R.; Niknam, T. A modified teaching-learning based optimization for multi-objective optimal power flow problem. *Energy Convers. Manag.* **2014**, *77*, 597–607. [CrossRef]

30. Duman, S.; Güvenç, U.; Sönmez, Y.; Yörükeren, N. Optimal power flow using gravitational search algorithm. *Energy Convers. Manag.* **2012**, *59*, 86–95. [CrossRef]
31. Adaryani, M.R.; Karami, A. Artificial bee colony algorithm for solving multi-objective optimal power flow problem. *Int. J. Electr. Power Energy Syst.* **2013**, *53*, 219–230. [CrossRef]
32. Ladumor, D.P.; Trivedi, I.N.; Bhesdadiya, R.H.; Jangir, P. A grey wolf optimizer algorithm for Voltage Stability Enhancement. In Proceedings of the Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB), Chennai, India, 27–28 February 2017; pp. 278–282.
33. El-Hana Bouchekara, H.R.; Abido, M.A.; Chaib, A.E. Optimal power flow using an improved electromagnetism-like mechanism method. *Electr. Power Compon. Syst.* **2016**, *44*, 434–449. [CrossRef]
34. Bouchekara, H.R.E.H.; Chaib, A.E.; Abido, M.A.; El-Sehiemy, R.A. Optimal power flow using an Improved Colliding Bodies Optimization algorithm. *Appl. Soft Comput.* **2016**, *42*, 119–131. [CrossRef]
35. Mohamed, A.A.A.; Mohamed, Y.S.; El-Gaafary, A.A.; Hemeida, A.M. Optimal power flow using moth swarm algorithm. *Electr. Power Syst. Res.* **2017**, *142*, 190–206. [CrossRef]
36. Ghasemi, M.; Ghavidel, S.; Ghanbarian, M.M.; Gharibzadeh, M.; Vahed, A.A. Multi-objective optimal power flow considering the cost, emission, voltage deviation and power losses using multi-objective modified imperialist competitive algorithm. *Energy* **2014**, *78*, 276–289. [CrossRef]
37. Le Anh, T.N.; Vo, D.N.; Ongsakul, W.; Vasant, P.; Ganesan, T. Cuckoo optimization algorithm for optimal power flow. In Proceedings of the 18th Asia Pacific Symposium on Intelligent and Evolutionary Systems; Springer: Cham, Switzerland, 2015; Volume 1, pp. 479–493.
38. Ghasemi, M.; Ghavidel, S.; Ghanbarian, M.M.; Gitizadeh, M. Multi-objective optimal electric power planning in the power system using Gaussian bare-bones imperialist competitive algorithm. *Inf. Sci.* **2015**, *294*, 286–304. [CrossRef]
39. Ara, A.L.; Kazemi, A.; Gahramani, S.; Behshad, M. Optimal reactive power flow using multi-objective mathematical programming. *Sci. Iran.* **2012**, *19*, 1829–1836.
40. Yang, X.S.; Deb, S. Cuckoo search via Lévy flights. In Proceedings of the World Congress on Nature & Biologically Inspired Computing (NaBIC 2009), Coimbatore, India, 9–11 December 2009; pp. 210–214.
41. Yuryevich, J.; Wong, K.P. Evolutionary programming based optimal power flow algorithm. *IEEE Trans. Power Syst.* **1999**, *14*, 1245–1250. [CrossRef]
42. Zimmerman, R.D.; Murillo-Sánchez, C.E.; Thomas, R.J. MATPOWER. Available online: http://www.pserc.cornell.edu/matpower (accessed on 10 January 2018).
43. Mernik, M.; Liu, S.H.; Karaboga, D.; Črepinšek, M. On clarifying misconceptions when comparing variants of the Artificial Bee Colony Algorithm by offering a new implementation. *Inf. Sci.* **2015**, *115*, 127–291. [CrossRef]