energies

Article

An Improved DA-PSO Optimization Approach for Unit Commitment Problem

Sirote Khunkitti [1], Neville R. Watson [2], Rongrit Chatthaworn [1], Suttichai Premrudeepreechacharn [3] and Apirat Siritaratiwat [1,*]

[1] Department of Electrical Engineering, Faculty of Engineering, Khon Kaen University, Khon Kaen 40002, Thailand; sirote_khunkitti@kkumail.com (S.K.); rongch@kku.ac.th (R.C.)

[2] Department of Electrical and Computer Engineering, University of Canterbury, Christchurch 8140, New Zealand; neville.watson@canterbury.ac.nz

[3] Department of Electrical Engineering, Faculty of Engineering, Chiang Mai University, Chiang Mai 50200, Thailand; suttic@eng.cmu.ac.th

* Correspondence: apirat@kku.ac.th; Tel.: +66-89-712-2893

Received: 18 May 2019; Accepted: 17 June 2019; Published: 18 June 2019

Abstract: Solving the Unit Commitment problem is an important step in optimally dispatching the available generation and involves two stages—deciding which generators to commit, and then deciding their power output (economic dispatch). The Unit Commitment problem is a mixed-integer combinational optimization problem that traditional optimization techniques struggle to solve, and metaheuristic techniques are better suited. Dragonfly algorithm (DA) and particle swarm optimization (PSO) are two such metaheuristic techniques, and recently a hybrid (DA-PSO), to make use of the best features of both, has been proposed. The original DA-PSO optimization is unable to solve the Unit Commitment problem because this is a mixed-integer optimization problem. However, this paper proposes a new and improved DA-PSO optimization (referred to as iDA-PSO) for solving the unit commitment and economic dispatch problems. The iDA-PSO employs a sigmoid function to find the optimal on/off status of units, which is the mixed-integer part of obtaining the Unit Commitment problem. To verify the effectiveness of the iDA-PSO approach, it was tested on four different-sized systems (5-unit, 6-unit, 10-unit, and 26-unit systems). The unit commitment, generation schedule, total generation cost, and time were compared with those obtained by other algorithms in the literature. The simulation results show iDA-PSO is a promising technique and is superior to many other algorithms in the literature.

Keywords: dragonfly algorithm; metaheuristic; particle swarm optimization; unit commitment

1. Introduction

The development of electricity markets has made it even more crucial to determine the optimal generator schedule to minimize costs while meeting load demand. Traditional economic dispatch (ED) does not perform decisions on which generators to commit and assumes all generators must be dispatched within their minimum and maximum generator limits. Unit Commitment (UC) is the optimization problem of determining the optimal set of in-service and out-of-service generating units and their output during the scheduling period to minimize the total production costs while satisfying all the constraints [1]. In the UC problem, two decision processes involved are unit scheduling and ED. The unit scheduling process is to determine the on/off status of generating units in each hour of the planning horizon while considering minimum up- and down-time of the units. ED aims to find the optimal power generation of the in-service generating units to meet the load demand and spinning reserve during each hour while maintaining generating unit limits.

The UC problem has been considered to be a large-scale, non-convex, and mixed-integer non-linear combinatorial optimization problem, which makes the UC problem difficult to be solved. In the past, many methods have been proposed to solve the UC problem [2]. Some of the proposed techniques for solving the UC problem are; integer programming [3,4], branch-and-bound methods [5], dynamic programming (DP) [6–11], mixed-integer programming [12], Lagrangian relaxation methods (LR) [13,14], priority list method [15]. However, each of these methods has some drawbacks when solving the UC problem. For instance, the integer and mixed-integer programming methods, which use linear programming to find an integer part of the solution require too large memory for large systems, and this results in a large computation burden. The computation time of the Branch-and-bound increases exponentially with system size. Although DP is flexible, it sometimes requires a large amount of computation time if various constraints are considered. The disadvantage of LR is the difficulty confronted in providing optimal solutions when solving complex problems. The priority list method is fast and easy to implement, but it cannot confirm the quality of the solution for the same reason as LR.

Apart from these traditional techniques, many metaheuristic algorithms have been applied, such as; genetic algorithms (GA) [16], particle swarm optimization (PSO) combined with the Lagrangian relaxation (PSO-LR) [17], evolutionary programming (EP) [18], new genetic approach (NGA) [19], local convergence averse binary particle swarm optimization (LCA-PSO) [20], improved binary particle swarm optimization (IPSO) [20], mutation-based particle swarm optimization (MPSO) [20], a two-stage genetic-based technique (TSGA) [21], inter-coded genetic algorithm (ICGA) [22], binary-coded genetic algorithm (BCGA) [22], simulated annealing (SA) [23], Seeded Memetic algorithm (SM) [23], a hybrid algorithm comprising of particle swarm optimization and grey wolf optimizer (PSO-GWO) [24] and hybrid particle swarm optimization (HPSO) [25]. These have been successfully applied to solving the UC problem due to their ability to find a near global solution and deal with large-scale non-linear problems. Moreover, several works have previously studied the scheduling of generation units in small to large power systems. For example, fuzzy-based particle swarm optimization (FPSO) has been proposed to minimize the operation cost and emission for ships [26], conditional value-at-risk (CVaR) method has been introduced to maximize the expected profit of a microgrid operator [27], a hybrid PSO and selective PSO method (PSO&SPSO) has been used to solve a proposed a day-ahead operational scheduling framework for reconfigurable microgrids (RMGs) [28], a metaheuristic approach based on PSO has been applied to solve an optimal simultaneous hourly reconfiguration and day-ahead scheduling framework in smart distribution systems [29], a stochastic model for optimal scheduling of security-constrained UC associated with demand response (AC-SUCDR) has been presented in [30], a two-stage stochastic programming model has been developed to minimize the expected cost of microgrid under different time-based rate programs [31], and a Fuzzy Self-Adaptive Particle Swarm Optimization (FSAPSO) has been applied to solve multi-operation management of a typical microgrids and of a renewable microgrid [32,33].

Many metaheuristic optimization algorithms have been proposed to solve other types of complex optimization problems such as in an optimal power-flow (OPF). Examples are; grey wolf optimizer (GWO) [34], dragonfly algorithm (DA) [35], ant colony optimization (ACO) [36] and artificial bee colony (ABC) [37]. However, these algorithms cannot solve a mixed-integer combinational optimization problem in their native form. A hybrid dragonfly algorithm and particle swarm optimization (DA-PSO) is a recent optimization method which has been applied to efficiently solve a complex optimization problem which is a multi-objective optimization problem [38]. Nevertheless, it is unable to solve the mixed-integer combinational optimization problem. Therefore, this paper proposes an improved DA-PSO algorithm (iDA-PSO) that can solve the UC problem. This is achieved by applying a sigmoid function to the DA-PSO to find the optimal on/off status of generating units, which is the mixed-integer part of the UC problem. The algorithm is tested of four test systems of differing sizes. Five-unit, six-unit, ten-unit, and 26-unit generating systems are used to investigate the effectiveness of the proposed approach. The simulation results were compared with other algorithms in the literature.

2. Formulation of the UC Problem

The UC problem aims to find the optimal generation schedule, which is gauged by the value of the objective function while satisfying a set of constraints.

2.1. Objective Function

The objective function is the total production costs over the scheduling horizon, and this must be minimized to obtain the optimal generator schedule. The total production costs consist of fuel cost and start-up cost of the operating units. Therefore, the objective function is:

$$TPC = \sum_{t=1}^{T} \sum_{i=1}^{Ng} [f_{Cost}(P^t_{gi}) + ST^t_i(1 - u^{t-1}_i)]u^{\,t}_i \quad (1)$$

where TPC is the total production cost ($), T is the total scheduling period, N_g is the number of generating units, ${P_{gi}}^t$ is the active power generation of the ith unit at time t, ${ST_i}^t$ is the start-up cost of the ith unit at time t, ${u_i}^t$ is the on or off status of the ith unit at time t, and $f_{Cost}({P_{gi}}^t)$ is the fuel cost function of the ith unit for the generator power output ${P_{gi}}^t$ which is calculated as:

$$f_{Cost}(P^t_{gi}) = a_i P^2_{gi} + b_i P_{gi} + c_i \quad (2)$$

where a_i, b_i, and c_i are the fuel cost coefficients of the ith generator.

The start-up cost is the cost of bringing the off-line unit on-line. It depends on the time that the unit has been off-line before starting up which is presented as follows:

$$ST^t_i = \begin{cases} HSC_i & if \quad MDT_i \le T^t_{i,off} \le (MDT_i + CSH_i) \\ CSC_i & if \quad T^t_{i,off} > (MDT_i + CSH_i) \end{cases} \quad (3)$$

where HSC_i is the hot start-up cost of the ith unit, CSC_i is the cold start-up cost of the ith unit, MDT_i is the minimum down-time of the ith unit, ${T^t}_{i,off}$ is the number of off hours of the ith unit until time t and CSH_i is the cold start hour of the ith unit.

2.2. Constraints

The optimization of the objective function must satisfy constraints imposed by the operational requirements. The set of constraints are as follows:

2.2.1. Power Balance Constraint

$$\sum_{i=1}^{Ng} P^t_{gi} u^{\,t}_i = P^t_D \quad (4)$$

where ${P^t}_D$ is the active power demand at time t.

2.2.2. Spinning Reserve Constraint

$$\sum_{i=1}^{Ng} P_{gi(\max)} u^{\,t}_i \ge P^t_D + P^t_R \quad (5)$$

where $P_{gi(\max)}$ is the maximum active power of the ith unit, and ${P^t}_R$ is the active power reserve at time t.

2.2.3. Generation Limit Constraints

$$P_{gi(\min)} \leq P_{gi}^{t} \leq P_{gi(\max)} \tag{6}$$

where $P_{gi(\min)}$ is the minimum active power of the ith unit.

2.2.4. Minimum Up-Time Constraint

$$T_{i,on}^{t} \geq MUT_i \tag{7}$$

where $T^t{}_{i,on}$ is the number of on hours of the ith unit until time t, and MUT_i is the minimum up-time of the ith unit.

2.2.5. Minimum Down-time Constraint

$$T_{i,off}^{t} \geq MDT_i \tag{8}$$

3. Overview of DA-PSO Optimization Algorithm and Related Algorithms

DA-PSO optimization algorithm is a hybrid algorithm which original combined the frameworks of the DA and PSO algorithms. This section aims to describe the formulations and concepts of the related algorithms including DA, PSO, and DA-PSO.

3.1. DA

DA is a metaheuristic method motivated by the flocking behavior of dragonflies in nature [35], and it has been successfully applied to solve complicated optimization problems, such as the OPF problem [39]. There are two main swarming goals of dragonflies, which are hunting (or static swarm), and migrating (or dynamic swarm). These can be related to two main phases of optimization, which are exploitation and exploration phases. The behavior of swarms follows three traditional rules [40]. The first rule is separation, which is to ensure collision avoidance. That is individuals avoid colliding with others in the neighborhood. Secondly, alignment, referring to velocity matching of an individual to that of other individuals in the neighborhood. The other is cohesion meaning the distance away of individuals to the center of mass of the neighborhood. Moreover, since survival is the main propose of any swarm, all the population should be attracted to food sources and repelled by the presence of enemies. Accordingly, the position updating of individuals are imitated from the aforementioned behavior, and can be mathematically formulated as follows:

Separation is formulated as follows:

$$\boldsymbol{S_i} = -\sum_{j=1}^{N} \boldsymbol{X} - \boldsymbol{X_j} \tag{9}$$

where $\boldsymbol{S_i}$ is the separation of the ith individual, N is the number of neighboring individuals, $\boldsymbol{X}$ is the current individual position, $\boldsymbol{X_j}$ is the position of the jth neighboring individual.

Alignment is formulation is:

$$\boldsymbol{A_i} = \frac{\sum_{j=1}^{N} \boldsymbol{V_j}}{N} \tag{10}$$

where $\boldsymbol{A_i}$ is the alignment of the ith individual, $\boldsymbol{Vj}$ is the velocity of the jth neighboring individual.

Cohesion is formulation is:

$$\boldsymbol{C_i} = \frac{\sum_{j=1}^{N} \boldsymbol{X_j}}{N} - \boldsymbol{X} \tag{11}$$

where $\boldsymbol{C_i}$ is the cohesion of the ith individual.

Attraction towards a food source is formulated as:

$$\boldsymbol{F_i} = \boldsymbol{X^+} - \boldsymbol{X} \tag{12}$$

where $\boldsymbol{F_i}$ is the food source of the ith individual, $\boldsymbol{X^+}$ is the food source position.

Repulsion from an enemy is formulated as:

$$\boldsymbol{E_i} = \boldsymbol{X^-} + \boldsymbol{X} \tag{13}$$

where $\boldsymbol{E_i}$ is the enemy of the ith individual, $\boldsymbol{X^-}$ is the enemy position.

The velocity of artificial dragonflies can be simulated by considering step vector ($\boldsymbol{\Delta X}$) representing the direction of their movement, which is calculated by the following equation:

$$\boldsymbol{\Delta X^{t+1}} = (s\boldsymbol{S_i} + a\boldsymbol{A_i} + c\boldsymbol{C_i} + f\boldsymbol{F_i} + e\boldsymbol{E_i}) + \omega^t \boldsymbol{\Delta X^t} \tag{14}$$

where $\boldsymbol{\Delta X}$ is the step vector of an artificial dragonfly, t is the present iteration, s is the separation weight, a is the alignment weight, c is the cohesion weight, f is the food factor, e is the enemy factor. The inertia weight factor, ω^t, is given by:

$$\omega^t = \omega_{\max} - \frac{\omega_{\max} - \omega_{\min}}{Iter_{\max}} \times Iter \tag{15}$$

The position of the artificial dragonflies is another factor to be considered to simulate their movement, which is computed using:

$$\boldsymbol{X^{t+1}} = \boldsymbol{X^t} + \boldsymbol{\Delta X^{t+1}} \tag{16}$$

where $\boldsymbol{X}$ is the position of an artificial dragonfly.

In the case of no neighboring solutions, the artificial dragonflies need to employ a *Levy* flight, which is a random walk to improve the exploration phase. The position of dragonflies in this situation is given by:

$$\boldsymbol{X^{t+1}} = \boldsymbol{X^t} + \boldsymbol{Levy}(d) \times \boldsymbol{X^t} \tag{17}$$

where the following equation is used to calculate the *Levy* flight:

$$Levy(d) = 0.01 \times \frac{r_1 \times \sigma}{|r_2|^{\frac{1}{\beta}}} \tag{18}$$

where r_1, r_2 are two uniformly generated random number in [0,1], β is a constant which is equal to 1.5 in this work. The parameter σ is calculated using the following equation:

$$\sigma = \left(\frac{\Gamma(1+\beta) \times \sin\left(\frac{\pi\beta}{2}\right)}{\Gamma\left(\frac{1+\beta}{2}\right) \times \beta \times 2^{\left(\frac{\beta-1}{2}\right)}} \right)^{1/\beta} \tag{19}$$

where $\Gamma(x) = (x-1)!$

3.2. PSO

PSO is one of the well-known population-based evolutionary and swarm intelligence algorithms, and has been successfully applied to solve many problems in different fields [41–43]. Moreover, PSO has been effectively employed to be hybrid with many other optimization algorithms because of its simplicity and fast convergence speed [24,38,44]. PSO was originally proposed by Eberhart and Kennedy in 1995 by mimicking the concepts of bird flocking and fish schooling behaviors [45]. In PSO, each particle flies around a multi-dimensional search space and represents a possible solution in an optimization problem. Each particle comprises of a position X_i and a velocity V_i. The particles are initialized in the search space with random velocity and position values. In each iteration, the velocity of each particle is updated based on its personal best experience, $\boldsymbol{X^t}_{pbesti}$, and the best experience among the whole swarm, $\boldsymbol{X^t}_{gbest}$, found so far. Therefore, the velocity and position of each particle can be mathematically formulated as follows:

$$\boldsymbol{V}_i^{t+1} = \omega^t \times \boldsymbol{V}_i^t + C_1 \times rand_1 \times (\boldsymbol{X}_{pbest_i}^t - \boldsymbol{X}_i^t) + C_2 \times rand_2 \times (\boldsymbol{X}_{gbest}^t - \boldsymbol{X}_i^t) \tag{20}$$

$$\boldsymbol{X}_i^{t+1} = \boldsymbol{X}_i^t + \boldsymbol{V}_i^{t+1} \tag{21}$$

where $\boldsymbol{V_i}$ is the velocity of the ith particle, t is the number of iteration, ω^t is defined as in (15), C_1 and C_2 are acceleration coefficients, $rand_1$ and $rand_2$ are uniformly generated random numbers, $\boldsymbol{X_i}$ is the position of the ith particle, $\boldsymbol{X_{pbesti}}$ is the personal best position of the ith particle, $\boldsymbol{X_{gbest}}$ is the global best position among the whole swarm.

3.3. DA-PSO

DA-PSO is a recently developed hybrid metaheuristic algorithm motivated by combining the advantages of the DA and PSO algorithms [38]. PSO applies both personal and global best experiences of the particles to find the optimal solution, is consequently good at exploitation, and often converges on the optimal solution quickly. However, PSO is sometimes trapped in the local optima rather than the global because it converges too quickly on an optimal solution. Conversely, DA is good at exploration since it employs the Levy flight to increase the stochastic behavior in the searching process. However, DA takes too long time to converge on the optimal solution. The hybrid DA-PSO algorithm was proposed to overcome these problems by merging the good exploration of DA together with the good exploitation of PSO, and it has been proven to successfully solve a complicated optimization problem such as multi-objective optimal power-flow (MO-OPF) problems, which is evident in [38]. The idea of the DA-PSO algorithm is that in the exploration phase, DA is employed to initially explore the solution space to provide the global solution area, and the best position of DA is provided. In the exploitation phase, the PSO equations are calculated but the velocity equation of PSO, Equation (20), is modified by replacing the global best position by the provided best position found so far by DA. The PSO then finds a better optimal solution from this starting point. Thus, the modified version of PSO equations can be written as:

$$\boldsymbol{V}_i^{t+1} = \omega^t \times \boldsymbol{V}_i^t + C_1 \times rand_1 \times (\boldsymbol{X}_{pbest_i}^t - \boldsymbol{X}_i^t) + C_2 \times rand_2 \times (\boldsymbol{X}_{DA}^{t+1} - \boldsymbol{X}_i^t) \tag{22}$$

$$\boldsymbol{X}_i^{t+1} = \boldsymbol{X}_i^t + \boldsymbol{V}_i^{t+1} \tag{23}$$

4. An Improved DA-PSO Optimization Approach (iDA-PSO) for UC Problem

The iDA-PSO algorithm is proposed to solve the UC problem by improving the traditional DA-PSO algorithm. An approach for the improvement, the related computational formulations, and the application of the approach are explained below.

4.1. An Approach of Improving DA-PSO to Solve a Binary Problem

Although many efficient metaheuristic algorithms have been proposed in recent years, most of them cannot be applied to solve problems involving binary values such as the UC problem, which is the objective of this work. The contribution of this work is including binary values in the optimization thereby developing an efficient metaheuristic algorithm able to solve the UC problem. The hybrid metaheuristic algorithm DA-PSO operates only on real value; however, it was taken as the starting point to develop the improved DA-PSO (iDA-PSO) approach, which is proposed in this paper.

The Binary PSO (BPSO) was proposed by Kennedy and Eberhart by a modification of the traditional PSO to enable solving binary problems [46]. They also showed that the BPSO could successfully solve the test functions from [47]. In the BPSO, a particle is seen to move by flipping the number of bits. Consequently, the velocity of the particle can be represented by the change of probabilities of bit changed per iteration. In other words, a particle moves in a search space by only taking on values of 0 or 1, where each velocity (${V^t}_{i,gi}$) represents the probability of a bit of position (${X^t}_{i,gi}$) which takes the value 1. Since the position (${X^t}_{i,gi}$) and the personal best (${X^t}_{pbest,i,gi}$) are integers (0 or 1), and the velocity (${V^t}_{i,gi}$), which is a probability, needs to be limited to be in the range [0,1]. A function used to accomplish this is called the sigmoid function and is mathematically formulated as follows:

$$S(V_{i,gi}^t) = \frac{1}{1 + \exp(-V_{i,gi}^t)} \tag{24}$$

The sigmoid function limits the velocity within the appropriate range to be used as a probability. The change in position is defined by comparing with the random uniformly generated numbers between 0 and 1 which is formulated as follows:

$$\text{If } rand() < S(V_{i,gi}^t), \text{ then } X_{i,gi}^t = 1, \text{ else } X_{i,gi}^t = 0 \tag{25}$$

In the UC problem, V_{gimax} is set to limit the range of $V_{i,gi}$, so $S({V^t}_{i,gi})$ is not too close to 0 or 1. A higher value of V_{gimax} represents a lower frequency of changing the state of a generator.

To improve the DA-PSO algorithm to be able to solve the UC problem, the sigmoid function described above is applied in the process of the DA-PSO algorithm. The equation of updating the position of dragonflies, Equations (16) and (17), are both replaced by the sigmoid function, Equation (25). Similarly, the position equation of PSO, Equation (23), is also replaced by the sigmoid function equation to find the on/off status of each generator.

4.2. Priority List

A unit operating at its maximum power output normally has a lower cost per produced unit than that operating at other power output levels; hence, a unit should be operated at its maximum power output. Priority list, in this case, is based on the average full-load cost (α) of a unit that is defined as the cost per maximum power of a unit as the following:

$$\alpha_i = \frac{f_{Cost}(P_{gi\max})}{P_{gi\max}} = a_i P_{gi\max} + b_i + \frac{c_i}{P_{gi\max}} \tag{26}$$

where a unit with the least α_i is prioritized to be dispatched first.

4.3. Spinning Reserve Constraint Satisfaction

The unit scheduling from the heuristic search may not satisfy the spinning reserve constraint. There are two main ways to deal with the unsatisfying-constraint results. The first one is a penalty function, which transforms the constrained problem into an unconstrained one. However, when the problem is highly constrained, it may be hard to find the near global solution because of the reduction of the search space. The other is to repair the violations that have occurred, which approach used in this paper. The implementation of repairing the spinning reserve violation is expressed below:

Step 1. At each hour t, calculate α_i by using (26) for all uncommitted unit at hour t, and sort them in an ascending order.
Step 2. Calculate the spinning reserve requirement at t as in (5)
Step 3. If the result from step 2 satisfies the spinning reserve constraint, go to step 5; otherwise, go to step 4.
Step 4. Commit one uncommitted unit with the least α_i from step 1.
Step 5. If $t < T$, $t = t + 1$ and go to step 1; otherwise, stop this process.

4.4. Minimum Up-Time and Down-Time Constraints Satisfaction

The results obtained for unit scheduling from the previous process may violate the minimum up- and down-time constraints required in the UC problem. To repair the violations of these constraints, the following implementation is employed.

Step 1. At each hour t, calculate the accumulated current on/off hours of the ith unit at hour t, $T^t{}_{i,cur}$ by referring to the accumulated hours of the previous state, $T^t{}_{i,prev}$. If $t = 1$, $T^t{}_{i,\,prev}$ = initial state; else $T^t{}_{i,prev}$ = accumulated on/off hours of the previous state, $T^{t-1}{}_{i,cur}$.

Step 2. At each unit i

Step 2.1. If $u_i{}^t = 1$ and $T^t{}_{i,prev} \geq 1$, $T^t{}_{i,cur} = T^t{}_{i,prev} + 1$
Step 2.2. If $u_i{}^t = 1$ and $T^t{}_{i,prev} \leq -MDT_i$, $T^t{}_{i,cur} = 1$
Step 2.3. If $u_i{}^t = 0$ and $T^t{}_{i,prev} \leq -1$, $T^t{}_{i,cur} = T^t{}_{i,prev} - 1$
Step 2.4. If $u_i{}^t = 0$ and $T^t{}_{i,prev} \geq MUT_i$, $T^t{}_{i,cur} = -1$
Step 2.5. If $u_i{}^t = 0$ and $T^t{}_{i,prev} < MUT_i$, set $u_i{}^t = 1$ and $T^t{}_{i,cur} = T^t{}_{i,prev} + 1$
Step 2.6. If $u_i{}^t = 1$ and $T^t{}_{i,prev} > -MDT_i$, set $u_i{}^t = 0$ and $T^t{}_{i,cur} = T^t{}_{i,prev} - 1$

Step 3. If $i < N_g$, $i = i + 1$ and go to step 2; otherwise, go to step 4.

Step 4. If $t < T$, $t = t + 1$ and go to step 1; otherwise, stop this process.

4.5. Economic Dispatch

Repairing the minimum up- and/or down-time constraints may result in either excessive generation or spinning reserves, which leads to a high generation cost, or insufficient generation, which cannot meet the load demand and spinning reserve. In case of the excessive spinning reserve, the committed units with the minimum priority will be decommitted by simultaneously considering the minimum up- and down-time constraints and spinning reserve constraint until no unit can be decommitted. In other words, the minimum up- and down-time constraints and the spinning reserve constraints must be checked before decommitting a unit. Moreover, after decommitting a unit, the accumulated current on/off time, $T^t{}_{i,cur}$, must be updated according to the change of a unit. In the case of the insufficient generation, which cannot meet the load demand and spinning reserve, conversely, the uncommitted units with the highest priority will be committed without violating the minimum up- and down-time constraints until the generations from the committed units satisfy the spinning reserve constraints (i.e., Equation (5)). Similarly, after committing a unit, the accumulated current on/off hours, $T^t{}_{i,cur}$, must be updated according to the change of a unit. After updating the status of the units without any violations of the constraints, to solve the ED problem, the lambda-iteration method [1] is employed to find the optimal values of $P^t{}_{gi}$ of all committed units to meet the load demand while satisfying the power balance and generation limit constraints. The implementation of these processes can be explained as follows:

Step 1. At each hour t, check if $\sum_{i=1}^{Ng} P_{gi(\max)} u_i^t \geq P_D^t + P_R^t$, go to step 2; otherwise, go to step 8.

Step 2. Calculate α_i by using (26) for all committed unit at hour t, sort them in a descending order, and name it descending order list (DOL^t). Name the first unit in the DOL^t to be the lowest priority (LP^t).

Step 3. Compute the excessive spinning reserve by $ExcessReserve = \sum_{i=1}^{Ng} P_{gi(\max)} u_i^t - P_D^t - P_R^t$.

Step 4. Check if $ExcessReserve$ is higher than the maximum power output of the LP^t go to step 5; otherwise, go to step 6.

Step 5. Check if decommitting the LP^t does not violate its minimum up- or down-time constraint, decommit the LP^t, and update the $T^t{}_{i,cur}$.

Step 6. Delete the LP^t from the DOL^t.

Step 7. Check if the DOL^t is not empty, set the new LP^t to be the first unit of the DOL^t and go to step 3; otherwise, go to step 13.

Step 8. Calculate α_i by using (26) for all uncommitted units at hour t, sort them in an ascending order, and name it ascending order list (AOL^t). Name the first unit in the AOL^t to be the highest priority (HP^t).

Step 9. Compute the lacking spinning reserve by $LackReserve = \sum_{i=1}^{Ng} P_{gi(\max)} u_i^t - P_D^t - P_R^t$.

Step 10. Check if the $LackReserve < 0$, go to step 11; otherwise, go to step 13.

Step 11. Check if committing the HP^t does not violate its minimum up- or down-time constraint, commit the HP^t, and update the $T^t{}_{i,cur}$.

Step 12. Let the HP^t be the next unit in the AOL^t, and go to step 9.

Step 13. Solve the ED problem by a lambda-iteration method, which finds the optimal value of $P^t{}_{gi}$ of all on-line units to meet the load demand while satisfying the power balance and generation limit constraints.

Step 14. If $t < T$, $t = t + 1$ and go to step 1; otherwise, stop this process.

4.6. The Application of the iDA-PSO Approach for Solving the UC Problem

The application of the iDA-PSO approach for solving the UC problem is as follows:

Step 1. Produce the initial population of dragonflies and particles by randomly generating them to be on or off status (1 or 0) over the time horizon T.

Step 2. Calculate the objective function of each dragonfly, and set the best one to be the first personal best (X_{pbesti}) of PSO.

Step 3. Compute the coefficients used in DA (s, a, c, f, e and ω).

Step 4. Update the food source and enemy of DA.

Step 5. Compute the representative behavior factors of DA, namely $\boldsymbol{S}$, $\boldsymbol{A}$, $\boldsymbol{C}$, $\boldsymbol{F}$, and $\boldsymbol{E}$ by (9)–(13).

Step 6. If each dragonfly consists of at least one neighboring, update the step vector ($\boldsymbol{\Delta X}$) of a dragonfly by (14), and check whether any element of each population violates its limit, then move $\boldsymbol{\Delta X}$ of that population into its minimum/maximum limit. Then, update the position of dragonfly ($\boldsymbol{X_{DA}}$) by sigmoid function as in (25), as described in Section 4.1. However, if a dragonfly does not have any neighboring, calculate the *Levy* flight as in (18) and multiply it by $\boldsymbol{X_{DA}}$, then update $\boldsymbol{X_{DA}}$ by the sigmoid function, (25), and set $\boldsymbol{\Delta X}$ to be zero.

Step 7. Set the best position provided by DA to be the global best position of PSO ($\boldsymbol{X_{gbest}}$).

Step 8. Update the velocity of each particle ($\boldsymbol{V}$) by (22), and check whether any element of each population violates its limit, then move $\boldsymbol{V}$ of that population into its minimum/maximum limit. Then, apply the sigmoid function, Equation (25), to update the position of each particle ($\boldsymbol{X_{PSO}}$) as described in Section 4.1.

Step 9. Change the status of units of the newly generated population to satisfy the spinning reserve constraint as presented in Section 4.3.

Step 10. Repair the newly generated population violating the minimum up- or down-time constraint as explained in Section 4.4.

Step 11. Solve ED problem as expressed in Section 4.5 to find the optimal $\boldsymbol{P^t{}_{gi}}$ of all on-line units of the newly generated population.

Step 12. Calculate start-up costs, which are hot or cold starts, of the units started in each hour by comparing with the status of the previous hour. For the first hour, compare the status with that of the initial status of each unit.

Step 13. Calculate the objective function of the newly generated population.

Step 14. Test whether any obtained objective function from an individual is better than that of the previous X_{pbesti}, then the newly generated population is set to be a new X_{pbesti}. Likewise, if the best X_{pbesti} is better than X_{gbest}, that X_{pbesti} is set to be new X_{gbest}.

Step 15. If the maximum number of iterations is not reached, go to step 3; otherwise, stop the implementation and the optimal solution of UC problem is the particle with the non-dominated X_{gbest}.

The flowchart of the iDA-PSO approach for solving the UC problem is presented in Figure 1.

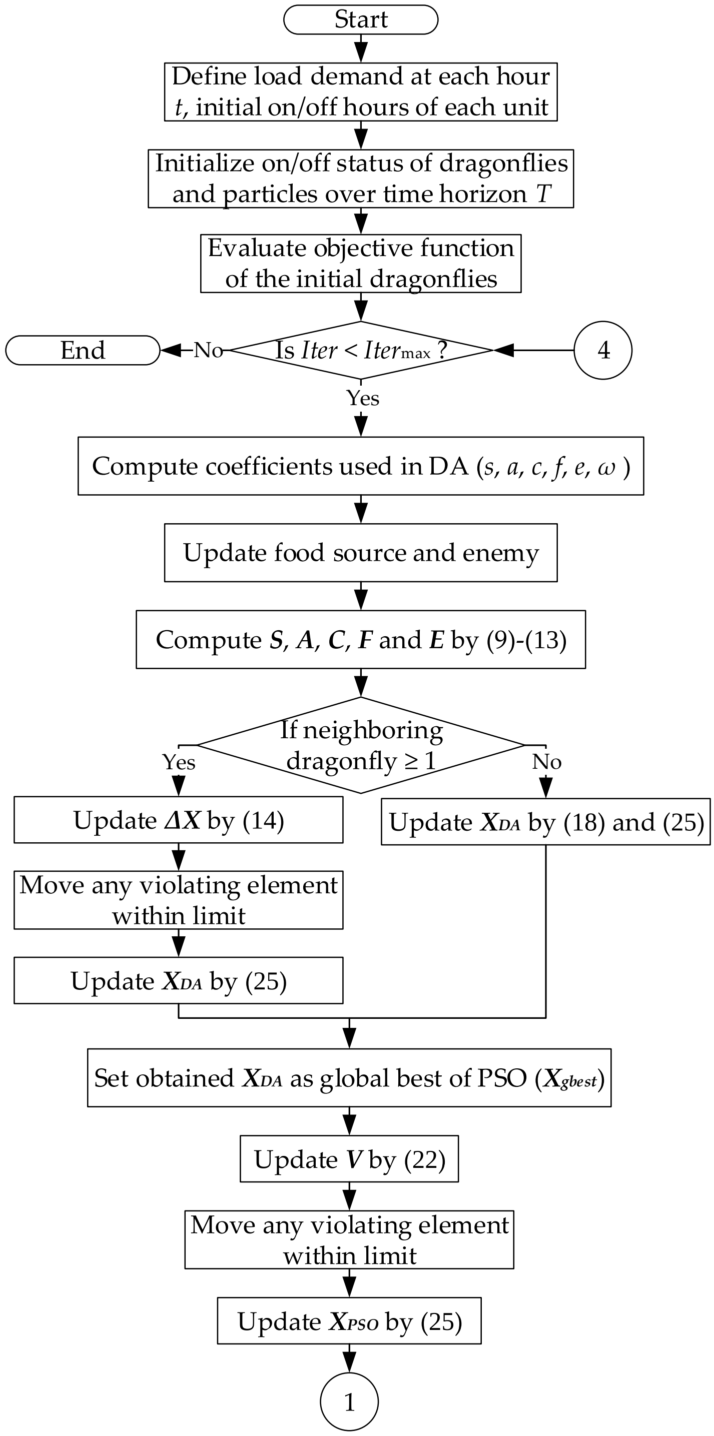

Figure 1. *Cont.*

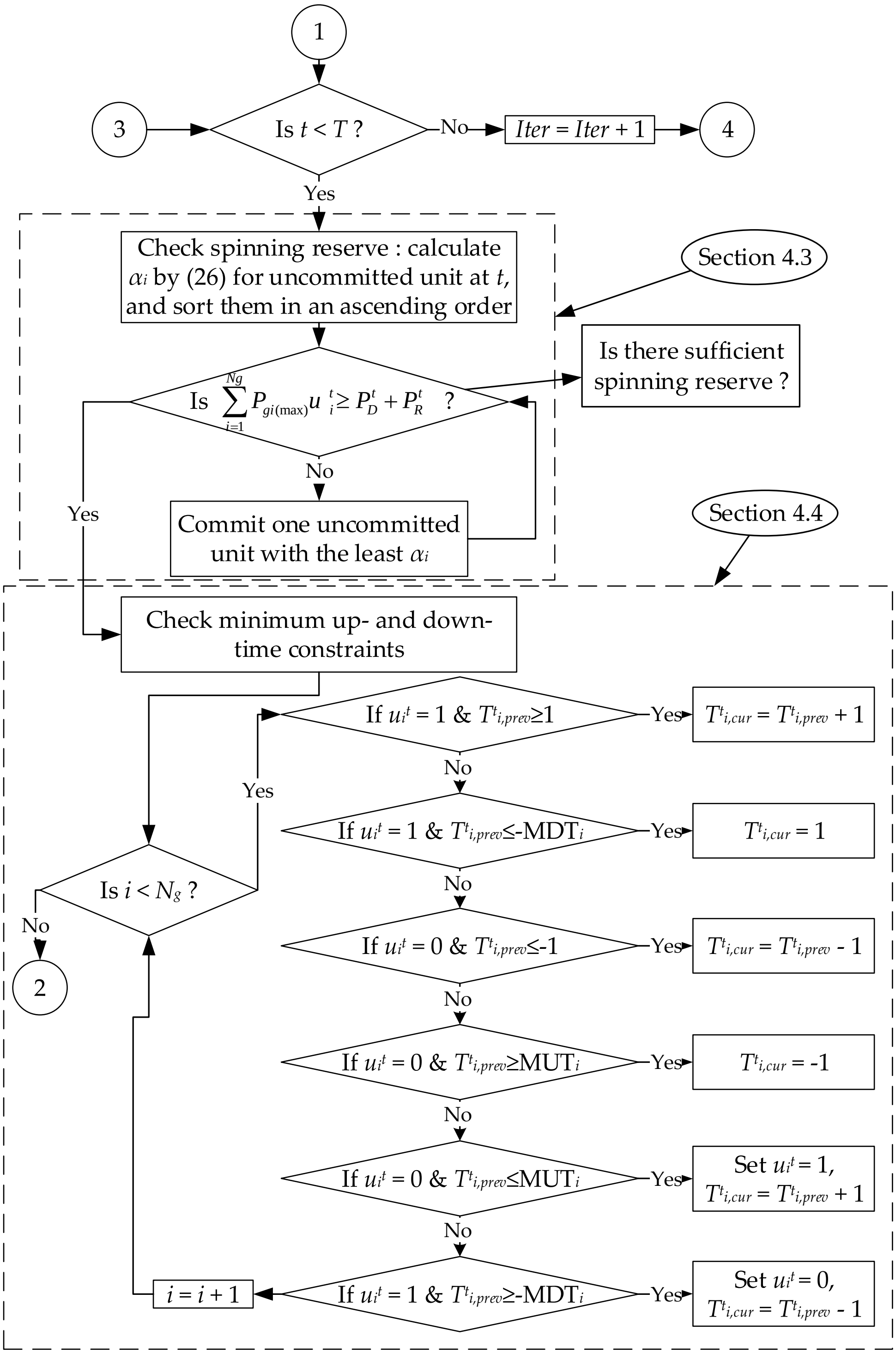

Figure 1. *Cont.*

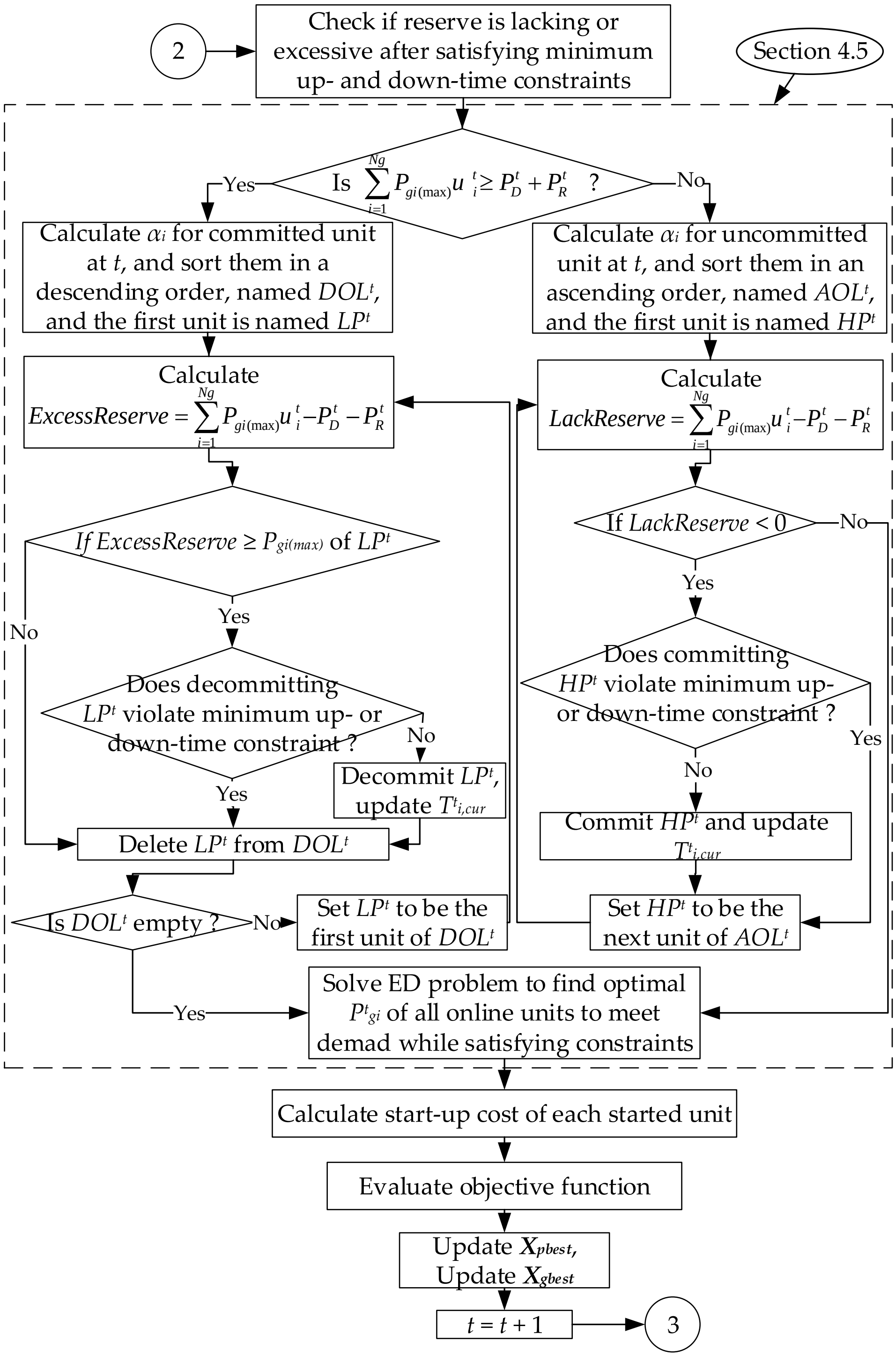

Figure 1. Flowchart of Improved Dragonfly Algorithm-Particle Swarm Optimization (iDA-PSO) approach for Unit Commitment (UC) problem.

5. Numerical Results

The effectiveness of the iDA-PSO algorithm is now examined by solving the UC problem using 24-hour scheduling time horizon for four systems of different sizes. The systems are the 5-unit system [48], 6-unit system [48], 10-unit system [16] and 26-unit system [49]. The spinning reserve requirement is equal to 10% of the total load demand of each hour in the 5-unit, 6-unit, and 10-unit systems. However, the spinning reserve requirement in the 26-unit system is equal to 5% of the total load demand of each hour as in [49]. The data of each system comprising of generator maximum and minimum limits, fuel cost coefficients, minimum up- and down-time limits, hot and cold start costs, cold start hours, and initial status of the units can be found in Tables 1–4. The 24-hour load demand for the 5-unit, 6-unit, 10-unit and 26-unit systems are provided in Tables 5–8, respectively. For each test system, the proposed approach operated for 30 independent runs, and the number of the population and maximum iteration number were set to be 100 and 200, respectively.

Table 1. System data for 5-unit system.

Unit No.	P_{gimax}	P_{gimin}	a ($/MW^2)	b ($/MW)	c ($/h)	MUT$_i$	MDT$_i$	HSC$_i$	CSC$_i$	CSH$_i$	IS$_i$
U1	250	10	0.00315	2	0	1	1	70	176	2	1
U2	140	20	0.0175	1.75	0	2	1	74	187	2	−3
U3	100	15	0.0625	1	0	1	1	50	113	1	−2
U4	120	10	0.00834	3.25	0	2	2	110	267	1	−3
U5	45	10	0.025	3	0	1	1	72	180	1	−2

Table 2. System data for 6-unit system.

Unit No.	P_{gimax}	P_{gimin}	a ($/MW^2)	b ($/MW)	c ($/h)	MUT$_i$	MDT$_i$	HSC$_i$	CSC$_i$	CSH$_i$	IS$_i$
U1	200	50	0.00375	2	0	1	1	70	176	2	1
U2	80	20	0.0175	1.7	0	2	2	74	187	1	−3
U3	50	15	0.0625	1	0	1	1	50	113	1	−2
U4	35	10	0.00834	3.25	0	1	2	110	267	1	−3
U5	30	10	0.025	3	0	2	1	72	180	1	−2
U6	40	12	0.025	3	0	1	1	40	113	1	−2

Table 3. System data for 10-unit system.

Unit No.	P_{gimax}	P_{gimin}	a ($/MW^2)	b ($/MW)	c ($/h)	MUT$_i$	MDT$_i$	HSC$_i$	CSC$_i$	CSH$_i$	IS$_i$
U1	455	150	0.00048	16.19	1000	8	8	4500	9000	5	8
U2	455	150	0.00031	17.26	970	8	8	5000	10000	5	8
U3	130	20	0.002	16.6	700	5	5	550	1100	4	−5
U4	130	20	0.00211	16.5	680	5	5	560	1120	4	−5
U5	162	25	0.0398	19.7	450	6	6	900	1800	4	−6
U6	80	20	0.00712	22.26	370	3	3	170	340	2	-3
U7	85	25	0.00079	27.74	480	3	3	260	520	2	−3
U8	55	10	0.00413	25.92	660	1	1	30	60	0	−1
U9	55	10	0.00222	27.27	665	1	1	30	60	0	−1
U10	55	10	0.00173	27.79	670	1	1	30	60	0	−1

Table 4. System data for 26-unit system.

Unit No.	P_{gimax}	P_{gimin}	a ($/MW^2)	b ($/MW)	c ($/h)	MUT$_i$	MDT$_i$	HSC$_i$	CSC$_i$	CSH$_i$	IS$_i$
U1	400	100	0.0019	7.5031	311.9102	8	5	500	500	10	10
U2	400	100	0.0019	7.4921	310.0021	8	5	500	500	10	10
U3	350	140	0.0015	10.8616	177.0575	8	5	300	200	8	10
U4	197	68.95	0.0026	23.2000	260.1760	5	4	200	200	8	−4
U5	197	68.95	0.0026	23.1000	259.6490	5	4	200	200	8	−4
U6	197	68.95	0.0026	23.0000	259.1310	5	4	200	200	8	−4
U7	155	54.25	0.0049	10.7583	143.5972	5	3	150	150	6	5
U8	155	54.25	0.0048	10.7367	134.3719	5	3	150	150	6	5
U9	155	54.25	0.0047	10.7154	143.0288	5	3	150	150	6	5
U10	155	54.25	0.0046	10.6940	142.7348	5	3	150	150	6	5
U11	100	25	0.0060	18.2000	218.7752	4	2	70	70	4	−3
U12	100	25	0.0061	18.1000	218.3350	4	2	70	70	4	−3
U13	100	25	0.0062	18.0000	217.8952	4	2	70	70	4	−3
U14	76	15.2	0.0093	13.4073	81.6259	3	2	50	50	3	3
U15	76	15.2	0.0091	13.3805	81.4641	3	2	50	50	3	3
U16	76	15.2	0.0089	13.3538	81.2980	3	2	50	50	3	3
U17	76	15.2	0.0088	13.3272	81.1364	3	0	50	50	3	3
U18	20	4	0.0143	37.8896	118.8206	0	0	20	20	2	−1
U19	20	4	0.0136	37.7770	118.4576	0	0	20	20	2	−1
U20	20	4	0.0126	37.6637	118.1083	0	0	20	20	2	−1
U21	20	4	0.0120	37.5510	117.7551	0	0	20	20	2	−1
U22	12	2.4	0.0285	26.0611	24.8882	0	0	0	0	1	−1
U23	12	2.4	0.0284	25.9318	24.7605	0	0	0	0	1	−1
U24	12	2.4	0.0280	25.8027	24.6382	0	0	0	0	1	−1
U25	12	2.4	0.0265	25.6753	24.4110	0	0	0	0	1	−1
U26	12	2.4	0.0253	25.5472	24.3891	0	0	0	0	1	−1

Table 5. 24-hour load demand for 5-unit system.

Hour	1	2	3	4	5	6	7	8	9	10	11	12
Demand	148	173	220	244	259	248	227	202	176	134	100	130
Hour	13	14	15	16	17	18	19	20	21	22	23	24
Demand	157	168	195	225	244	241	230	210	176	157	138	103

Table 6. 24-hour load demand for 6-unit system.

Hour	1	2	3	4	5	6	7	8	9	10	11	12
Demand	166	196	229	267	283.4	272	246	213	192	161	147	160
Hour	13	14	15	16	17	18	19	20	21	22	23	24
Demand	170	185	208	232	246	241	236	225	204	182	161	131

Table 7. 24-hour load demand for 10-unit system.

Hour	1	2	3	4	5	6	7	8	9	10	11	12
Demand	700	750	850	950	1000	1100	1150	1200	1300	1400	1450	1500
Hour	13	14	15	16	17	18	19	20	21	22	23	24
Demand	1400	1300	1200	1050	1000	1100	1200	1400	1300	1100	900	800

Table 8. 24-hour load demand for 26-unit system.

Hour	1	2	3	4	5	6	7	8	9	10	11	12
Dmd.	2223	2052	1938	1881	1824	1825.5	1881	1995	2280	2508	2565	2593.5
Hour	13	14	15	16	17	18	19	20	21	22	23	24
Dmd.	2565	2508	2479.5	2479.5	2593.5	2850	2821.5	2764.5	2679	2662	2479.5	2308.5

The simulation results of the proposed iDA-PSO approach for the 5-unit system are shown in Table 9, and the convergence curve is presented in Figure 2. The unit schedule and generation schedule for the 24-hour duration and the total generation cost are presented in this Table. The total generation cost through the scheduling duration obtained from the proposed iDA-PSO algorithm is equal to $11,830.94. The total generation cost provided by the iDA-PSO solution is better than that obtained by PSO-GWO, which is documented in the literature, for solving this UC problem. PSO-GWO achieved a generation cost of $12,281 [24].

Table 9. Commitment and generation schedule of the 5-unit system by Improve Dragonfly Algorithm-Particle Swarm Optimization (iDA-PSO) approach.

Hour	Unit Schedule					Generation Schedule				
	U1	U2	U3	U4	U5	U1	U2	U3	U4	U5
1	1	0	1	0	0	133	0	15	0	0
2	1	0	0	0	0	173	0	0	0	0
3	1	0	1	0	0	205	0	15	0	0
4	1	0	1	0	0	229	0	15	0	0
5	1	0	1	0	0	244	0	15	0	0
6	1	0	1	0	0	233	0	15	0	0
7	1	0	0	0	0	227	0	0	0	0
8	1	0	0	0	0	202	0	0	0	0
9	1	0	1	0	0	161	0	15	0	0
10	1	0	1	0	0	119	0	15	0	0
11	1	0	0	0	0	100	0	0	0	0
12	1	0	1	0	0	115	0	15	0	0
13	1	0	0	0	0	157	0	0	0	0
14	1	0	0	0	0	168	0	0	0	0
15	1	0	1	0	0	180	0	15	0	0
16	1	0	1	0	0	210	0	15	0	0
17	1	0	1	0	0	229	0	15	0	0
18	1	0	1	0	0	226	0	15	0	0
19	1	0	1	0	0	215	0	15	0	0
20	1	0	0	0	0	210	0	0	0	0
21	1	0	0	0	0	176	0	0	0	0
22	1	0	0	0	0	157	0	0	0	0
23	1	0	0	0	0	138	0	0	0	0
24	1	0	0	0	0	103	0	0	0	0
Total Cost ($) 11,830.94										

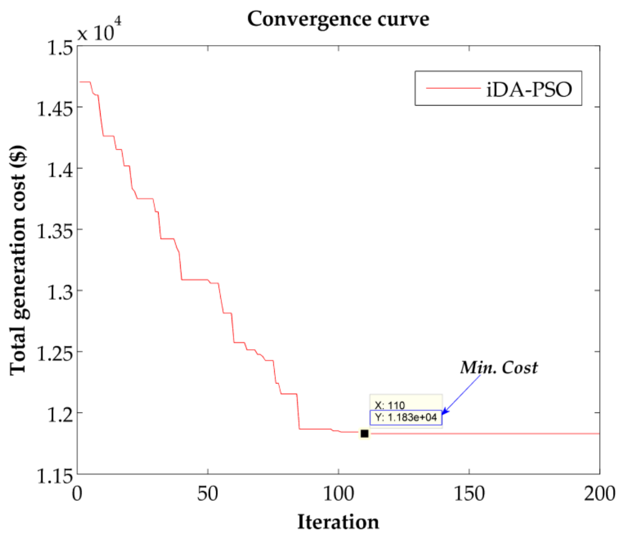

Figure 2. Convergence curve of the iDA-PSO approach for the 5-unit system.

Table 10 presents the unit schedule, generation schedule for the 24-hour duration and the total generation cost obtained by the proposed algorithm for the 6-unit system, and Figure 3 demonstrates the convergence curve of the algorithm for this system. The total generation cost of the proposed approach, which is $13,292.28, is once again better than that of the PSO-GWO, which is $13,600 [24].

Table 10. Commitment and generation schedule of the 6-unit system by iDA-PSO.

Hour	Unit Schedule						Generation Schedule					
	U1	U2	U3	U4	U5	U6	U1	U2	U3	U4	U5	U6
1	1	1	1	0	0	0	131	20	15	0	0	0
2	1	1	0	0	0	0	176	20	0	0	0	0
3	1	1	1	0	0	0	194	20	15	0	0	0
4	1	1	1	0	0	0	200	52	15	0	0	0
5	1	1	1	0	0	0	200	68.4	15	0	0	0
6	1	1	1	0	0	0	200	57	15	0	0	0
7	1	1	0	0	0	0	200	46	0	0	0	0
8	1	1	0	0	0	0	193	20	0	0	0	0
9	1	1	0	0	0	0	172	20	0	0	0	0
10	1	0	0	0	0	0	161	0	0	0	0	0
11	1	0	0	0	0	0	147	0	0	0	0	0
12	1	1	0	0	0	0	140	20	0	0	0	0
13	1	1	0	0	0	0	150	20	0	0	0	0
14	1	1	0	0	0	0	165	20	0	0	0	0
15	1	1	0	0	0	0	188	20	0	0	0	0
16	1	1	0	0	0	0	200	32	0	0	0	0
17	1	1	0	0	0	0	200	46	0	0	0	0
18	1	1	0	0	0	0	200	41	0	0	0	0
19	1	1	0	0	0	0	200	36	0	0	0	0
20	1	1	0	0	0	0	200	25	0	0	0	0
21	1	1	0	0	0	0	184	20	0	0	0	0
22	1	1	0	0	0	0	162	20	0	0	0	0
23	1	0	0	0	0	0	161	0	0	0	0	0
24	1	0	0	0	0	0	131	0	0	0	0	0
Total Cost ($) 13,292.28												

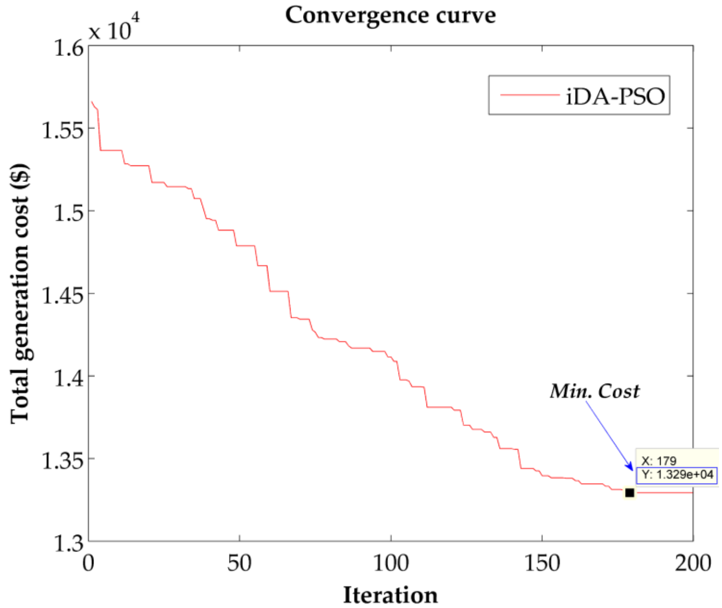

Figure 3. Convergence curve of the iDA-PSO approach for the 6-unit system.

For the 10-unit system, the simulation results including unit and generation schedule for the 24-hour duration and the total generation cost of the iDA-PSO approach are given in Table 11, and its convergence curve is provided in Figure 4. Through the scheduling duration, the total generation cost provided by the iDA-PSO is equal to $565,807.3094, which is slightly worse than those obtained by some algorithms in the literature. However, the total generation cost obtained by the iDA-PSO is significantly better than that of many algorithms presented in the literature. The algorithms GA [16], DP [16], LR [16], PSO-LR [17], EP [18], NGA [19], LCA-PSO [20], IPSO [20], MPSO [20], TSGA [21], ICGA [22], BCGA [22], SA [23], SM [23], PSO-GWO [24], HPSO [25], improve Lagrangian relaxation method (ILR) [25] and greedy randomized adaptive search procedure (GRASP) [50] are compared with the proposed approach as shown in Table 12. The best, average and worst generation costs and the computation times of the proposed iDA-PSO and other algorithms are also presented in Table 12. The computation time of the proposed iDA-PSO is slightly slower than those of some algorithms because of the sequential process of both DA and PSO.

Table 11. Commitment and generation schedule of the 10-unit system by iDA-PSO.

Unit Schedule										Generation Schedule									
U1	U2	U3	U4	U5	U6	U7	U8	U9	U10	U1	U2	U3	U4	U5	U6	U7	U8	U9	U10
1	1	0	0	0	0	0	0	0	0	455	245	0	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0	0	455	295	0	0	0	0	0	0	0	0
1	1	0	0	1	0	0	0	0	0	455	370	0	0	25	0	0	0	0	0
1	1	0	0	1	0	0	0	0	0	455	455	0	0	40	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	455	455	0	65	25	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	455	455	35	130	25	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	455	455	85	130	25	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	455	455	130	130	30	0	0	0	0	0
1	1	1	1	1	1	1	0	0	0	455	455	130	130	85	20	25	0	0	0
1	1	1	1	1	1	1	1	0	0	455	455	130	130	162	33	25	10	0	0
1	1	1	1	1	1	1	1	1	0	455	455	130	130	162	73	25	10	10	0
1	1	1	1	1	1	1	1	1	1	455	455	130	130	162	80	58	10	10	10
1	1	1	1	1	1	1	1	0	0	455	455	130	130	162	33	25	10	0	0
1	1	1	1	1	1	1	0	0	0	455	455	130	130	85	20	25	0	0	0
1	1	0	1	1	1	1	0	0	0	455	455	0	130	115	20	25	0	0	0
1	1	0	1	1	0	0	0	0	0	455	455	0	115	25	0	0	0	0	0
1	1	0	1	1	0	0	0	0	0	455	455	0	65	25	0	0	0	0	0
1	1	0	1	1	0	0	1	0	0	455	455	0	130	50	0	0	10	0	0
1	1	0	1	1	1	1	0	0	0	455	455	0	130	115	20	25	0	0	0
1	1	1	1	1	1	1	1	0	0	455	455	130	130	162	33	25	10	0	0
1	1	1	1	1	1	1	0	0	0	455	455	130	130	85	20	25	0	0	0
1	1	1	0	1	1	0	0	0	0	455	455	130	0	40	20	0	0	0	0
1	1	1	0	0	0	0	0	0	0	455	425	20	0	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	0	455	325	20	0	0	0	0	0	0	0
Total Cost ($) 565,807.3094																			

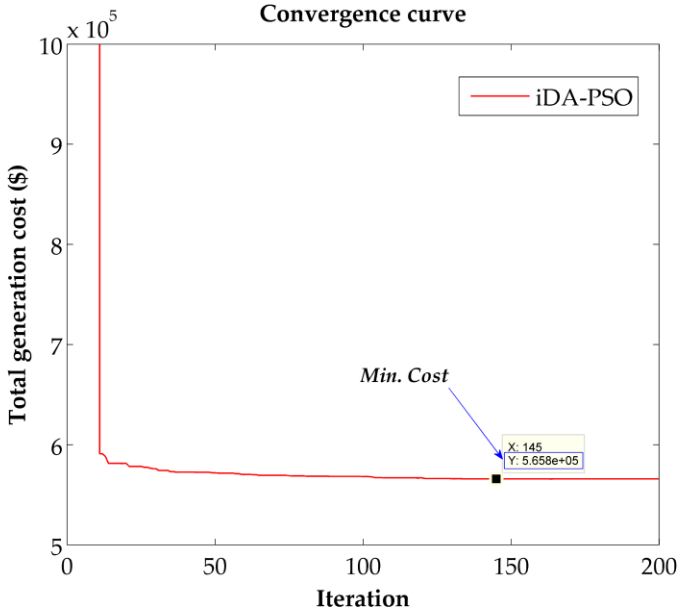

Figure 4. Convergence curve of the iDA-PSO approach for the 10-unit system.

Table 12. Simulation results of the iDA-PSO approach compared with other algorithms in the literature for the 10-generating unit system.

Methods	Total Generation Cost ($)			Time (s)
	Best	Average	Worst	
GA [16]	565,825	-	570,032	221
DP [16]	565,825	-	-	-
LR [16]	565,825	-	-	257
PSO-LR [17]	565,869	-	-	42
EP [18]	564,551	-	566,231	100
NGA [19]	591,715	-	-	677
LCA-PSO [20]	570,006	-	-	18.34
IPSO [20]	599,782	-	-	14.48
MPSO [20]	574,905	-	-	15.73
TSGA [21]	568,314.56	-	-	-
ICGA [22]	-	566,404	-	7.4
BCGA [22]	567,367	-	-	3.7
SA [23]	565,828	565,988	566,260	3.35
SM [23]	566,686	566,787	567,022	-
PSO-GWO [24]	565,210.2564	-	-	-
HPSO [25]	574,153	-	-	-
ILR [25]	565,823	-	-	-
GRASP [50]	565,825	-	-	17
iDA-PSO	565,807.3094	565,827.0145	565,891.7599	231.31

In the larger 26-unit system, the outcome of the UC for 24-hour duration together with the total generation cost provided by the proposed approach are shown by the non-zero numbers in Table 13, and Figure 5 displays the convergence curve of the proposed approach for this system. The total generation cost obtained by the iDA-PSO is equal to $741,587.7088 and is better than those of other algorithms, including GA [49], discrete binary particle swarm optimization (BPSO) [49], and modified particle swarm optimization (MPSO) [51] in the literature as presented in Table 14.

Table 13. Generation schedule of the 26-unit system by iDA-PSO.

Generation Schedule (Units 1–13)												
1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**
400	400	350	0	0	0	155	155	155	155	0	35.4	100
400	400	350	0	0	0	155	155	155	155	0	25	29
400	400	350	0	0	0	155	155	155	155	0	25	67
400	400	350	0	0	0	155	155	155	155	0	25	25
400	400	350	0	0	0	155	155	155	155	0	0	0
400	400	350	0	0	0	155	155	155	155	0	0	0
400	400	350	0	0	0	155	155	155	155	0	0	0
400	400	350	0	0	0	155	155	155	155	0	0	0
400	400	350	0	0	101.2	155	155	155	155	0	0	100
400	400	350	0	0	120.4	155	155	155	155	100	100	100
400	400	350	0	68.95	122.05	155	155	155	155	100	100	100
400	400	350	0	68.95	150.55	155	155	155	155	100	100	100
400	400	350	0	68.95	122.05	155	155	155	155	100	100	100
400	400	350	0	120.4	0	155	155	155	155	100	100	100
400	400	350	0	98.3	0	155	155	155	155	100	100	100
400	400	350	0	98.3	0	155	155	155	155	100	100	100
400	400	350	68.95	150.55	0	155	155	155	155	100	100	100
400	400	350	74.8	197	197	155	155	155	155	100	100	100
400	400	350	68.95	181.55	197	155	155	155	155	100	100	100
400	400	350	68.95	124.55	197	155	155	155	155	100	100	100
400	400	350	98.4	0	197	155	155	155	155	100	100	100
400	400	350	83.8	0	197	155	155	155	155	100	100	100
400	400	350	0	0	98.3	155	155	155	155	100	100	100
400	400	350	0	0	0	155	155	155	155	25	97.5	100
Generation Schedule (Units 14–26)												
14	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**
76	76	76	76	0	0	0	4	0	2.4	2.4	2.4	2.4
0	76	76	76	0	0	0	0	0	0	0	0	0
0	0	0	76	0	0	0	0	0	0	0	0	0
0	0	0	61	0	0	0	0	0	0	0	0	0
0	0	15.2	38.8	0	0	0	0	0	0	0	0	0
0	0	15.2	40.3	0	0	0	0	0	0	0	0	0
0	15.2	19.8	76	0	0	0	0	0	0	0	0	0
15.2	53	76	76	0	0	0	0	0	0	0	2.4	2.4
76	76	76	76	0	0	0	0	0	0	0	2.4	2.4
76	76	76	76	0	0	0	4	0	2.4	2.4	2.4	2.4
76	76	76	76	0	0	0	0	0	0	0	0	0
76	76	76	76	0	0	0	0	0	0	0	0	0
76	76	76	76	0	0	0	0	0	0	0	0	0
76	76	76	76	0	0	0	4	0	2.4	2.4	2.4	2.4
76	76	76	76	0	0	0	0	0	0	2.4	2.4	2.4
76	76	76	76	0	0	0	0	0	0	2.4	2.4	2.4
76	76	76	76	0	0	0	0	0	0	0	0	0
76	76	76	76	0	0	0	0	0	0	2.4	2.4	2.4
76	76	76	76	0	0	0	0	0	0	0	0	0
76	76	76	76	0	0	0	0	0	0	0	0	0
76	76	76	76	0	0	0	0	0	2.4	2.4	2.4	2.4
76	76	76	76	0	0	0	0	0	0	2.4	2.4	2.4
76	76	76	76	0	0	0	0	0	0	2.4	2.4	2.4
76	76	76	76	0	0	0	0	2.4	2.4	2.4	2.4	2.4
Total Cost ($) 741,587.7088												

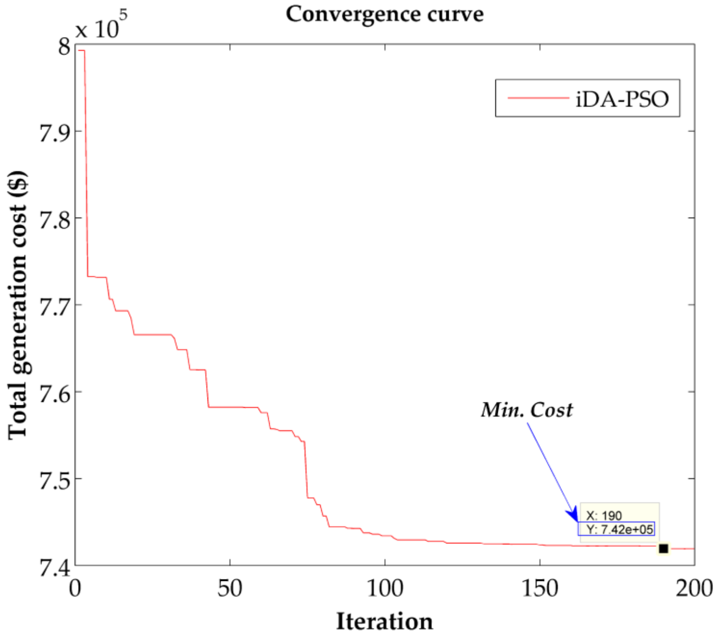

Figure 5. Convergence curve of the iDA-PSO approach for the 26-unit system.

Table 14. Simulation results of the iDA-PSO approach compared with other algorithms for the 26-generating unit system.

Methods	Total Generation Cost ($)			Time (s)
	Best	Average	Worst	
GA [49]	782,373	784,910	786,522	87.33
BPSO [49]	773,191	774,653	776,342	516.57
MPSO [51]	746,600.6	-	-	-
iDA-PSO	741,587.7088	743,176.1415	745,894.2814	327.76

From the generation schedule of each system, it can be noticed that the different units are dispatched in different ways. This is because the different units have different fuel cost coefficients, generation limits, minimum up- and down-time constraints, hot and cold and start-up costs and cold start hours, etc. Therefore, the units which have the cheapest fuel cost coefficient should be prioritized to be firstly dispatched, and the units which have the highest fuel cost coefficient should be dispatched only in the high-demand hour. However, these also depend on the start-up cost of each unit. Another noticeable point is most of the units keep a constant level of production over different time intervals. This is because when any unit has been turned off and turned on again, the start-up cost is added to the total generation cost causing a higher cost. Thus, if the units have low fuel cost coefficients and high maximum power generation, it is unnecessary to turn them off and on again.

According to all simulation results presented in Tables 9–14, the proposed approach can efficiently find the optimal unit schedule during 24-hour time horizon for four different system sizes. The total generation cost obtained by the proposed iDA-PSO approach is better than that of the recently proposed algorithm, PSO-GWO, for the 5- and 6-unit systems. For the 10-unit system, the iDA-PSO could provide considerably better total generation cost than many algorithms in the literature. The iDA-PSO could also produce considerably better total generation cost than several algorithms in the literature for the larger 26-unit system. Thus, adopting the sigmoid function to the recently proposed efficient optimization algorithm, DA-PSO, could make it able to solve the UC problem, which is a mixed-integer combinational optimization problem. The optimal on/off status of generating units, which is the mixed-integer part of the UC problem, could be efficiently provided for all studied systems, and the optimal total generation costs could also be obtained and are significantly better than that of many algorithms in the literature.

6. Conclusions

This paper has presented an improved DA-PSO algorithm that is capable of solving the UC problem in an electrical power system. The DA-PSO is a recent and efficient optimization algorithm, which has been proven to successfully solve a complicated optimization problem, which is a multi-objective, such as the OPF problem. However, DA-PSO cannot solve mixed-integer combination optimization problem such as the UC problem. To overcome this limitation, a new iDA-PSO algorithm has been proposed which employed the sigmoid function to enable finding the optimal on/off status of generation units, while satisfying the system constraints. The four test systems of different sizes (consisting of 5-unit, 6-unit, 10-unit and 26-unit systems) were used to demonstrate the effectiveness of the iDA-PSO algorithm. The proposed approach proved reliable by could successfully finding the optimal results for the generation schedule for a 24-hour duration for the test systems. The total generation costs over the scheduled time horizon obtained by iDA-PSO are less than those of many algorithms reported in the literature. Thus, applying the sigmoid function to the DA-PSO algorithm could enable it to solve the UC problem, which is a mixed-integer combinational problem, and the iDA-PSO also has a superiority over many algorithms reported in the literature. In the future work, the iDA-PSO approach could be improved and tested against other hybrid metaheuristic approaches such as fuzzy adaptive PSO.

Author Contributions: Conceptualization, S.K. and N.R.W.; Methodology, S.K.; Software, S.K.; Validation, S.K., N.R.W., A.S., and R.C.; Formal Analysis, S.K. and N.R.W.; Investigation, S.K.; Resources, N.R.W.; Data Curation, S.R.; Writing-Original Draft Preparation, S.R.; Writing-Review and Editing, S.K., N.R.W., A.S., and S.P.; Visualization, S.K.; Supervision, N.R.W., and A.S.; Project Administration, N.R.W., and A.S.; Funding Acquisition, A.S.

Funding: This research was funded by the Thailand Research Fund through the Royal Golden Jubilee Ph.D. Program (Grant no. PHD/0192/2557) to Mr Sirote Khunkitti and Professor Dr Apirat Siritaratiwat.

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

References

1. Wood, A.J.; Wollenberg, B.F. *Power Generation, Operation, and Control*, 2nd ed.; Wiley: Hoboken, NJ, USA, 1996; ISBN 9780471790556.
2. Padhy, N.P. Unit commitment—A bibliographical survey. *IEEE Trans. Power Syst.* **2004**, *19*, 1196–1205. [CrossRef]
3. Dillon, T.S.; Edwin, K.W.; Kochs, H.-D.; Taud, R.J. Integer Programming Approach to the Problem of Optimal Unit Commitment with Probabilistic Reserve Determination. *IEEE Trans. Power Appar. Syst.* **1978**, *PAS-97*, 2154–2166. [CrossRef]
4. Garver, L.L. Power Generation Scheduling by Integer Programming—Development of Theory. *Trans. Am. Inst. Electr. Eng. Part III Power Appar. Syst.* **1962**, *81*, 730–734. [CrossRef]
5. Cohen, A.; Yoshimura, M. A Branch-and-Bound Algorithm for Unit Commitment. *IEEE Trans. Power Appar. Syst.* **1983**, *PAS-102*, 444–451. [CrossRef]
6. Snyder, W.L.; Powell, H.D.; Rayburn, J.C. Dynamic programming approach to unit commitment. *IEEE Trans. Power Syst.* **1987**, *2*, 339–348. [CrossRef]
7. Lowery, P.G. Generating Unit Commitment by Dynamic Programming. *IEEE Trans. Power Appar. Syst.* **1966**, *PAS-85*, 422–426. [CrossRef]
8. Pang, C.K.; Chen, H.C. Optimal short-term thermal unit commitment. *IEEE Trans. Power Appar. Syst.* **1976**, *95*, 1336–1346. [CrossRef]
9. Pang, C.K.; Sheble, G.B.; Albuyeh, F. Evaluation of dynamic programming based methods and multiple area representation for thermal unit commitments. *IEEE Trans. Power Appar. Syst.* **1981**, *PAS-100*, 1212–1218. [CrossRef]
10. Su, M.-C.; Hsu, Y.-Y. Fuzzy Dynamic Programming: An Application to Unit Commitment. *IEEE Trans. Power Syst.* **1991**, *6*, 1231–1237. [CrossRef]

11. Ouyang, Z.; Shahidehpour, S.M. An intelligent dynamic programming for unit commitment application. *IEEE Trans. Power Syst.* **1991**, *6*, 1203–1209. [CrossRef]
12. Muckstadt, J.A.; Wilson, R.C. An Application of Mixed-Integer Programming Duality to Scheduling Thermal Generating Systems. *IEEE Trans. Power Appar. Syst.* **1968**, *PAS-87*, 1968–1978. [CrossRef]
13. André, M.; Sandrin, P. A New Method for Unit Commitment at Electricité de France. *IEEE Power Eng. Rev.* **1983**, *5*, 38–39.
14. Zhuang, F.; Galiana, F.D. Towards a more rigorous and practical unit commitment by lagrangian relaxation. *IEEE Trans. Power Syst.* **1988**, *3*, 763–773. [CrossRef]
15. Sheble, G.B. Solution of the unit commitment problem by the method of unit periods. *IEEE Trans. Power Syst.* **1990**, *5*, 257–260. [CrossRef]
16. Kazarlis, S.A.; Bakirtzis, A.G.; Petridis, V. A genetic algorithm solution to the unit commitment problem. *IEEE Trans. Power Syst.* **1996**, *11*, 83–92. [CrossRef]
17. Balci, H.; Valenzuela, J. Scheduling electric power generators using particle swarm optimization combined with the Lagrangian relaxation method. *Int. J. Appl. Math. Comput. Sci.* **2004**, *14*, 411–422.
18. Juste, K.A.; Kita, H.; Tanaka, E.; Hasegawa, J. An evolutionary programming solution to the unit commitment problem. *IEEE Trans. Power Syst.* **1999**, *14*, 1452–1459. [CrossRef]
19. Ganguly, D.; Sarkar, V.; Pal, J. A new genetic approach for solving the unit commitment problem. In Proceedings of the 2004 International Conference on Power System Technology, Singapore, 21–24 November 2004; Volume 1, pp. 542–547.
20. Wang, B.; Li, Y.; Watada, J. Re-Scheduling the Unit Commitment Problem in Fuzzy Environment. In Proceedings of the 2011 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2011), Taipei, Taiwan, 27–30 June 2011; pp. 1090–1095.
21. Eldin, A.S.; El-sayed, M.A.H.; Youssef, H.K.M. A two-stage genetic based technique for the unit commitment optimization problem. In Proceedings of the 2008 12th International Middle-East Power System Conference, Aswan, Egypt, 12–15 March 2008; pp. 425–430. [CrossRef]
22. Damousis, I.G.; Bakirtzis, A.G.; Dokopoulos, P.S. A solution to the unit-commitment problem using integer-coded genetic algorithm. *IEEE Trans. Power Syst.* **2004**, *19*, 1165–1172. [CrossRef]
23. Simopoulos, D.N.; Kavatza, S.D.; Vournas, C.D. Unit Commitment by an Enhanced Simulated Annealing Algorithm. *IEEE Trans. Power Syst.* **2006**, *21*, 68–76. [CrossRef]
24. Kamboj, V.K. A novel hybrid PSO–GWO approach for unit commitment problem. *Neural Comput. Appl.* **2016**, *27*, 1643–1655. [CrossRef]
25. Sriyanyong, P.; Song, Y.H. Unit commitment using particle swarm optimization combined with Lagrange relaxation. In Proceedings of the 2005 IEEE Power Engineering Society General Meeting, San Francisco, CA, USA, 16 June 2005; Volume 3, pp. 2752–2759. [CrossRef]
26. Kanellos, F.D.; Anvari-Moghaddam, A.; Guerrero, J.M. A cost-effective and emission-aware power management system for ships with integrated full electric propulsion. *Electr. Power Syst. Res.* **2017**, *150*, 63–75. [CrossRef]
27. Vahedipour-Dahraie, M.; Anvari-Moghaddam, A.; Guerrero, J.M. Evaluation of reliability in risk-constrained scheduling of autonomous microgrids with demand response and renewable resources. *IET Renew. Power Gener.* **2018**, *12*, 657–667. [CrossRef]
28. Yaprakdal, F.; Baysal, M.; Anvari-Moghaddam, A. Optimal Operational Scheduling of Reconfigurable Microgrids in Presence of Renewable Energy Sources. *Energies* **2019**, *12*, 1858. [CrossRef]
29. Esmaeili, S.; Anvari-Moghaddam, A.; Jadid, S.; Guerrero, J.M. Optimal simultaneous day-ahead scheduling and hourly reconfiguration of distribution systems considering responsive loads. *Int. J. Electr. Power Energy Syst.* **2019**, *104*, 537–548. [CrossRef]
30. Vahedipour-Dahraei, M.; Najafi, H.R.; Anvari-Moghaddam, A.; Guerrero, J.M. Security-constrained unit commitment in AC microgrids considering stochastic price-based demand response and renewable generation. *Int. Trans. Electr. Energy Syst.* **2018**, *28*, e2596. [CrossRef]
31. Vahedipour-Dahraie, M.; Najafi, H.; Anvari-Moghaddam, A.; Guerrero, J. Study of the Effect of Time-Based Rate Demand Response Programs on Stochastic Day-Ahead Energy and Reserve Scheduling in Islanded Residential Microgrids. *Appl. Sci.* **2017**, *7*, 378. [CrossRef]
32. Moghaddam, A.A.; Seifi, A.; Niknam, T. Multi-operation management of a typical micro-grids using Particle Swarm Optimization: A comparative study. *Renew. Sustain. Energy Rev.* **2012**, *16*, 1268–1281. [CrossRef]

33. Moghaddam, A.A.; Seifi, A.; Niknam, T.; Alizadeh Pahlavani, M.R. Multi-objective operation management of a renewable MG (micro-grid) with back-up micro-turbine/fuel cell/battery hybrid power source. *Energy* **2011**, *36*, 6490–6507. [CrossRef]
34. 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]
35. Mirjalili, S. Dragonfly algorithm: A new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems. *Neural Comput. Appl.* **2016**, *27*, 1053–1073. [CrossRef]
36. Sliman, L.; Bouktir, T. Economic Power Dispatch of Power System with Pollution Control using Multiobjective Ant Colony Optimization. *Int. J. Comput. Intell. Res.* **2007**, *3*, 145–153. [CrossRef]
37. Rezaei Adaryani, M.; 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]
38. Khunkitti, S.; Siritaratiwat, A.; Premrudeepreechacharn, S.; Chatthaworn, R.; Watson, N. A Hybrid DA-PSO Optimization Algorithm for Multiobjective Optimal Power Flow Problems. *Energies* **2018**, *11*, 2270. [CrossRef]
39. Bashishtha, T.K. Nature Inspired Meta-heuristic dragonfly Algorithms for Solving Optimal Power Flow Problem. *Int. J. Electron. Electr. Comput. Syst.* **2016**, *5*, 111–120.
40. Reynolds, C.W. Flocks, herds and schools: A distributed behavioral model. *ACM SIGGRAPH Comput. Graph.* **1987**, *21*, 25–34. [CrossRef]
41. Hu, X.; Shi, Y.; Eberhart, R. Recent advances in particle swarm. In Proceedings of the 2004 Congress on Evolutionary Computation, Portland, OR, USA, 19–23 June 2004; Volume 1, pp. 90–97. [CrossRef]
42. Banks, A.; Vincent, J.; Anyakoha, C. A review of particle swarm optimization. Part I: Background and development. *Nat. Comput.* **2007**, *6*, 467–484. [CrossRef]
43. Niknam, T.; Narimani, M.R.; Aghaei, J.; Azizipanah-Abarghooee, R. Improved particle swarm optimisation for multi-objective optimal power flow considering the cost, loss, emission and voltage stability index. *IET Gener. Transm. Distrib.* **2012**, *6*, 515. [CrossRef]
44. Narimani, M.R.; Azizipanah-Abarghooee, R.; Zoghdar-Moghadam-Shahrekohne, B.; Gholami, K. A novel approach to multi-objective optimal power flow by a new hybrid optimization algorithm considering generator constraints and multi-fuel type. *Energy* **2013**, *49*, 119–136. [CrossRef]
45. Eberhart, R.; Kennedy, J. A New Optimizer Using Particle Swarm Theory. In Proceedings of the Sixth International Symposium on Micro Machine and Human Science, Nagoya, Japan, 4–6 October 1995; pp. 39–43.
46. Kennedy, J.; Eberhart, R.C. A discrete binary version of the particle swarm algorithm. In Proceedings of the 1997 IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation, Orlando, FL, USA, 12–15 October 1997; Volume 5, pp. 4104–4108. [CrossRef]
47. De Jong, K.A. *Analysis of the Behavior of a Class of Genetic Adaptive Systems*; University of Michigan: Ann Arbor, MI, USA, 1975.
48. Anita, J.M.; Raglend, I.J. Solution of Unit Commitment Problem Using Shuffled Frog Leaping Algorithm. In Proceedings of the 2012 International Conference on Computing, Electronics and Electrical Technologies (ICCEET), Kumaracoil, India, 21–22 March 2012; pp. 109–115.
49. Gaing, Z.-L. Discrete particle swarm optimization algorithm for unit commitment. In Proceedings of the 2003 IEEE Power Engineering Society General Meeting, Toronto, ON, Canada, 13–17 July 2003; Volume 1, pp. 418–424. [CrossRef]
50. Viana, A.; De Sousa, J.P.; Matos, M. Using GRASP to Solve the Unit Commitment Problem. *Ann. Oper. Res.* **2003**, *120*, 117–132. [CrossRef]
51. Moussa, A.M.; Gammal, M.E.; Attia, A.I. An Improved Particle Swarm Optimization Technique for Solving the Unit Commitment Problem. *Online J. Power Energy Eng.* **2011**, *2*, 217–222.