In this paper, a hybrid optimization algorithm is proposed to solve multiobjective optimal power flow problems (MO-OPF) in a power system. The hybrid algorithm, named DA-PSO, combines the frameworks of the dragonfly algorithm (DA) and particle swarm optimization (PSO) to find the optimized solutions for the power system. The hybrid algorithm adopts the exploration and exploitation phases of the DA and PSO algorithms, respectively, and was implemented to solve the MO-OPF problem. The objective functions of the OPF were minimization of fuel cost, emissions, and transmission losses. The standard IEEE 30-bus and 57-bus systems were employed to investigate the performance of the proposed algorithm. The simulation results were compared with those in the literature to show the superiority of the proposed algorithm over several other algorithms; however, the time computation of DA-PSO is slower than DA and PSO due to the sequential computation of DA and PSO.

Keywords: dragonfly algorithm; metaheuristic; optimal power flow; particle swarm optimization

1. Introduction

For the past few decades, the optimal power flow (OPF) problem has played an essential role in studying the economy terms of power systems [1,2]. The OPF problem is a nonlinear, nonconvex, large-scale, and static programming problem [3] that optimizes selected objective functions while satisfying a set of equality and inequality constraints. The power balance equations are the equality constraints, and the limits of state and control variables are the inequality constraints of the OPF problem. The state variables consist of slack bus active power generation, load bus voltages, reactive power generation, and apparent power flow. The control variables involve active power generation except at slack bus, generator bus voltages, tap ratios of transformers, and reactive powers of shunt compensation capacitors. In recent years, because of the rise in fuel cost, which increases generation cost, fuel cost has become the objective function to be optimized in the OPF problem. Moreover, due to the release of emissions from thermal power plants into the atmosphere, emissions are yet another concern for power system operation and planning [4]. At the same time, because the demand for electricity has outpaced the expansion of transmission capacity, the inadequate reactive power sources of power systems have increased losses in transmission lines. Thus, emissions and transmission losses must also be considered as part of the objective functions of the OPF problem.
To solve the OPF problem, several traditional optimization techniques, such as nonlinear programming [5], quadratic programming [6], and the interior point method [7], have been successfully applied. However, these algorithms’ nonlinear characteristics make them impractical to use in practical systems. The nonlinear characteristics may cause the obtained solutions to be trapped in local optima, and these algorithms require an enormous amount of computational effort and time. Therefore, many optimization methods need to be improved to overcome these shortcomings [8,9]. Recently, several population-based optimization algorithms, including the OPF problem, have been employed to solve a complex constrained optimization problem in the field of power systems. Some of the other proposed techniques include the genetic algorithm (GA) [10], tabu search (TS) [11], differential evolution (DE) [12], evolutionary programming (EP) [13], probabilistic optimal power-flow (P-OPF) [14], preventive security-constrained power flow optimization [15], ant colony optimization (ACO) [16], grey wolf optimizer (GWO) [17], artificial bee colony (ABC) [18], particle swarm optimization (PSO) [19], and the dragonfly algorithm (DA) [20]. Even with the successful optimization of single-objective population-based optimization techniques, minimizing only one objective function is not sufficient in the power system because there are many problems, such as fuel cost, emissions, and transmission losses, which also need to be minimized. Consequently, many objective functions should be considered because this is a multiobjective optimization problem. Since there are three independent objective functions in this study (i.e., fuel cost, emissions, and transmission losses), the number of incompatible optimal solutions between the objective functions is infinite, and these optimal solutions are called Pareto optimal solutions [21].

Several optimization algorithms have been proposed and applied to solve the multiobjective OPF (MO-OPF) problem by many researchers. One of these methods was carried out by converting the multiobjective problem into a single-objective problem and then solving the problem by using a single-objective optimizer. However, this method has some drawbacks, such as the limitation of the available choices, the need for weights for each objective, and the requirement of multiple optimizer runs. To overcome these weaknesses, many researchers have proposed multiobjective evolutionary algorithms, such as the improved strength Pareto evolutionary algorithm (ISPEA2) [22], hybrid modified particle swarm optimization-shuffle frog leaping algorithms (HMPSO-SFLA) [23], modified teaching–learning-based optimization (MTLBO) [24], GWO [17], DE [17], multiobjective modified imperialist competitive algorithm (MOMICA) [25], differential search algorithm (DSA) [26], modified shuffle frog leaping algorithm (MSFLA) [27], modified Gaussian bare-bones multiobjective imperialist competitive algorithm (MGBICA) [28], multiobjective harmony search (MOHS) [29], adaptive real coded geography-based optimization (ARCBBO) [30], multiobjective differential evolution algorithm (MO-DEA) [31], hybrid modified imperialist competitive algorithm and teaching–learning algorithm (MICA-TLA) [32], etc., to successfully solve the OPF problem. In the past few decades, various well-proposed multiobjective evolutionary algorithms have been successfully applied and improved in many applications; however, most of them have not been extensively investigated in the OPF problem. Moreover, improving the search performance of the multiobjective evolutionary algorithm for solving the OPF problem is also important. In this paper, a hybrid DA-PSO algorithm is proposed to deal with the MO-OPF problem. The concept of the hybrid algorithm is the combination of the exploration and exploitation phases of the DA and PSO algorithms, respectively. The performance of the proposed algorithm was evaluated on the standard IEEE 30-bus and IEEE 57-bus power systems. Three different objective functions—fuel cost, emissions, and transmission losses—were individually and simultaneously considered as parts of the objective function in the OPF problem. The obtained results were compared with other evolutionary algorithms and the traditional DA and PSO.

The rest of the article is classified into five sections as follows. Section 2 introduces the formulation and constraints of the multiobjective optimization. In Section 3, the traditional DA and PSO are explained, and Section 4 depicts the concept of the proposed algorithm. Section 5 presents the optimization results and the comparisons between the solutions from the proposed algorithm and the
solution from other algorithms based on IEEE 30-bus and IEEE 57-bus systems. Finally, in Section 6, the conclusions of the simulation results of the proposed algorithm are described.

2. Problem Formulation and Constraints for Multi Objective Optimization for OPF

Multi-objective optimization is a model that optimizes more than one objective function to find optimal control variables while simultaneously satisfying equality and inequality constraints. The compromised solutions, nondominated solutions, which have more than one optimal solution between each objective, are the optimal solutions referred to as the Pareto front. The multiobjective problem is mathematically formulated as follows:

\[
\begin{align*}
\min \ f &= \{ f_1(x, u), f_2(x, u), \ldots, f_{N_{obj}}(x, u) \} \\
\text{subject to} \quad g(x, u) &= 0 \\
\quad h(x, u) &\leq 0
\end{align*}
\]

where \( f \) is a vector of objective functions to be optimized, \( N_{obj} \) is the number of objective functions, \( g(x,u) \) are the equality constraints, and \( h(x,u) \) are the inequality constraints.

\( x \) is a vector of state variables including slack bus active power, load bus voltages, generator reactive powers, and apparent power flows, expressed as follows:

\[
x = [P_{g\text{slack}}, V_{L1}, \ldots, V_{LN_L}, Q_{g1}, \ldots, Q_{gN_{gen}}, S_{11}, \ldots, S_{NN_L}]
\]

where \( P_{g\text{slack}} \) is the active power generation at slack bus, \( V_{Li} \) is the load voltage at bus \( i \), \( N_L \) is number of load buses, \( Q_{gi} \) is the reactive power generation at bus \( i \), \( N_{gen} \) is the number of total generators, \( S_{ii} \) is the apparent power flow at branch \( i \), and \( N_L \) is the number of transmission lines.

\( u \) is a vector of control variables consisting of active power generations except at slack bus, generator bus voltages, transformer tap ratios, and reactive powers of shunt compensation capacitors, expressed as:

\[
u = [P_{g\in PVbus}, \ldots, P_{gN_{gen}}, V_{g1}, \ldots, V_{gN_{gen}}, T_1, \ldots, T_{N_{tran}}, Q_{c1}, \ldots, Q_{cN_{cap}}]
\]

where \( P_{g\in PVbus} \) is the active power generation at bus \( i \), \( PV_{bus} \) is the set of generator buses except at slack bus, \( V_{gi} \) is the generator bus voltage at bus \( i \), \( T_i \) is the transformer tap ratio at bus \( i \), \( N_{tran} \) is the number of transformer taps, \( Q_{ci} \) is the shunt compensation capacitor at bus \( i \), and \( N_{cap} \) is the number of compensation capacitors.

2.1. Objective Functions

In this study, the objective functions of the OPF, consisting of fuel cost, emissions, and transmission line losses, are considered as shown below.

2.1.1. Fuel Cost

The total fuel cost of the generators is considered to be minimized and is given as follows:

\[
f_C(x, u) = \sum_{i=1}^{N_{gen}} (a_iP_{gj}^2 + b_jP_{gj} + c_i)
\]

where \( f_C \) is the total fuel cost of generators function ($/h$), and \( a_i, b_i \) and \( c_i \) are the fuel cost coefficients of the \( i \)th generator units.
2.1.2. Emissions

The emissions function can be represented as the sum of all considered emission types, such as sulphur oxides (SO$_x$), nitrogen oxides (NO$_x$), thermal emission, etc. However, in the present study, two important emission types, NO$_x$ and SO$_x$, are taken into account, as expressed below:

$$f_E(x, u) = \sum_{i=1}^{N_{\text{gen}}} \left( \gamma_i P_{gi}^2 + \beta_i P_{gi} + \alpha_i + \xi_i \exp(\lambda_i P_{gi}) \right)$$ (7)

where $f_E$ is the total emission generations function (ton/h), and $\gamma_i$, $\beta_i$, $\alpha_i$, $\xi_i$ and $\lambda_i$ are emission coefficients of the $i^{th}$ generator units.

2.1.3. Transmission Line Losses

The system active power loss in the transmission line is formulated as follows:

$$f_L(x, u) = \sum_{k=1}^{N_L} g_k (V_i^2 + V_j^2 - 2V_i V_j \cos(\theta_{ij}))$$ (8)

where $f_L$ is the total transmission loss function (MW), $g_k$ is the conductance of the $k^{th}$ line, $V_i$ is the voltage at bus $i$, $V_j$ is the voltage at bus $j$, and $\theta_{ij}$ is the voltage phase angle difference between buses $i$ and $j$.

2.2. Constraints

2.2.1. Equality Constraints

The OPF equality constraints are the active and reactive power balance constraints, as follows:

$$P_{gi} - P_{di} = \sum_{j=1}^{N_{\text{bus}}} V_i V_j (G_{ij} \cos(\theta_{ij}) + B_{ij} \sin(\theta_{ij}))$$ (9)

$$Q_{gi} - Q_{di} = \sum_{j=1}^{N_{\text{bus}}} V_i V_j (G_{ij} \sin(\theta_{ij}) - B_{ij} \cos(\theta_{ij}))$$ (10)

where $P_{di}$ is the active power demand at bus $i$, $N_{\text{bus}}$ is the number of buses, $G_{ij}$ is the transfer conductance between buses $i$ and $j$, $B_{ij}$ is the transfer susceptance between buses $i$ and $j$, and $Q_{di}$ is the reactive power demand at bus $i$.

2.2.2. Inequality Constraints

$$P_{gimin} \leq P_{gi} \leq P_{gimax} \quad i = 1, 2, \ldots, N_{\text{gen}}$$ (11)

$$Q_{gimin} \leq Q_{gi} \leq Q_{gimax} \quad i = 1, 2, \ldots, N_{\text{gen}}$$ (12)

$$V_{gimin} \leq V_{gi} \leq V_{gimax} \quad i = 1, 2, \ldots, N_{\text{gen}}$$ (13)

$$|S_{li}| \leq S_{limax}$$ (14)

$$V_{Limin} \leq V_{li} \leq V_{Limax} \quad i = 1, 2, \ldots, N_L$$ (15)

$$Q_{cimin} \leq Q_{ci} \leq Q_{cimax} \quad i = 1, 2, \ldots, N_{\text{cap}}$$ (16)

$$T_{imin} \leq T_{i} \leq T_{imax} \quad i = 1, 2, \ldots, N_{\text{tran}}$$ (17)

where $P_{gimin}$ and $P_{gimax}$ are the minimum and maximum active power generations at bus $i$, respectively, $Q_{gimin}$ and $Q_{gimax}$ are the minimum and maximum reactive power generations at bus $i$, respectively,
$V_{g_{\text{min}}}$, $V_{g_{\text{max}}}$ are the minimum and maximum generator voltage at bus $i$, respectively, $S_{l_{\text{max}}}$ is the maximum apparent power flow at branch $i$, $V_{L_{\text{min}}}$, $V_{L_{\text{max}}}$ are the minimum and maximum load voltage at bus $i$, respectively, $Q_{c_{\text{min}}}$ and $Q_{c_{\text{max}}}$ are the minimum and maximum shunt compensation capacitor at bus $i$, respectively, $T_{i_{\text{min}}}$, $T_{i_{\text{max}}}$ are the minimum and maximum transformer tap-ratio at bus $i$, respectively.

2.2.3. Constraints Handling

The inequality of dependent variables, including slack bus active power generation, load bus voltage magnitudes, reactive power generations, and apparent power flows, are integrated into the penalized objective function to maintain these variables within their limits and to refuse infeasible solutions. The penalty function can be expressed as follows [27]:

$$J(x, u) = f(x, u) + K_p (P_{\text{slack}} - P_{\text{lim slack}})^2 + K_V \sum_{i=1}^{N_{\text{load}}} (V_{L_i} - V_{\text{lim } L_i})^2$$
$$+ K_Q \sum_{i=1}^{N_{\text{line}}} (Q_{g_i} - Q_{\text{lim } g_i})^2 + K_S \sum_{i=1}^{N_{\text{line}}} (S_{L_i} - S_{\text{lim } L_i})^2$$

where $J(x, u)$ is the penalized objective function, $K_p$, $K_Q$, $K_V$ and $K_S$ are the penalty factors, and $x_{\text{lim}}$ is the limit value of the dependent variables, determined as follows:

$$x_{\text{lim}} = \begin{cases} x_{\text{max}} & \text{if } x > x_{\text{max}} \\ x & \text{if } x_{\text{min}} < x < x_{\text{max}} \\ x_{\text{min}} & \text{if } x < x_{\text{min}} \end{cases}$$

3. Related Optimization Techniques

3.1. DA

DA is a metaheuristic algorithm which was inspired by the static and dynamic swarming behaviors of dragonflies in nature [33]. Dragonflies swarm for two goals: Hunting (static swarm) and migration (dynamic swarm). In the dynamic swarm, many dragonflies swarm when roaming over long distances and different areas, which is the purpose of the exploration phase. In the static swarm, dragonflies move in larger swarms and along one direction with local movements and sudden changes in the flying path, which is suitable in the exploitation phase.

The behavior of dragonflies can be represented through five principles, which are separation, alignment, cohesion, attraction to a food source, and distraction of an enemy. These five behaviors are described and calculated as follows:

Separation, which is the avoidance of the static crashing of individuals into other individuals in the neighborhood, is calculated by Equation (20).

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

where $S_i$ is the separation of the $i$th individual, $N$ is the number of neighboring individuals, $X$ is the position of the current individual, and $X_j$ is the position of $j$th neighboring individual.

Alignment, which refers to the velocity matching of individuals to the velocity of others in the neighborhood, is computed by Equation (21).

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

where $A_i$ is the alignment of the $i$th individual, and $V_j$ is the velocity of $j$th neighboring individual.
Cohesion, which is the propensity of individuals towards the center of mass of the neighborhood, is formulated by Equation (22).

\[
C_i = \sum_{j=1}^{N} X_j - X
\]  

(22)

where \(C_i\) is the cohesion of the \(i^{th}\) individual.

Attraction towards a food source computed by Equation (23), should be the main objective of any swarm to survive.

\[
F_i = X^+ - X
\]  

(23)

where \(F_i\) is the food source of the \(i^{th}\) individual, and \(X^+\) is the position of the food source.

Distraction of an enemy, which is computed by Equation (24), is another survival objective of the swarm.

\[
E_i = X^- + X
\]  

(24)

where \(E_i\) is the position of enemy of the \(i^{th}\) individual, and \(X^-\) is the position of the enemy source.

To simulate the movement of artificial dragonflies and update their positions, step vector (\(\Delta X\)) and position vector (\(X\)) are considered. The step vector represents the direction of the movement of the artificial dragonflies and is formulated as follows:

\[
\Delta X^{t+1} = (sS_i + aA_i + cC_i + fF_i + eE_i) + \omega t \Delta X^t
\]  

(25)

where \(\Delta X^{t+1}\) is the step vector at iteration \(t+1\), \(\Delta X^t\) is the step vector at iteration \(t\), \(s\), \(a\), \(c\), \(f\) and \(e\) are the separation weight, alignment weight, cohesion weight, food factor and enemy factor, respectively, and \(\omega t\) is the inertia weight factor at iteration \(t\) and is calculated by Equation (26).

\[
\omega t = \omega_{max} - \omega_{max} \frac{\omega_{min}}{\text{Iter}_{max}} \times \text{Iter}
\]  

(26)

where \(\omega_{max}\) and \(\omega_{min}\) are set to 0.9 and 0.4, respectively, \(\text{Iter}\) is the iteration, and \(\text{Iter}_{max}\) is the maximum iteration.

The position of the artificial dragonflies can be updated by the following equation:

\[
X^{t+1} = X^t + \Delta X^{t+1}
\]  

(27)

where \(X^{t+1}\) is the position at iteration \(t+1\), and \(X^t\) is the position at iteration \(t\).

When the search space does not have a neighboring solution, the artificial dragonflies need to move around the search space by applying random walk (Levy flight) to improve their stochastic behavior. So, in this case, the position of the dragonflies can be calculated by Equation (28).

\[
X^{t+1} = X^t + \text{Levy}(d) \times X^t
\]  

(28)

where \(d\) is the dimension of the position vectors, and the \(\text{Levy}\) is the Levy flight which is computed by Equation (29).

\[
\text{Levy}(d) = 0.01 \times \frac{r_1 \times \sigma}{|r_2|^\beta}
\]  

(29)

where \(r_1\) and \(r_2\) are two uniform random values in a range of \([0, 1]\), and \(\sigma\) is calculated by Equation (30).

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

(30)

where \(\beta\) is the constant (which is equal to 1.5 in this work), and \(\Gamma(x) = (x - 1)!\).
3.2. PSO

PSO is a population-based stochastic global optimization technique which was first introduced by Eberhart and Kennedy [34]. The idea of PSO came from the flocking behavior of birds or the schooling of fishes in their food hunting. In the PSO system, the population moves around a multidimensional search space where each particle represents a possible solution. Each particle contains the information of control variables and is associated with a fitness value that indicates its performance in the fitness space. Each particle $i$ consists of its position $X_i = (x_{i,1}, x_{i,2}, \ldots, x_{i,Nvar})$, where $Nvar$ represents the number of control variables, velocity $V_i = (v_{i,1}, v_{i,2}, \ldots, v_{i,Nvar})$ and personal best experience $X_{pbest_i} = (x_{pbest_{i,1}}, x_{pbest_{i,2}}, \ldots, x_{pbest_{i,Nvar}})$, and a swarm has a global best experience $X_{gbest} = (x_{gbest_1}, x_{gbest_2}, \ldots, x_{gbest_{Nvar}})$. During each iteration, each particle moves in the direction of its own personal best position provided so far as well as in the direction of the global best position obtained so far by particles in the swarm. The particles are operated according to the equations expressed as follows:

\[
V_i^{t+1} = \omega_i \times V_i^t + C_1 \times rand_1 \times (X_{pbest_i}^t - X_i^t) + C_2 \times rand_2 \times (X_{gbest}^t - X_i^t) \tag{31}
\]

\[
X_i^{t+1} = X_i^t + V_i^{t+1} \tag{32}
\]

where $V_i^{t+1}$ is the velocity of particle $i$ at iteration $t + 1$, $V_i^t$ is the velocity of particle $i$ at iteration $t$, $C_1$ and $C_2$ are two positive acceleration constants, $rand_1$ and $rand_2$ are two uniform random values in a range of $[0, 1]$, $X_{pbest_i}^t$ is the personal best position of particle $i$ at iteration $t$, $X_i^t$ is the position of particle $i$ at iteration $t$, $X_{gbest}^t$ is the global best position among all particles at iteration $t$, and $X_i^{t+1}$ is the position of particle $i$ at iteration $t + 1$.

4. Proposed Hybrid DA-PSO Optimization Algorithm for MO-OPF Problem

Many optimization algorithms have been proposed to overcome the optimization problem of being trapped in the local optima while the algorithms try to find the best solution. PSO has been proven in several works from the literature to find the optimal solution in various problems [35–38]. Because of its equations in finding the optimal solution by using the best experience of the particles, PSO could quickly converge on the optimal solution, i.e., it is good at exploitation. However, PSO is sometimes still trapped in the local optima because it converges on the optimal solution too quickly. In other words, PSO is poor at exploration, which is an important task of the optimization process. In DA, it applies the Levy flight to improve the randomness and stochastic behavior when there is no neighboring dragonfly. This could significantly improve the exploration process of the algorithm. However, the best experience, which is the personal best, of dragonflies is not applied during the operation. This causes the DA to converge on the optimal solution very slowly and can sometimes cause it to be trapped in the local optima. To overcome these problems, a new algorithm is proposed which combines the prominent points of the DA and PSO algorithms, which are the exploration of DA and the exploitation of PSO. At first, the dragonflies in DA are initialized to explore the search space to find the area of the global solution. Then, the best position of DA is obtained. The obtained best position from DA is then substituted as the global best position in the PSO equation (Equation (31)). After that, the PSO algorithm, which is the exploitation phase, operates by using the global best position from DA, allowing it to provide the expected optimal solution. The velocity and position equations of PSO can be modified as follows:

\[
V_i^{t+1} = \omega_i \times V_i^t + C_1 \times rand_1 \times (X_{pbest_i}^t - X_i^t) + C_2 \times rand_2 \times (X_{DA}^{t+1} - X_i^t) \tag{33}
\]

\[
X_i^{t+1} = X_i^t + V_i^{t+1} \tag{34}
\]

where $X_{DA}^{t+1}$ is the best position obtained from DA at iteration $t + 1$.

The application of the proposed DA-PSO algorithm for solving the MO-OPF problem can be described as follows:
Step 1. Clarify the system data comprising the fuel cost coefficients of the generators, emission coefficients of the generators, initial values of generator active powers, initial values of generator bus voltages, initial values of transformer tap ratios, initial values of shunt compensation capacitors, upper limit of $S_{li}$, lower and upper limits of $P_{gi}$, $Q_{gi}$, $V_{Li}$, $Q_{ci}$, and $T_{j}$, the parameters of DA and PSO, the number of dragonflies and particles, the number of iterations, and the archive size.

Step 2. Generate the initial population of dragonflies and particles.

Step 3. Convert the constrained multi objective problem to an unconstrained one by using Equation (18).

Step 4. Perform the power flow and calculate the objective functions for the initial population of dragonflies.

Step 5. Find the nondominated solutions and save them to the initial archive.

Step 6. Set the fitness value of the initial population as the food source.

Step 7. Calculate the parameters of DA ($s, a, c, f$, and $e$).

Step 8. Update the food source and enemy of DA.


Step 10. Check if a dragonfly has at least one neighboring dragonfly, then update step vector ($\Delta X$) and the position of dragonfly ($X_{DA}$) by Equations (25) and (27), respectively, and if each dragonfly has no neighboring dragonfly, then update $X_{DA}$ by Equation (28) and set $\Delta X$ to be zero.

Step 11. If any component of each population breaks its limit, then $\Delta X$ or $X_{DA}$ of that population is moved into its minimum/maximum limit.

Step 12. Set the best position obtained from DA as the global best of PSO ($X^{\text{gbest}}$).

Step 13. Update the velocity of the particle ($V$) and the position of the particle ($X_{PSO}$) by Equations (33) and (34), respectively.

Step 14. If any component of each population breaks its limit, then $V$ or $X_{PSO}$ of that population is moved into its minimum/maximum limit.

Step 15. Calculate the objective functions of the new produced population.

Step 16. Employ the Pareto front method to save the nondominated solutions to the archive and update the archive.

Step 17. If the maximum number of iterations is reached, the algorithm is stopped; otherwise, go to step 7.

The flowchart of the DA-PSO algorithm for the MO-OPF problem is shown in Figure 1.
Start

Initialize system data including upper 
and lower limits of $P_{gi}$, $Q_{gi}$, $V_{gi}$, $V_{li}$, $Q_{ci}$, $T_i$, 
Number of dragonfly and particle, 
Number of iteration, Archive size

Initialize dragonfly ($X_{DA}$, $\Delta X$), Initialize particle ($X_{PSO}$, $V$, $X^{pbest}$)

Covert the constrained problem to an unconstrained one by Equation (18)

Perform the power flow and evaluate fitness function of each dragonfly

Find nondominated solution and save to an initialized archive

Is $t < \text{Max iteration}$?

No

Yes

Calculate parameters of DA

$s$, $a$, $c$, $f$, $r$, $w$

Update food source and enemy

Calculate $S$, $A$, $C$, $F$, and $E$ by Equations (20) to (24)

Yes

If neighboring dragonfly $\geq 1$

No

Update $\Delta X$ by Equation (25), 
Update $X_{DA}$ by Equation (27)

Update $X_{PSO}$ by Equation (28)

Move any component breaking its limit into its minimum/maximum limit

Set $X_{DA}$ as global best of PSO ($X^{gbest}$)

Update $V$ by Equation (33), 
Update $X_{PSO}$ by Equation (34), 
Update $X^{pbest}$

Calculate objective function

Find nondominated solution and save to the archive by Pareto front method

$t = t + 1$

End

Figure 1. Flowchart of the dragonfly algorithm and particle swarm optimization (DA-PSO) algorithm for solving the multiobjective optimal power flow (MO-OPF) problem.
5. Simulation Results

To investigate the performance of the proposed algorithm, the IEEE 30-bus and IEEE 57-bus test systems were employed. The proposed algorithm operated for 30 independent runs for each test system. To validate the superiority of the proposed algorithm for solving the economic dispatch optimization problem, the results provided by the proposed algorithm were compared with those of other metaheuristic algorithms from the literature. In order to investigate both the single-objective optimization and multiobjective optimization, the simulation was divided into two cases. Single-objective optimization was evaluated in the first case. In the second case, multiobjective optimization to solve the MO-OPF by using the proposed DA-PSO algorithm was evaluated.

5.1. IEEE 30-Bus Test System

The proposed DA-PSO algorithm was applied to the IEEE 30-bus system to evaluate its performance. The IEEE 30-bus test system was composed of 6 generators at buses 1, 2, 5, 8, 11, and 13, 4 transformers between buses 6 and 9, buses 6 and 10, buses 4 and 12, and buses 27 and 28, and 41 transmission lines. The total system demand was 283.4 MW and 126.6 MVAR. The bus and branch data is given in [39]. The population number and the size of the Pareto archive were set to be 100 and 100, respectively.

5.1.1. Single-Objective OPF

To evaluate the performance of the proposed algorithm for solving the single-objective optimization, three different objective functions consisting of fuel cost, emissions, and transmission loss minimizations were individually considered as part of the objective function. The obtained results by the traditional DA, PSO, and the proposed DA-PSO algorithm for three individual objective functions are shown in Table 1. In Table 2, the best fuel cost of generators provided by the DA-PSO algorithm are compared with other algorithms in the literature, including TS [11], EP [13], ACO [16], SFLA [27], MSFLA [27], improved evolutionary programming (IEP) [40], modified differential evolution optimal power flow (MDE-OPF) [41], stochastic genetic algorithm (SGA) [42], evolutionary-programming-based optimal power flow (EP-OPF) [43], honey bee mating optimization (HBMO) [44], PSO, and DA. The comparison of the best emission values of the DA-PSO algorithm with various algorithms in the literature, including ACO [16], HMPSO-SFLA [23], TLBO [24], MTLBO [24], DSA [26], MSFLA [27], SFLA [27], GA [27], GBICA [28], improved particle swarm optimization (IPSO) [45], PSO, and DA, is shown in Table 3. In Table 4, the best transmission losses provided by DA-PSO algorithm are compared with other algorithms in the literature, including GWO [17], DE [17], MOHS [29], enhanced genetic algorithm with decoupled quadratic load flow (EGA-DQLF) [46], efficient evolutionary algorithm (EEA) [47], enhanced genetic algorithm (EGA) [47], PSO, and DA. It can be seen that the proposed DA-PSO provided better results compared with those of other algorithms for all three objective functions, which can be confirmed by the results in Tables 1–4. However, the computation time of the proposed DA-PSO is much slower than other algorithms in the literature because the proposed algorithm consumed the sequential computation time of DA and PSO as presented in Tables 2–4.

<table>
<thead>
<tr>
<th>Variables</th>
<th>Best Fuel Cost</th>
<th>Best Emission</th>
<th>Best Ploss</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_{g1}$ (MW)</td>
<td>176.2376</td>
<td>176.5128</td>
<td>176.1861</td>
</tr>
<tr>
<td>$P_{g2}$ (MW)</td>
<td>48.8432</td>
<td>48.6955</td>
<td>48.8318</td>
</tr>
<tr>
<td>$P_{g3}$ (MW)</td>
<td>21.5184</td>
<td>21.4431</td>
<td>21.5119</td>
</tr>
<tr>
<td>$P_{g4}$ (MW)</td>
<td>22.1257</td>
<td>22.0995</td>
<td>22.0737</td>
</tr>
<tr>
<td>$P_{g5}$ (MW)</td>
<td>12.2000</td>
<td>12.0673</td>
<td>12.2005</td>
</tr>
<tr>
<td>$P_{g6}$ (MW)</td>
<td>12.0000</td>
<td>12.0673</td>
<td>12.2005</td>
</tr>
<tr>
<td>$V_{b1}$ (p.u.)</td>
<td>1.0500</td>
<td>1.0500</td>
<td>1.0500</td>
</tr>
</tbody>
</table>
Table 3. Comparison of the results from DA-PSO and other algorithms when considering only fuel cost as part of the objective function for IEEE 30-bus system.

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>P2 (MW)</th>
<th>P3 (MW)</th>
<th>P4 (MW)</th>
<th>P5 (MW)</th>
<th>P6 (MW)</th>
<th>Emission (ton/h)</th>
<th>Loss ($/h)</th>
<th>Cost ($/h)</th>
<th>Time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>TS [11]</td>
<td>64.3720</td>
<td>72.1604</td>
<td>49.5438</td>
<td>32.9099</td>
<td>39.7390</td>
<td>945.8507</td>
<td>3.9368</td>
<td>0.221000</td>
<td>-</td>
</tr>
<tr>
<td>EP [13]</td>
<td>64.8148</td>
<td>68.0992</td>
<td>50.0000</td>
<td>34.9999</td>
<td>30.0000</td>
<td>948.3052</td>
<td>4.8439</td>
<td>0.202000</td>
<td>-</td>
</tr>
<tr>
<td>ACO [16]</td>
<td>64.2220</td>
<td>67.7485</td>
<td>49.9391</td>
<td>34.9894</td>
<td>30.0000</td>
<td>947.4392</td>
<td>3.9816</td>
<td>0.203000</td>
<td>-</td>
</tr>
<tr>
<td>MTLBO [18]</td>
<td>64.0275</td>
<td>67.5771</td>
<td>50.0000</td>
<td>35.0000</td>
<td>30.0000</td>
<td>944.4096</td>
<td>3.2347</td>
<td>0.208260</td>
<td>-</td>
</tr>
<tr>
<td>DSA [26]</td>
<td>65.7979</td>
<td>68.2688</td>
<td>50.0000</td>
<td>34.9999</td>
<td>39.9990</td>
<td>951.5106</td>
<td>5.6437</td>
<td>0.206000</td>
<td>-</td>
</tr>
<tr>
<td>MDE-OPF [41]</td>
<td>65.2570</td>
<td>67.8304</td>
<td>49.8797</td>
<td>35.0000</td>
<td>30.0000</td>
<td>944.4656</td>
<td>3.0967</td>
<td>0.204900</td>
<td>-</td>
</tr>
<tr>
<td>EP-OPF [43]</td>
<td>67.0320</td>
<td>68.4723</td>
<td>50.0000</td>
<td>35.0000</td>
<td>30.0000</td>
<td>944.4535</td>
<td>3.7920</td>
<td>0.205600</td>
<td>-</td>
</tr>
<tr>
<td>HIMO [42]</td>
<td>64.1678</td>
<td>67.6962</td>
<td>50.0000</td>
<td>35.0000</td>
<td>30.0000</td>
<td>944.4594</td>
<td>3.3964</td>
<td>0.204860</td>
<td>-</td>
</tr>
<tr>
<td>DA-PSO</td>
<td>64.0997</td>
<td>67.6295</td>
<td>50.0000</td>
<td>35.0000</td>
<td>30.0000</td>
<td>944.7193</td>
<td>3.3292</td>
<td>0.205853</td>
<td>290.01</td>
</tr>
</tbody>
</table>

Table 4. Comparison of the results from DA-PSO and other algorithms when considering only losses as part of the objective function for IEEE 30-bus system.

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>P2 (MW)</th>
<th>P3 (MW)</th>
<th>P4 (MW)</th>
<th>P5 (MW)</th>
<th>P6 (MW)</th>
<th>Cost ($/h)</th>
<th>Loss ($/h)</th>
<th>Emission (ton/h)</th>
<th>Loss ($/h)</th>
<th>Time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPU [17]</td>
<td>51.8100</td>
<td>80.0000</td>
<td>50.0000</td>
<td>35.0000</td>
<td>30.0000</td>
<td>968.3800</td>
<td>3.4100</td>
<td>0.207310</td>
<td>3.4100</td>
<td>15.90</td>
</tr>
<tr>
<td>DE [17]</td>
<td>51.8200</td>
<td>79.9900</td>
<td>49.9900</td>
<td>35.0000</td>
<td>29.9800</td>
<td>968.2300</td>
<td>3.207311</td>
<td>3.3800</td>
<td>16.50</td>
<td>-</td>
</tr>
<tr>
<td>GA [13]</td>
<td>66.2759</td>
<td>79.4134</td>
<td>46.8855</td>
<td>34.8880</td>
<td>29.1213</td>
<td>928.5099</td>
<td>2.122840</td>
<td>3.5165</td>
<td>5.15</td>
<td>-</td>
</tr>
<tr>
<td>EDA [14]</td>
<td>51.6098</td>
<td>80.0000</td>
<td>50.0000</td>
<td>35.0000</td>
<td>30.0000</td>
<td>967.4600</td>
<td>2.072851</td>
<td>3.2088</td>
<td>3.2088</td>
<td>-</td>
</tr>
<tr>
<td>DGA [47]</td>
<td>59.3216</td>
<td>74.8312</td>
<td>49.8547</td>
<td>34.9048</td>
<td>28.1099</td>
<td>952.3785</td>
<td>2.067358</td>
<td>3.2823</td>
<td>5.72</td>
<td>-</td>
</tr>
<tr>
<td>EP [13]</td>
<td>51.6740</td>
<td>79.9700</td>
<td>50.0000</td>
<td>35.0000</td>
<td>30.0000</td>
<td>967.9300</td>
<td>2.072727</td>
<td>3.2440</td>
<td>29.71</td>
<td>-</td>
</tr>
<tr>
<td>PSO [16]</td>
<td>51.9415</td>
<td>80.0000</td>
<td>50.0000</td>
<td>35.0000</td>
<td>30.0000</td>
<td>967.8696</td>
<td>2.072790</td>
<td>3.1941</td>
<td>102.81</td>
<td>-</td>
</tr>
<tr>
<td>DA-PSO</td>
<td>51.5893</td>
<td>80.0000</td>
<td>50.0000</td>
<td>35.0000</td>
<td>30.0000</td>
<td>967.8756</td>
<td>2.072797</td>
<td>3.1893</td>
<td>292.33</td>
<td>-</td>
</tr>
</tbody>
</table>
5.1.2. MO-OPF

In this subsection, the proposed algorithm is investigated as a multiobjective optimization problem, while every two and three objective functions are optimized simultaneously. The best two-dimensional Pareto fronts obtained from the DA, PSO, and DA-PSO algorithms for the IEEE 30-bus system are shown in Figures 2–4. However, DA could not provide the convergent Pareto front when simultaneously considering the emissions and losses as parts of the objective function. This shows that DA is suitable for some objective functions, but that it is not suitable for every objective function for finding optimal solutions. In Figure 5, the Pareto front provided by the DA-PSO algorithm for the three-dimensional Pareto front is shown. For all figures in this system, most of the nondominated solutions obtained by the DA-PSO algorithm are better than those from the DA and PSO algorithms. For instance, at the same level of the fuel cost, the emissions provided by DA-PSO are less than those of DA and PSO. This shows that the new proposed hybrid DA-PSO algorithm, which adopts the exploration phase of the DA and the exploitation phase of the PSO, could improve the performance of the original DA and PSO algorithms.

Figure 2. Two-dimensional Pareto fronts when considering fuel cost and emissions as part of the objective function for the IEEE 30-bus system.

Figure 3. Two-dimensional Pareto fronts when considering fuel cost and transmission losses as part of the objective function for the IEEE 30-bus system.
simultaneously considering the emissions and losses as parts of the objective function. This shows that DA is suitable for some objective functions, but that it is not suitable for every objective function for finding optimal solutions. In Figure 5, the Pareto front provided by the DA-PSO algorithm for the three-dimensional Pareto front is shown. For all figures in this system, most of the nondominated solutions obtained by the DA-PSO algorithm are better than those from the DA and PSO algorithms. For instance, at the same level of the fuel cost, the emissions provided by DA-PSO are less than those of DA and PSO. This shows that the new proposed hybrid DA-PSO algorithm, which adopts the exploration phase of the DA and the exploitation phase of the PSO, could improve the performance of the original DA and PSO algorithms.

5.2. IEEE 57-Bus Test System

The proposed hybrid DA-PSO was also tested on the IEEE 57-bus system to investigate its performance. The system active and reactive power demands were 1250.8 MW and 336.4 MVAR, respectively. It consisted of 7 generators located at buses 1, 2, 3, 6, 8, 9, and 12, 15 transformers, and...
80 transmission lines. The detail data were taken from [48]. The population number was 100 and the size of the Pareto archive was 100.

5.2.1. Single-Objective OPF

To verify its performance for solving the single-objective OPF in a larger system, the proposed algorithm was also applied to the IEEE 57-bus test system. Three different objective functions, i.e., fuel cost, emissions, and transmission losses, were individually considered as part of the objective function. The results provided by DA, PSO, and the proposed DA-PSO algorithm for the three individual objectives are shown in Table 5. The best results from DA-PSO are compared with those of: MTLB [23], DSA [25], GBICA [27], MGBKCA [27], ARCBBO [29], MO-DEA [30], MICA-TLA [31], TLBO [48], Levy mutation teaching–learning-based optimization (LTLBO) [49], new particle swarm optimization (NPSO) [50], fuzzy genetic algorithm (Fuzzy-GA) [51], differential evolution pattern search (DE-PS) [52], ABC [53], particle swarm optimization algorithm with linearly decreasing inertia weight (LDI-PSO) [53], evolving ant direction differential evolution (EADDE) [54], gravitational search algorithm (GSA) [55], adaptive particle swarm optimization strategy (APSO) [56], PSO, and DA for the fuel cost objective function; GBICA [27], MGBKCA [27], PSO, and DA for the emission objective function; and PSO and DA for the transmission loss objective function—all of which is summarized in Tables 5–8. From these tables, it is obvious that the proposed algorithm could provide more optimized results than the compared algorithms for all three objective functions.

**Table 5.** Comparison of the simulation results from PSO, DA, and DA-PSO for IEEE 57-bus system.

<table>
<thead>
<tr>
<th>Variables</th>
<th>Best Fuel Cost</th>
<th>Best Emission</th>
<th>Best ( P_{\text{loss}} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( P_{\text{c}} ) (MW)</td>
<td>142.7472</td>
<td>154.8513</td>
<td>141.4617</td>
</tr>
<tr>
<td>( P_{\text{f}} ) (MW)</td>
<td>8.8427</td>
<td>76.6227</td>
<td>67.7806</td>
</tr>
<tr>
<td>( Q_{\text{c}} ) (MW)</td>
<td>44.9025</td>
<td>49.4440</td>
<td>44.6638</td>
</tr>
<tr>
<td>( T_{\text{f}} ) (MW)</td>
<td>70.8490</td>
<td>100.0000</td>
<td>73.6254</td>
</tr>
<tr>
<td>( P_{\text{p}} ) (MW)</td>
<td>458.6003</td>
<td>438.7375</td>
<td>438.9904</td>
</tr>
<tr>
<td>( P_{\text{a}} ) (MW)</td>
<td>100.0000</td>
<td>100.0000</td>
<td>97.4953</td>
</tr>
<tr>
<td>( V_{\text{c}} ) (p.u.)</td>
<td>1.0590</td>
<td>1.0590</td>
<td>1.0590</td>
</tr>
<tr>
<td>( V_{\text{f}} ) (p.u.)</td>
<td>1.0792</td>
<td>1.0792</td>
<td>1.0792</td>
</tr>
<tr>
<td>( V_{\text{d}} ) (p.u.)</td>
<td>1.0455</td>
<td>1.0455</td>
<td>1.0455</td>
</tr>
<tr>
<td>( P_{\text{d}} ) (MW)</td>
<td>0.9249</td>
<td>1.1000</td>
<td>1.0221</td>
</tr>
<tr>
<td>( T_{\text{d}} ) (MW)</td>
<td>0.9901</td>
<td>1.0253</td>
<td>0.9983</td>
</tr>
<tr>
<td>( T_{\text{a}} ) (MW)</td>
<td>0.9277</td>
<td>1.0731</td>
<td>0.9582</td>
</tr>
<tr>
<td>( T_{\text{t}} ) (MW)</td>
<td>0.9000</td>
<td>0.9879</td>
<td>0.9636</td>
</tr>
<tr>
<td>( T_{\text{l}} ) (MW)</td>
<td>0.9247</td>
<td>1.0731</td>
<td>0.9582</td>
</tr>
<tr>
<td>( T_{\text{r}} ) (MW)</td>
<td>1.0151</td>
<td>1.1000</td>
<td>1.0196</td>
</tr>
<tr>
<td>( E_{\text{f}} ) (MW)</td>
<td>1.0494</td>
<td>1.0494</td>
<td>1.0494</td>
</tr>
<tr>
<td>( E_{\text{d}} ) (MW)</td>
<td>1.0479</td>
<td>1.0479</td>
<td>1.0479</td>
</tr>
<tr>
<td>( E_{\text{a}} ) (MW)</td>
<td>1.0628</td>
<td>1.0628</td>
<td>1.0628</td>
</tr>
<tr>
<td>( E_{\text{t}} ) (MW)</td>
<td>1.0464</td>
<td>1.0464</td>
<td>1.0464</td>
</tr>
<tr>
<td>( E_{\text{l}} ) (MW)</td>
<td>1.0410</td>
<td>1.0410</td>
<td>1.0410</td>
</tr>
<tr>
<td>( E_{\text{r}} ) (MW)</td>
<td>0.9249</td>
<td>1.0000</td>
<td>1.0000</td>
</tr>
<tr>
<td>( E_{\text{f}} ) (MW)</td>
<td>0.9901</td>
<td>1.0253</td>
<td>0.9983</td>
</tr>
<tr>
<td>( E_{\text{d}} ) (MW)</td>
<td>0.9277</td>
<td>1.0731</td>
<td>0.9582</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Fuel Cost ($/h)**

| 41,698.37 | 41,828.45 | 41,674.62 |

**Emission (ton/h)**

| 1.9027 | 1.6883 | 1.9087 |

**\( P_{\text{loss}} \) (MW)**

| 15.4903 | 16.5502 | 14.9380 |

**Energies 2018, 11, 2270**

14 of 21
Table 6. Comparison of the results from DA-PSO and other algorithms when considering only fuel cost as part of the objective function for the IEEE 57-bus system.

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>Cost ($/h)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MTLBO [23]</td>
<td>41,638.3822</td>
</tr>
<tr>
<td>DSA [25]</td>
<td>41,686.8200</td>
</tr>
<tr>
<td>GBICA [27]</td>
<td>41,740.2884</td>
</tr>
<tr>
<td>MGBICA [27]</td>
<td>41,715.7101</td>
</tr>
<tr>
<td>ARCBBO [29]</td>
<td>41,686.0000</td>
</tr>
<tr>
<td>MO-DEA [30]</td>
<td>41,683.0000</td>
</tr>
<tr>
<td>MICA-TLA [31]</td>
<td>41,675.0545</td>
</tr>
<tr>
<td>TLBO [48]</td>
<td>41,695.6629</td>
</tr>
<tr>
<td>LTLBO [49]</td>
<td>41,679.5451</td>
</tr>
<tr>
<td>NPSO [50]</td>
<td>41,699.5163</td>
</tr>
<tr>
<td>Fuzzy-GA [51]</td>
<td>41,716.2808</td>
</tr>
<tr>
<td>DE-PS [52]</td>
<td>41,685.2950</td>
</tr>
<tr>
<td>ABC [53]</td>
<td>41,693.9589</td>
</tr>
<tr>
<td>LDI-PSO [53]</td>
<td>41,815.5035</td>
</tr>
<tr>
<td>EADDE [54]</td>
<td>41,713.6200</td>
</tr>
<tr>
<td>GSA [55]</td>
<td>41,695.8717</td>
</tr>
<tr>
<td>APSO [56]</td>
<td>41,713.8868</td>
</tr>
<tr>
<td>PSO</td>
<td>41,698.3672</td>
</tr>
<tr>
<td>DA</td>
<td>41,828.4473</td>
</tr>
<tr>
<td>DA-PSO</td>
<td>41,674.6209</td>
</tr>
</tbody>
</table>

Table 7. Comparison of the results from DA-PSO and other algorithms when considering the only emissions as part of the objective function for the IEEE 57-bus system.

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>Emission (ton/h)</th>
</tr>
</thead>
<tbody>
<tr>
<td>GBICA [27]</td>
<td>1.1881</td>
</tr>
<tr>
<td>MGBICA [27]</td>
<td>1.1724</td>
</tr>
<tr>
<td>PSO</td>
<td>1.0814</td>
</tr>
<tr>
<td>DA</td>
<td>1.3097</td>
</tr>
<tr>
<td>DA-PSO</td>
<td>1.0799</td>
</tr>
</tbody>
</table>

Table 8. Comparison of the results from DA-PSO and its traditional algorithms when considering only transmission losses as part of the objective function for the IEEE 57-bus system.

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>Loss (MW)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSO</td>
<td>10.7076</td>
</tr>
<tr>
<td>DA</td>
<td>13.6430</td>
</tr>
<tr>
<td>DA-PSO</td>
<td>10.1212</td>
</tr>
</tbody>
</table>

5.2.2. MO-OPF

This case proposes a multiobjective optimization problem by using the proposed DA-PSO algorithm to evaluate its performance for the IEEE 57-bus test system. The two-dimensional Pareto fronts provided by the PSO and DA-PSO algorithms for this system are shown in Figures 6–8, while DA could not provide the convergent Pareto fronts for any multiobjective functions in this system. In Figure 9, the three-dimensional Pareto front obtained from DA-PSO is shown. From all figures for this system, the fronts obtained from DA-PSO algorithm are superior to those from PSO, while the fronts obtained from DA could not converge because the best experience of dragonflies in DA is not applied during the operation and the obtained solutions are trapped in the local optima. From the results, it can be seen that the proposed hybrid DA-PSO performs better than the original DA and PSO algorithms once again.
Table 7. Comparison of the results from DA-PSO and other algorithms when considering the only emissions as part of the objective function for the IEEE 57-bus system.

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>Emission (ton/h)</th>
</tr>
</thead>
<tbody>
<tr>
<td>GBICA [27]</td>
<td>1.1881</td>
</tr>
<tr>
<td>MGBICA [27]</td>
<td>1.1724</td>
</tr>
<tr>
<td>PSO</td>
<td>1.0814</td>
</tr>
<tr>
<td>DA</td>
<td>1.3097</td>
</tr>
<tr>
<td>DA-PSO</td>
<td>1.0799</td>
</tr>
</tbody>
</table>

Table 8. Comparison of the results from DA-PSO and its traditional algorithms when considering only transmission losses as part of the objective function for the IEEE 57-bus system.

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>Loss (MW)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSO</td>
<td>10.7076</td>
</tr>
<tr>
<td>DA</td>
<td>13.6430</td>
</tr>
<tr>
<td>DA-PSO</td>
<td>10.1212</td>
</tr>
</tbody>
</table>

5.2.2. MO-OPF

This case proposes a multiobjective optimization problem by using the proposed DA-PSO algorithm to evaluate its performance for the IEEE 57-bus test system. The two-dimensional Pareto fronts provided by the PSO and DA-PSO algorithms for this system are shown in Figures 6–8, while DA could not provide the convergent Pareto fronts for any multiobjective functions in this system. In Figure 9, the three-dimensional Pareto front obtained from DA-PSO is shown. From all figures for this system, the fronts obtained from DA-PSO algorithm are superior to those from PSO, while the fronts obtained from DA could not converge because the best experience of dragonflies in DA is not applied during the operation and the obtained solutions are trapped in the local optima. From the results, it can be seen that the proposed hybrid DA-PSO performs better than the original DA and PSO algorithms once again.

Figure 6. Two-dimensional Pareto fronts when considering fuel cost and emissions as part of the objective function for the IEEE 57-bus system.

Figure 7. Two-dimensional Pareto fronts when considering fuel cost and transmission losses as part of the objective function for the IEEE 57-bus system.

Figure 8. Two-dimensional Pareto fronts when considering emissions and transmission losses as part of the objective function for the IEEE 57-bus system.

Figure 9. Three-dimensional Pareto fronts when considering fuel cost, emissions, and transmission losses as part of the objective function for the IEEE 57-bus system.
6. Conclusions

In this paper, a hybrid DA-PSO algorithm is proposed to solve the MO-OPF problem in a power system. As the DA is an algorithm that applies Levy flight to improve its randomness and stochastic behavior, this could significantly develop the exploration phase of the algorithm in an optimization. The PSO could quickly converge on the optimal solution because of its equations for finding optimal solutions by using the best experience of the particles. This makes PSO perform well at the exploitation phase in an optimization. The new hybrid DA-PSO algorithm combines the prominent points of these two algorithms, which are the exploration phase of DA and the exploitation phase of the PSO, to improve its performance for finding the optimal solution of the OPF problem. The proposed algorithm was used to minimize fuel cost, emissions, and transmission losses, which are considered to be parts of the objective function. The standard IEEE 30-bus and 57-bus systems were employed to investigate the performance of the proposed algorithm to find the optimal settings of the control variables. In order to investigate the single-objective and multiobjective optimizations, the simulation was divided into two cases. First, the proposed algorithm was used to solve a single-objective function. The results from the proposed algorithm show its superiority over other optimization algorithms in the literature. For the other case, the DA-PSO was successfully employed to solve the MO-OPF problem because the Pareto fronts generated by DA-PSO are better than those obtained by the original DA and PSO algorithms. All simulation results support the applicability, potential, and effectiveness of the proposed algorithm. However, the computation time of the DA-PSO is much slower than other algorithms in the literature because of the sequential computation of DA and PSO.


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

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

ABC artificial bee colony
ACO ant colony optimization
APSO adaptive particle swarm optimization strategy
ARCBBO adaptive real coded biogeography-based optimization
DA dragonfly algorithm
DE differential evolution
DE-PS differential evolution pattern search
DSA differential search algorithm
EADDE evolving ant direction differential evolution
EEA efficient evolutionary algorithm
EGA enhanced genetic algorithm
EGA-DQLF enhanced genetic algorithm with decoupled quadratic load flow
EP evolutionary programming
EP-OPF evolutionary-programming-based optimal power flow
Fuzzy-GA fuzzy genetic algorithm
GA genetic algorithm
GSA gravitational search algorithm
GWO grey wolf optimizer
HBMO honey bee mating optimization
HMPSO-SFLA hybrid modified particle swarm optimization-shuffle frog leaping algorithms
IEP improved evolutionary programming
IPSO improved particle swarm optimization
ISPEA2 improved strength Pareto evolutionary algorithm
LDI-PSO particle swarm optimization algorithm with linearly decreasing inertia weight
LTLBO Levy mutation teaching–learning-based optimization
MDE-OPF modified differential evolution optimal power flow
MGBICA modified Gaussian bare-bones multiobjective imperialist competitive algorithm
MICA-TLA hybrid modified imperialist competitive algorithm and teaching–learning algorithm
MO-DEA multiobjective differential evolution algorithm
MOHS multiobjective harmony search
MOMICA multiobjective modified imperialist competitive algorithm
MO-OPF multiobjective optimal power flow
MSFLA modified shuffle frog leaping algorithm
MTLBO modified teaching–learning-based optimization
NPSO new particle swarm optimization
OPF optimal power flow
P-OPF probabilistic optimal power flow
PSO particle swarm optimization
SGA stochastic genetic algorithm
TS tabu search

References


© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).