Hindawi
Journal of Electrical and Computer Engineering
Volume 2020, Article ID 6382507, 11 pages
https://doi.org/10.1155/2020/6382507

Research Article

Optimal Reactive Power Flow for Large-Scale Power Systems Using an Effective Metaheuristic Algorithm

Thanh Long Duong,[1] Minh Quan Duong,[2] Van-Duc Phan,[3] and Thang Trung Nguyen[4]

[1]*Faculty of Electrical Engineering Technology, Industrial University of Ho Chi Minh City, Ho Chi Minh City, Vietnam*
[2]*Electrical Engineering Faculty, The University of DaNang, University of Science and Technology, DaNang City, Vietnam*
[3]*Faculty of Automobile Technology, Van Lang University, Ho Chi Minh City 700000, Vietnam*
[4]*Power System Optimization Research Group, Faculty of Electrical and Electronics Engineering, Ton Duc Thang University, Ho Chi Minh City 700000, Vietnam*

Correspondence should be addressed to Thang Trung Nguyen; nguyentrungthang@tdtu.edu.vn

Received 20 August 2019; Revised 7 January 2020; Accepted 13 January 2020; Published 1 March 2020

Guest Editor: Alessandro Niccolai

In this paper, stochastic fractal search method (SFS) is employed for solving the optimal reactive power flow (ORPF) problem with a target of optimizing total active power losses (TPL), voltage deviation (VD), and voltage stability index (VSI). SFS is an effective metaheuristic algorithm consisting of diffusion process and two update processes. ORPF is a complex problem giving challenges to applied algorithms by taking into account many complex constraints such as operating voltage from generators and loads, active and reactive power generation of generators, limit of capacitors, apparent power limit from branches, and tap setting of transformers. For verifying the performance, solutions of IEEE 30 and 118-bus system with TPL, VD, and VSI objectives are found by the SFS method with different control parameter settings. Result comparisons indicate that SFS is more favorable than other methods about finding effective solutions and having faster speed. As a result, it is suggested that SFS should be used for ORPF problem, and modifications performed on SFS are encouraged for better results.

1. Introduction

In the power system, optimal reactive power flow (ORPF) is not only one of the best famous optimization problems but also a very complex problem. In the ORPF problem, two variables need to be considered such as control variables and dependent variables. Control variables are voltage of generation buses, on load tap-changer setting of transformers and generated reactive power of capacitor banks, while dependent variables are voltage of load buses, apparent power flow of transmission lines, and reactive power of generators. So, the major objectives of such ORPF problem is to find control variable so that others have values falling into a permitted operating range [1, 2]. Traditionally, the ORPF problem concentrates on reducing three individual objectives such as power losses of transmission lines, voltage deviation, and voltage stability index. So, a power system economically and stably operates when these goals are fully achieved.

In the last decades of the 20th century, the ORPF problem has been successfully addressed by many conventional methodologies called deterministic methodologies such as the Newton method [3], linear programming [4–7], interior point method [8, 9], quadratic programming method [10, 11], and dynamic programming method [12]. With appearance of the mentioned methods, they proved their strong points in dealing with the ORPF problem having linear constraints and differentiable functions for application, but a large system or more complicated constraints and their applicability must be stopped to make rooms for new methods which have a promising ability.

Luckily, developing computer science supported researchers much in creating new population-based methods to handle drawbacks of conventional methods. These

methods have been successfully and widely applied to solving the ORPF problem, consisting many original methods, improve/modified methods, or combined/hybrid methods. They have been constantly developed and have become a big method family such as particle swarm optimization (PSO) family [13–17], differential evolution (DE) family [18–21], and genetic algorithm (GA) family [22–25], while many standard methods have been also applied in [26–34]. Sahli et al. [16] presented a combination between particle swarm optimization and tabu search (PSO-TS) by incorporating the best search function of PSO and TS. It was capable of finding the global solution and avoiding to fall into local optimum. By the way of evaluation of results attained from the standard IEEE 30-bus system with objective of power loss minimization, PSO-TS has seen better solution quality than other methods as conventional DE, PSO, and TS. Furthermore, a modified version of PSO called modified pseudogradient search-particle swarm optimization (MPG-PSO) has been proposed in [17]. MPG-PSO has the most powerful ability in the PSO family due to applying pseudogradient theory for determining the best velocity direction. As a result, the method has overtaken other PSO methods involving PSO using the time-varying inertia weight factor (PSO-TVIW), PSO using time-varying acceleration coefficients (PSO-TVAC), self-organizing particle swarm optimization using time-varying acceleration coefficients (SPSO-TVAC), PSO using constriction factor (PSO-CF), pseudogradient-based PSO (PG-PSO), PSO using stochastic weight trade-off factor (SWT-PSO), and SWT-PSO using pseudogradient method (PGSWT-PSO). Differential evolution (DE) family has been offered for ORPF-like traditional differential evolution (DE), hybrid ant system and differential evolution method (HAS-DE) [20], and hybrid double differential evolution technique and modified teaching learning technique (DDET-MTLT) [21]. In [20], Huang and Huang have replaced the selection operation of DE by the ant system to enhance the global search capability and avoid falling into local minima and decrease computational time. DDET-MTLT [21] was a combination of double differential evolution technique (DDET) and modified teaching learning technique (MTLT). The obtained results of DDET-MTLT have been compared to some methods on IEEE-30 and IEEE 118-bus systems. Besides, variants of genetic algorithm have been applied for ORPF such as genetic algorithm (GA) [22], enhanced genetic algorithm (EGA) [23], modified NSGA-II (MNSGA-II) [24], self-adaptive real coded genetic algorithm (SARCGA) [25], and hybrid evolutionary programming technique HEP [25].

In addition to the three above method family, other standard methodologies have been also applied for solving ORPF problem such as gravitational search algorithm (GSA) [26], ant lion optimizer (ALO) [27], quasi-oppositional teaching learning based optimization (QOTLBO) [28], teaching learning based optimization (TLBO) [28], Pooled-neighbor swarm intelligence algorithm (PNSA) [29], hybrid Nelder–Mead simplex-based firefly algorithm (HFA-NMS) [30], chaotic krill herd algorithm (CKHA) [31], artificial bee colony algorithm (ABC) [32], exchange market algorithm (EMA) [33], backtracking search algorithm (BTSA) [34], and harmony search algorithm (HSA) [35]. In summary, all methods have demonstrated their qualification for addressing almost constraints of ORPF problem with acceptable solutions.

In this article, we present a standard stochastic fractal search (SFS) with the goal determining minimization of three different individual objectives of ORPF problem such as minimizing TPL, reducing of VD, and enhancing VSI. The standard stochastic fractal search [36] was developed in 2014 by Salimi. It has been applied for addressing twenty-three standard benchmark functions and has proven its proficiency in finding optimal solutions better than many methods available in this literature. In [37], Tran at al. have applied an improved SFS (ISFS) method and SFS method for dealing with the ORPF problem. Only the IEEE 30-bus system with three objectives consisting of total power losses, voltage deviation and L-index have been employed for comparing and evaluating the real performance of the SFS method and ISFS method. From the minimum, average, and maximum fitness functions obtained from the three objectives, ISFS has been considered to be more effective than the SFS method. However, Tran at al. [37] have not taken the setting of control parameters into account. In fact, SFS has three important control parameters consisting of walk factor, population size, and the maximum number of iterations. Among the three parameters, the walk factor has high impact on working performance of the diffusion process; meanwhile, population size has high contribution to the first update and the second update processes. So, in this paper, we focus on the setting of control parameters to overcome such mentioned shortcoming of the work in [37]. Furthermore, we also expand study cases by considering both IEEE 30-bus system and IEEE 118-bus system with the three mentioned objectives. As a result, the novelty and the main contribution of the paper are as follows:

(i) Finding optimal solutions for IEEE 30-bus and 118-bus transmission power networks of ORPF problem by using the SFS method

(ii) Testing the real performance of the SFS method with the change of population size and iterations

(iii) Tuning the best walk factor for determining more appropriate equation for diffusion of the SFS method

(iv) Illustrating the fluctuations of search process of SFS method with different settings;the simulation can support to evaluate the real performance of SFS and the impact of each parameter on the real performance of the SFS method

(v) Demonstrating the effectiveness and robustness of the SFS method by comparing total power losses, load bus voltage, and enhancement of voltage profile

Apart from the introduction, other parts of the paper are as follows: single objective functions and constraints of electric components are mathematically formulated in Section 2. The structure of the SFS method consisting of diffusion and the first and the second update techniques are described in detail in Section 3. Computation steps of solving ORPF problem by using the SFS method are shown in

Section 4. The results obtained by SFS and other methods from two standard IEEE transmission power networks with 30 buses and 118 buses are compared and discussed in Section 5. The whole work of the paper is summarized in and concluded in Section 6.

2. Formulation of ORPF Problem

The ORPF problem is constructed by considering minimization of total active power losses (P_{loss}), reduction of load bus voltage deviation (VD), and enhancement of voltage stability as objectives and taking into account operating voltage of generators and loads, apparent power limit of branches, reactive power generation of generators, and active and reactive power balance equations. The structure of ORPF problem is described in detail as follows.

2.1. Objectives of ORPF Problem. Three main objectives of the ORPF problem considered in the paper consist of minimization of total active power losses, minimization of voltage deviation at load buses, and minimization of L-index in which minimization of L-index is corresponding to the enhancement of voltage stability. The three objectives can be seen in formulas (1)–(3) as follows:

$$\text{Minimize} \sum P_{\text{loss}} = \sum_{i=1}^{N_{\text{bus}}} \sum_{\substack{j=1 \\ j\neq i}}^{N_{\text{bus}}} G_{ij}\left[V_i^2 + V_j^2 - 2V_iV_j\cos\left(\beta_i - \beta_j\right)\right], \tag{1}$$

$$\text{Minimize VD} = \sum_{i=1}^{N_{\text{load}}} \left|V_{\text{loadi}} - V_{\text{ref}}\right|, \tag{2}$$

$$\text{Minimize L} - \text{index} = \max\left(L_j\right); \quad j = 1, \ldots, N_{\text{bus}}, \tag{3}$$

where G_{ij} is the conductance of conductor ij; β_i and β_j are the phases of voltage at buses i and j, respectively; V_{loadi} is the voltage of load bus i; V_{ref} is expected to be the voltage equaling 1.0 pu; and L_j is called L-index of-bus j [16].

Basically, L-index is within the range from 0 to 1 in which 0 is the best value and 1 is the worst value. The power system is considered to be working stably when L-index is close to 0, and it is working unstably or it will be collapsed in some seconds if L-index is close to 1. So, the main task to keep the power system working stably and economically is to reduce L-index close to 0. However, it is hard to tune control parameter for obtaining 0 value for L-index.

2.2. Constraints of ORPF Problem. ORPF are constrained by equality and inequality constraints covering the whole transmission power network. The equality constraints are considering active and reactive power balance while the inequality constraints are upper and lower limitations of electricity components and working parameters of power network.

The two equality constraints are as follows:

$$P_{Gi} - P_{di} = V_i \sum_{j=1}^{N_{\text{bus}}} V_j\left[g_{ij}\cos\left(\beta_i - \beta_j\right) + b_{ij}\sin\left(\beta_i - \beta_j\right)\right], \tag{4}$$

$$Q_{Gi} + Q_{ci} - Q_{di} = V_i \sum_{j=1}^{N_{\text{bus}}} V_j\left[g_{ij}\sin\left(\beta_i - \beta_j\right) - b_{ij}\cos\left(\beta_i - \beta_j\right)\right]. \tag{5}$$

where g_{ij} and b_{ij} are real and unreal terms of admittance of conductor ij, respectively. The inequality constraints are limitations of reactive power output and voltage of generators, the reactive power output of capacitors, tap values of transformers, voltage of load buses, and apparent power of lines. All the inequality constraints are as follows:

$$Q_{Gi,\min} \le Q_{Gi} \le Q_{Gi,\max}; \quad i = 1, \ldots, N_G, \tag{6}$$

$$V_{Gi,\min} \le V_{Gi} \le V_{Gi,\max}; \quad i = 1, \ldots, N_G, \tag{7}$$

$$Q_{ci,\min} \le Q_{ci} \le Q_{ci,\max}; \quad i = 1, \ldots, N_c, \tag{8}$$

$$T_{i,\min} < T_i < T_{i,\max}; \quad i = 1, \ldots, N_t, \tag{9}$$

$$V_{\text{loadi},\min} \le V_{\text{loadi}} \le V_{\text{loadi},\max}; \quad i = 1, \ldots, N_{\text{load}}, \tag{10}$$

$$S_l \le S_{l,\max}; \quad l = 1, \ldots, N_{\text{branch}}. \tag{11}$$

3. Stochastic Fractal Search Algorithm

3.1. Diffusion Technique. SFS is an improved version of fractal search and was developed by Salimi in 2014 [36]. SFS is constructed by three main processes including diffusion and two different update processes. Consequently, SFS has three new solution generations in each iteration in which the diffusion process plays the most important role. The diffusion process uses Gaussian random walk for generating new solutions as follows:

$$X_{s,\text{new}}^1 = \text{Gaussian}\left(G_{\text{best}}, \sigma\right) + \text{rand} \times \left(G_{\text{best}} - X_s\right), \tag{12}$$

$$X_{s,\text{new}}^1 = \text{Gaussian}\left(X_s, \sigma\right), \tag{13}$$

$$\sigma = \left|\frac{\log\left(\text{CI}\right)}{\text{CI}} \times \left(X_s - G_{\text{best}}\right)\right|, \tag{14}$$

where G_{best} is the best solution among the set of points. CI is the current iteration.

As seen from equations (12) and (13), the diffusion process can be accomplished by using either equations (12) or (13) meanwhile the two equations have a major difference. Equation (12) uses Gaussian random walk around the best solution and an updated step by using $(G_{\text{best}} - X_s)$ while equation (13) only employs random walk around X_i. Due to the difference, SFS must propose one more control parameter, called walk factor (WF) to control the use of either equations (12) or (13). The walk factor is compared to a

random number and the result will lead to a decision of used equation. If walk factor is higher than the random number, equation (12) is employed for producing new solution for the sth solution. Otherwise, equation (13) is selected. The diffusion technique can be performed by using Algorithm 1.

3.2. Second Update Technique. The first update is the second new solution generation and is performed by the following equation:

$$X_{s,\text{new}}^2 = X_{\text{random1}} - \text{rand} \times \left(X_{\text{random2}} - X_s\right), \tag{15}$$

where X_{random1} and X_{random2} are solutions chosen randomly in the current set of solutions and $X_{s,\text{new}}^2$ is the new solution of solution X_s.

3.3. Third Update Technique. After implementing the first update, the second update equivalent to the third new solutions is carried out for updating solutions of the second generation and is expressed by

$$X_{s,\text{new}}^3 = \begin{cases} X_{s,\text{new}}^2 - \text{rand} \times \left(X_{\text{random1}} - G_{\text{best}}\right), & \text{for } RN_s \le 0.5 \\ X_{s,\text{new}}^2 + \text{rand} \times \left(X_{\text{random2}} - X_{\text{random3}}\right), & \text{for } RN_s > 0.5 \end{cases} \tag{16}$$

where X_{random1}, X_{random2}, and X_{random3} are three random solutions chosen from the current set and RN_s is a random number ranging from 0 to 1 for solution s.

4. Implementation of SFS for ORPF Problem

4.1. Initialization. The ORPF problem can be solved by using mathpower 4.1 programming after a set of control variables is predetermined by the SFS problem. In fact, the control variable set that has to be inserted into the program is the voltage of thermal generators ($V_{\text{G1}}, \ldots, V_{\text{GNG}}$), tap values of transformers ($T_1, \ldots, T_{\text{Nt}}$), and reactive power output of capacitor banks ($Q_{\text{c1}}, \ldots, Q_{\text{cNc}}$). Thus, each point (corresponding to each solution) of the SFS method must represent all the variables and is randomly produced as follows:

$$X_s = X_{\min} + \text{rand} \times \left(X_{\max} - X_{\min}\right); \quad s = 1, \ldots, N_{\text{pop}}, \tag{17}$$

where $X_{\min}$ and $X_{\max}$ are the lowest and highest values of such control variables.

4.2. Fitness Function. As a result, the fitness function is constructed equaling the sum of objective function and penalty terms [17]. For three single objective functions including total power loss, voltage deviation, and L-index, three corresponding fitness functions are formulated as follows:

$$\begin{aligned} \text{FF}_s &= \sum P_{\text{loss}} + \text{VPF} \cdot \left(\Delta Q_{Gi} + \Delta V_{\text{loadi}} + \Delta S_l\right)^2, \\ \text{FF}_s &= \text{VD} + \text{VPF} \cdot \left(\Delta Q_{Gi} + \Delta V_{\text{loadi}} + \Delta S_l\right)^2, \\ \text{FF}_s &= \text{L} - \text{index} + \text{VPF} \cdot \left(\Delta Q_{Gi} + \Delta V_{\text{loadi}} + \Delta S_l\right)^2. \end{aligned} \tag{18}$$

```
Set a value to WF
For s = 1 to N_pop
    Randomly produce a random number λ_s for the
    sth solution
    If WF is higher than λ_s
        Using equation (12) for updating the
        sth new solution
    Else
        Using equation (13) for updating the
        sth new solution
    End
end
```

Algorithm 1: Diffusion technique for the first generation of new solutions.

where $\sum P_{\text{loss}}$, VD and L-index are obtained by using equations (1)–(3), VPF is the penalty factor for the violation of the dependent variable, and ΔQ_{Gi}, ΔV_{loadi}, and ΔS_l are penalty terms corresponding to the violation of Q_{Gi} in constraint (6), V_{loadi} in constraint (10), and S_l in constraint (11).

The penalty terms are determined by

$$\begin{aligned} \Delta Q_{Gi} &= \begin{cases} 0, & \text{if } Q_{Gi,\min} < Q_{Gi} < Q_{Gi,\max}, \\ \left(Q_{Gi,\min} - Q_{Gi}\right), & \text{if } Q_{Gi,\min} \ge Q_{Gi}, \\ \left(Q_{Gi} - Q_{Gi,\max}\right)^2, & \text{if } Q_{Gi,\max} \le Q_{Gi}, \end{cases} \\ \Delta V_{loadi} &= \begin{cases} 0, & \text{if } V_{\text{loadi},\min} < V_{\text{loadi}} < V_{\text{loadi},\max}, \\ \left(V_{\text{loadi},\min} - V_{\text{loadi}}\right), & \text{if } V_{\text{loadi},\min} \ge V_{\text{loadi}}, \\ \left(V_{\text{loadi}} - V_{\text{loadi},\max}\right)^2, & \text{if } V_{\text{loadi},\max} \le V_{\text{loadi}}, \end{cases} \\ \Delta S_l &= \begin{cases} 0, & \text{if } S_l \le S_{l,\max}, \\ \left(S_l - S_{l,\max}\right), & \text{else.} \end{cases} \end{aligned} \tag{19}$$

4.3. Termination Condition. The three main processes of the SFS method will be terminated when current iteration (CI) is equal to the maximum iteration (MI), which is predetermined initially.

5. Numerical Results

In the section, we test the performance of the SFS method for the ORPF problem with two systems having 30 buses and 118 buses under considering three objectives such as power loss, voltage deviation, and L-index. The method is executed on Matlab program language and a computer with the processor of Core i7, 2.4 GHz, and 4 GB of RAM.

5.1. Results Obtained on IEEE 30-Bus System. In the section, we implement the SFS method for optimizing total power losses, voltage deviation, and L-index of the IEEE 30-bus

system by setting different values to control parameters such as population size, maximum iteration, and walk factor. The IEEE 30-bus system consists of 6 generators, 24 loads, and 41 branches, 9 VAR compensators, and 4 transformers [38]. In order to indicate the impact of control parameters on results, we perform three testing cases as follows:

(1) The first testing case: tune different values for population size while fixing the maximum iteration at value of 200

(2) The second testing case: tune different values for the maximum iteration while fixing population size to the best value obtained by the first testing case

(3) The third testing case: tune different values for walk factor while fixing population size and the maximum iteration at the best value obtained by the first testing case and the second testing case

The three testing cases are presented in Section 5.1.1, while the comparison of results from SFS method and other ones is shown in Section 5.1.2 and Section 5.2.

5.1.1. Analysis of Control Parameters on Results Obtained by SFS Method. The results in terms of the best, mean, worst power losses, standard deviation, and simulation time for each run (ST) are shown in Tables 1–3 corresponding to the first, second, and third testing cases. The results from Table 1 indicate that population size of 10 is high enough for finding the best solution while increasing the population size to 15 and 20 cannot find a better solution but average and maximum power losses can be improved whereas simulation time is increased. Figure 1 also gives the same evaluation since the curve in blue has high number of solutions with better fitness than other curves.

Table 2 show the results obtained by setting different values to the maximum iteration while keeping the population size at 10. The maximum iteration is increased from 50 to 250 with a step of 50. The best power loss implies that SFS method can find the best solution at different values of the maximum iteration such as 200 and 250; meanwhile, smaller number of iterations cannot reach the same best solution. The best optimal solution of MI = 200 and MI = 250 has the same quality; meanwhile, the mean fitness function value of 50 runs from MI = 250 is smaller. In spite of the advantage of MI = 250, SFS should adopt the most appropriate number of iterations is 200 because it can find as good as solution but it can reduce simulation time. Figure 2 shows that the curve in blue has many solutions with the same fitness as those from curve in black.

From Tables 1 and 2 as well as from Figures 1 and 2, it can result in the selection of population size and the maximum iteration in which the best value of the comer is 10 while the most appropriate value of the later is 200. By using the two values, walk factor's impact on performance of the SFS method is tried by setting it to 0, 0.2, 0.4, 0.6, 0.8, and 1. Table 3 and Figure 3 show that the walk factor to be 1 is the best value for reaching the best solution, whereas other values cannot reach the same best solution. As observing from the best power loss, the walk factor with higher value can support the SFS method find superior solutions. In fact, power loss is the highest at WF = 0, and it decreases since the factor approaches to 1. So, it can conclude that diffusion technique becomes more effective if equation (12) is used to replace equation (13).

5.1.2. Result Comparisons for IEEE 30-Bus System. The results in terms of minimum, average, maximum, and standard deviation accompany with control parameters including MI and N_{pop} from SFS and other methods are reported in Tables 4–6 for power loss, voltage deviation, and L-index, respectively. Besides, saving percentage (%) of the SFS method compared with each one is also calculated and reported in the tables for further comparisons. Saving percentage from these tables can see that SFS outperforms most methods excluding PSO-TS [16] and ISFS [37] for power loss objective and QOTLBO [28] for voltage deviation objective. Saving values show that these methods get improvement over SFS by 0.137%, 0.295%, and 2.45%; however, only ISFS [37] has found better solution than SFS, meanwhile PSO-TS [16] has not reported MI and N_{pop} for comparison of convergence speed and recalculated minimum of QOTLBO is 0.1031, which is much higher than reported value of 0.0856. Clearly, QOTLBO is less effective than SFS. SFS can get improvement over other methods for power loss optimization from 0.7% to 8.365%, for voltage deviation optimization from 1.68% to 57.51%, and for L-index from 13.26% to 32.82%. Clearly, the improvement is significant and optimal solution of SFS is much better than those from other methods. Furthermore, convergence speed of SFS is also faster or approximate with other compared methods since SFS uses MI = 200, $N_{\text{pop}} = 10$, while those from others are 100 and 50, 200 and 20, or MI = 30,000.

Optimal solutions of three optimization cases are given in Table 7.

5.2. Results Obtained on IEEE 118-Bus System. This section uses the IEEE 118-bus system consisting of 54 generator buses, 64 load buses, and 186 transmission lines, 14 VAR compensators, and 9 transformers [37] (Table 8). We implement the SFS method for optimizing total power losses of the system by setting different values to population size and the maximum iteration. In the first trial, we increase the population size from 10 to 100 with a change of 10 and fixing the maximum iteration to 150 and 200. In the second trial, population size is selected to be 50 and 75; meanwhile, the maximum iteration is increased from 50 to 400 with a step of 50. The optimal solution obtained by SFS method is given in Table 9. In addition, the results in terms of the best power loss, mean power loss, and success rate (SR) in percentage are, respectively, shown in Tables 9 and 10. From the two tables, it is clear that SFS can improve search ability since the

Table 1: Results obtained by setting different values to N_{pop}, and fixing MI = 200 and WF = 1.0.

Population size	The best TPL (MW)	Mean TPL (MW)	The worst TPL (MW)	Std. dev. TPL (MW)	ST (s)
5	5.0708	5.2901	6.3250	0.1847	10.2
10	4.5275	4.6732	5.1844	0.1399	19.7
15	4.5275	4.6639	5.1688	0.1370	29.4
20	4.5275	4.6499	5.1430	0.1352	38.2

Table 2: Results obtained by setting different values to MI, and fixing $N_{pop} = 10$ and WF = 1.0.

MI	The best TPL (MW)	Mean TPL (MW)	The worst TPL (MW)	Std. dev. TPL (MW)	ST (s)
50	6.3838	7.1500	8.1395	0.2252	6.2
100	6.2561	7.0070	7.9767	0.2207	12.1
150	4.7539	6.3063	7.2588	0.1876	18.6
200	4.5275	4.6732	5.1844	0.1399	23.9
250	4.5275	4.6078	5.0029	0.1321	30.0

Table 3: Results obtained by setting different values to WF and fixing $N_{pop} = 10$ and MI = 200.

Walk factor (WF)	The best TPL (MW)	Mean TPL (MW)	The worst TPL (MW)	Std. dev. TPL (MW)	ST (s)
0	6.9724	7.1687	7.6936	0.2154	20.2
0.2	5.9854	6.6453	7.6325	0.2129	20.1
0.4	5.5236	6.1780	7.2167	0.1707	19.7
0.6	4.7539	4.7900	5.3659	0.1560	19.6
0.8	4.5366	4.7293	5.7132	0.1430	19.8
1.0	4.5275	4.6732	5.1844	0.1399	19.5

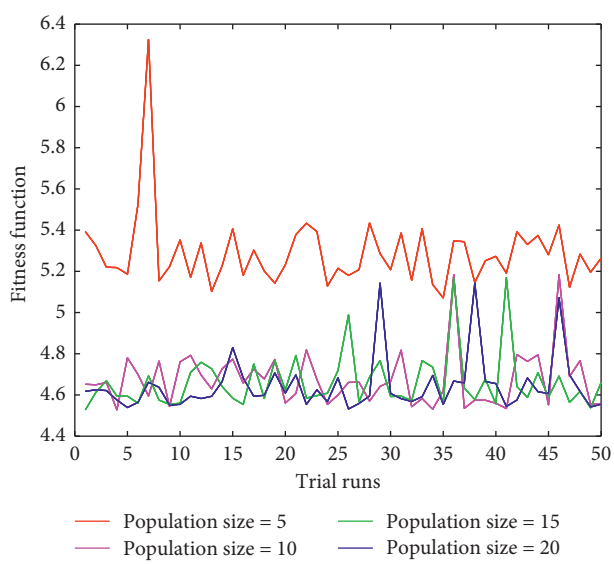

Figure 1: The impact of population size on obtained fitness values of 50 trial runs.

population size and the maximum iteration are increased but simulation time is also increased. 113.021 MW in Table 9 indicates SFS can find the best solution with the setting of $N_{pop} = 60$, 70, 80, 90, and 100, and MI = 150, and $N_{pop} = 50$, 60, 70, 80, 90, and 100 and MI = 200. Clearly, $N_{pop} = 60$ and MI = 150 are the best selection for the case of tuning population size and fixing the maximum iteration. SFS cannot improve result better if the population size is set to higher than 60 for MI = 150 and 70 for MI = 200. Similarly, 113.021 MW in Table 10 indicates that $N_{pop} = 50$ and MI = 200 and $N_{pop} = 75$ and MI = 150 are the best selection for finding the best performance of the SFS method. Improvement of the best solution fails if increasing MI to higher than 200 for $N_{pop} = 50$ and to higher than 150 for

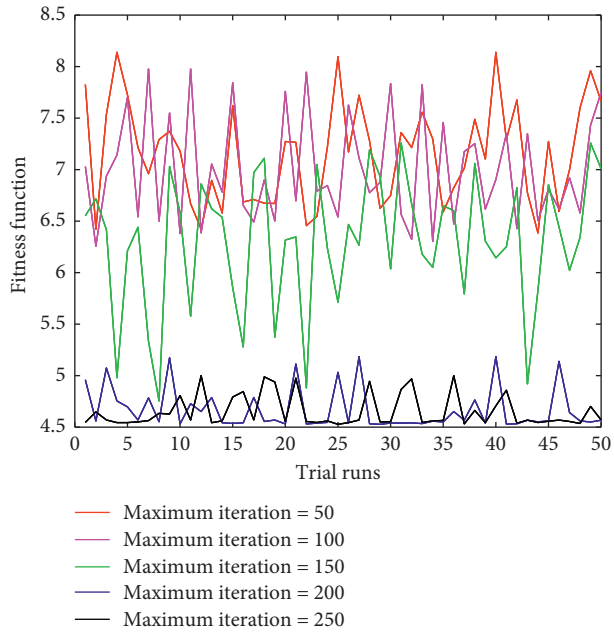

Figure 2: The impact of the maximum iteration on obtained fitness values of 50 trial runs.

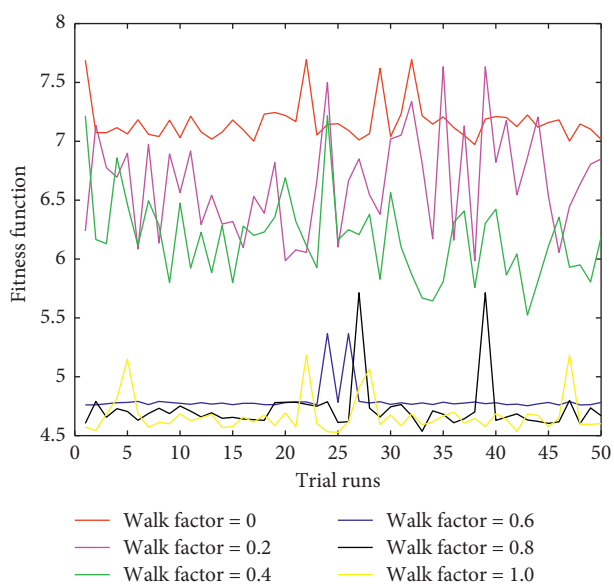

Figure 3: The impact of walk factor on obtained fitness values of 50 trial runs.

Table 4: Comparisons for TPL of the IEEE 30-bus system.

Method	The best TPL (MW)	Mean TPL (MW)	The worst TPL (MW)	Std. dev. TPL (MW)	MI	N_{pop}	Saving percentage (%)
PSO-TS [16]	4.5213	—	—	—	—	—	−0.137
TS [16]	4.9203	—	—	—	—	—	7.983
PSO [16]	4.6862	—	—	—	—	—	3.387
ALO [27]	4.59				100	—	1.362
QOTLBO [28]	4.5594	4.5601	4.5617	0.037	100	50	0.700
TLBO [28]	4.5629	4.5695	4.5748	0.0564	100	50	0.776
SGA [35]	4.9408	5.0378	5.1651	—	30,000	—	8.365
PSO [35]	4.9239	4.972	5.0576	—	30,000	—	8.051
HSA [35]	4.9059	4.924	4.9653	—	30,000	—	7.713
SFS [37]	4.5777	—	—	1.05	100	20	1.097
ISFS [37]	4.5142	—	—	0.012	—	—	−0.295
SFS	4.5275	4.6732	5.1844	0.1399	200	10	

Table 5: Comparisons for VD of the IEEE 30-bus system.

Method	The best VD (pu)	Mean VD (pu)	The worst VD (pu)	Std. dev. VD (pu)	MI	N_{pop}	Saving percentage (%)
PSO-TVIW [17]	0.1038	0.1597	0.5791	0.1112	200	20	15.51
PSO-TVAC [17]	0.2064	0.2376	0.5796	0.0153	200	20	57.51
SPSO-TVAC [17]	0.1354	0.1558	0.1833	0.0103	200	20	35.23
PSO-CF [17]	0.1287	0.1557	0.4041	0.0404	200	20	31.86
PG-PSO [17]	0.1202	0.144	0.2593	0.0222	200	20	27.04
SWT-PSO [17]	0.1614	0.1814	0.2296	0.133	200	20	45.66
PGSWT-PSO [17]	0.1539	0.2189	0.5532	0.0656	200	20	43.01
MPG-PSO [17]	0.0892	0.1078	0.2518	0.0298	200	20	1.68
QOTLBO [28]	0.0856	0.0872	0.0907	0.0314	200	10	−2.45
TLBO [28]	0.0913	0.0934	0.0988	0.0403	100	50	3.94
SFS [37]	0.122	—	—	0.016	100	20	28.115
ISFS [37]	0.089	—	—	0.0031	100	20	1.461
SFS	0.0877	0.1207	0.1577	0.0155	200	10	—

Table 6: Comparisons for VSI of the IEEE 30-bus system.

Method	The best L-index (pu)	Mean L-index (pu)	The worst L-index (pu)	Std. dev. L-index (pu)	MI	N_{pop}	Saving percentage (%)
PSO-TVIW [17]	0.1258	0.127	0.1289	0.0008	200	20	19.95
PSO-TVAC [17]	0.1499	0.1513	0.1544	0.0009	200	20	32.82
SPSO-TVAC [17]	0.1271	0.1285	0.1297	0.0006	200	20	20.77
PSO-CF [17]	0.1261	0.1279	0.1295	0.0008	200	20	20.14
PG-PSO [17]	0.1264	0.1297	0.1313	0.0008	200	20	20.33
SWT-PSO [17]	0.1488	0.1634	0.1806	0.0074	200	20	32.33
PGSWT-PSO [17]	0.1394	0.1537	0.1749	0.0081	200	20	27.76
MPG-PSO [17]	0.1241	0.1266	0.1298	0.001	200	20	18.86
BA [27]	0.1191	—	—	—	100	40	15.45
GWO [27]	0.118	—	—	—	100	40	14.66
ABC [27]	0.1161	—	—	—	100	40	13.26
ALO [27]	0.1161	—	—	—	100	40	13.26
QOTLBO [28]	0.1242	0.1245	0.1247	0.0452	100	50	18.92
TLBO [28]	0.1252	0.1254	0.1258	0.0454	100	50	19.57
SFS [37]	0.1252	—	—	0.021	100	20	19.569
ISFS [37]	0.1245	—	—	0.004	100	20	19.116
SFS	0.1007	0.1043	0.1138	0.0026	200	10	—

Table 7: Optimal solutions of the IEEE 30-bus system.

Variable	TPL objective	VD objective	L-index objective
V_{G1}	1.1000	1.0055	1.0807
V_{G2}	1.0952	1.0011	1.0538
V_{G5}	1.0750	1.0173	1.0739
V_{G8}	1.0765	1.0120	1.0089
V_{G11}	1.0873	1.0320	1.0801
V_{G13}	1.0999	1.0234	1.0851
T_1	1.0500	1.0500	0.9000
T_2	0.9200	0.9000	0.9000
T_3	1.0100	1.0000	0.9000
T_4	0.9800	0.9700	0.9000
Q_{c1}	5.0000	4.0000	5.0000
Q_{c2}	5.0000	2.0000	5.0000
Q_{c3}	5.0000	4.0000	0.0000
Q_{c4}	5.0000	3.0000	0.0000
Q_{c5}	3.0000	5.0000	5.0000
Q_{c6}	5.0000	3.0000	3.0000
Q_{c7}	3.0000	5.0000	5.0000
Q_{c8}	4.0000	5.0000	5.0000
Q_{c9}	2.0000	3.0000	1.0000

Table 8: Optimal solution of the IEEE 118-bus transmission power network.

V_{G1} (pu)	1.0081	V_{G62}	1.0223	V_{G113}	1.005
V_{G4}	1.0439	V_{G65}	1.0563	V_{G116}	1.0379
V_{G6}	1.0376	V_{G66}	1.051	T_8 (pu)	0.9762
V_{G8}	1.0081	V_{G69}	1.0574	T_{32}	1.0192
V_{G10}	1.0267	V_{G70}	1.0308	T_{36}	1.0031
V_{G12}	1.0309	V_{G72}	1.0204	T_{51}	0.9785
V_{G15}	1.004	V_{G73}	1.0374	T_{93}	0.9981
V_{G18}	1.0021	V_{G74}	1.0043	T_{95}	1.0081
V_{G19}	0.9992	V_{G76}	0.989	T_{102}	1.0132
V_{G24}	1.0069	V_{G77}	1.0298	T_{107}	0.9963
V_{G25}	1.0436	V_{G80}	1.0471	T_{127}	0.9821
V_{G26}	1.0356	V_{G85}	1.0349	Q_{C5} (MVAR)	−2.349
V_{G27}	1.0366	V_{G87}	0.9944	Q_{C34}	7.4927
V_{G31}	1.023	V_{G89}	1.0764	Q_{C37}	−16.55
V_{G32}	1.0212	V_{G90}	1.0575	Q_{C44}	6.8138
V_{G34}	1.0266	V_{G91}	1.0464	Q_{C45}	3.8515
V_{G36}	1.0244	V_{G92}	1.0515	Q_{C46}	5.0318
V_{G40}	1.0371	V_{G99}	1.0285	Q_{C48}	7.1142
V_{G42}	1.0388	V_{G100}	1.0516	Q_{C74}	5.985
V_{G46}	1.0173	V_{G103}	1.0361	Q_{C79}	10.574
V_{G49}	1.0469	V_{G104}	1.0148	Q_{C82}	14.366
V_{G54}	1.0366	V_{G105}	1.0133	Q_{C83}	6.8917
V_{G55}	1.0321	V_{G107}	1.0265	Q_{C105}	6.4333
V_{G56}	1.0328	V_{G110}	1.0201	Q_{C107}	2.3679
V_{G59}	1.0388	V_{G111}	1.0197	Q_{C110}	5.3222
V_{G61}	1.0221	V_{G112}	1.0261		

Table 9: Result obtained by setting MI = 150, MI = 200 and different values for population size.

N_{pop}	MI = 150				MI = 200			
	The best TPL (MW)	Mean TPL (MW)	SR (%)	ST (s)	The best TPL (MW)	Mean TPL (MW)	SR (%)	ST (s)
10	161.620	206.233	15	46	156.772	193.034	18.69	61
20	154.186	196.747	25	90	149.560	184.155	31.15	119
30	145.552	187.696	47	126	141.185	175.684	54	163
40	135.945	179.062	54	181	131.867	167.602	61	239
50	122.894	159.423	65	226	113.021	149.220	75	302
60	**113.021**	134.793	88	275	113.021	126.166	89	354
70	113.021	130.075	92	317	113.021	121.750	94	422
80	113.021	125.783	95	359	113.021	124.651	96	477
90	113.021	121.758	94	402	113.021	120.662	98	534
100	113.021	117.618	95	449	113.021	116.559	98	572

Table 10: Result obtained by setting $N_{pop} = 50$ and $N_{pop} = 75$ and different values for the maximum iterations.

MI	$N_{pop} = 50$				$N_{pop} = 75$			
	The best TPL (MW)	Mean TPL (MW)	SR (%)	ST (s)	The best TPL (MW)	Mean TPL (MW)	SR (%)	ST (s)
50	154.308	212.230	12	72	124.323	198.647	25	107
100	125.454	202.467	54	147	119.124	189.510	69	221
150	119.011	193.154	77	226	**113.021**	130.221	82	340
200	**113.021**	131.027	91	293	113.021	121.887	90	437
250	113.021	129.590	93	365	113.021	121.107	92	545
300	113.021	126.856	95	438	113.021	120.545	94	656
350	113.021	123.036	97	511	113.021	114.952	96	768
400	113.021	119.578	98	583	113.021	114.648	96	876

$N_{pop} = 75$. So, it can conclude that the best selection for population size and the maximum iterations are $N_{pop} = 60$ and MI = 150. The results from the setting are reported for comparisons with other methods shown in Table 11. The saving percentage of SFS compared with other methods is from 0.09% to 9.1%, while SFS is only less effective than QOTLBO [28] by −0.66%. The comparison indication can conclude that optimal solution of SFS is the second best

Table 11: Comparisons for TPL of the IEEE 118-bus system.

Method	The best TPL (MW)	Mean TPL (MW)	The worst TPL (MW)	Std. dev. TPL (MW)	MI	N_{pop}	Saving percentage (%)
PSO-TVIW [17]	116.8976	118.234	126.6222	1.6009	200	40	3.32
PSO-TVAC [17]	124.3335	129.749	134.1254	2.156	200	40	9.10
SPSO-TVAC [17]	116.2026	117.355	118.139	0.4696	200	40	2.74
PSO-CF [17]	115.6469	116.986	119.8378	0.8655	200	40	2.27
PG-PSO [17]	116.6075	119.396	127.0772	2.107	200	40	3.08
SWT-PSO [17]	124.1476	129.371	141.6147	3.309	200	40	8.96
PGSWTPSO [17]	119.427	122.781	125.762	1.2455	200	40	5.36
MPG-PSO [17]	115.06	116.462	118.35	0.528	200	40	1.77
SARCGA [25]	113.12	113.968	—	0.0002	300	15	0.09
HEP [25]	115.58	115.8	—	0.0018	300	15	2.21
QOTLBO [28]	112.2789	113.769	115.4516	0.0244	100	50	−0.66
TLBO [28]	116.4003	121.390	118.4427	0.0482	100	50	2.90
SFS	113.0213	134.793	233.4538	2.91894	150	60	

among all compared methods. Furthermore, search speed of SFS is in the fastest group once MI and N_{pop} are, respectively, 150 and 60, while those from others are 200 and 40, 300 and 15, and 100 and 50. In summary, SFS can find optimal solutions with higher quality than most compared methods; however, compared with some best methods, SFS is also less potential. Consequently, we suggest SFS should be used as an optimization tool for the ORPF problem and it is more promising if SFS is improved by proposing modifications.

6. Conclusion

In the paper, we apply the SFS method for finding optimal solutions of the ORPF problem for different objectives consisting of power loss, voltage deviation, and voltage stability index. Two different power systems with 30 buses and 118 buses are employed for running the SFS method and results found by SFS together with control parameters are compared with those from other methods. As a result, SFS becomes one of the best methods searching the best optimal solutions for each case and its search speed is also faster than most methods. SFS can own the outstanding points, thanks to its construction consisting of three new solution generations, diffusion process, first update process, and the second update process. However, the performance of SFS still copes with constriction, leading to worse results than several methods about solution and speed. Thus, we suggest SFS should be used for finding solutions of the ORPF problem but modifications should be performed on the conventional SFS for improving the search ability.

Abbreviations

N_{bus}: Number of all buses in network
V_i, V_j: Voltage at bus i and bus j
N_{load}: Number of loads
P_{di}, Q_{di}: Active power and reactive power of load i
Q_{ci}: Reactive power generation of capacitor i
N_G: Number of available generators
$Q_{Gi,min}$, $Q_{Gi,max}$: Minimum and maximum limits of reactive power for generator i
$V_{Gi,min}$, $V_{Gi,max}$: Minimum and maximum voltage of generator i, respectively
$Q_{ci,min}$, $Q_{ci,max}$: Minimum and maximum reactive power output of capacitor at bus i
N_c: Number of capacitor buses
$T_{i,min}$, $T_{i,max}$: Minimum and maximum tap changer of transformer at bus i
N_t: Number of transformer buses
$V_{loadi,min}$, $V_{loadi,max}$: Minimum and maximum operating voltage of load i
$S_{l,max}$: Capacity of branch l.

Data Availability

The information of transmission lines and loads in IEEE 30-bus power transmission power network and IEEE 118-bus power transmission power network used to support the finding of this study have been taken from [38].

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This research was funded by Funds for the Science and Technology Development of the University of Danang, under project no. B2019-DN01-19.

References

[1] T. Nguyen, D. Vo, N. Vu Quynh, and L. Van Dai, "Modified Cuckoo search algorithm: a novel method to minimize the fuel cost," *Energies*, vol. 11, no. 6, p. 1328, 2018.
[2] J. Chintam and M. Daniel, "Real-power rescheduling of generators for congestion management using a novel Satin Bowerbird optimization algorithm," *Energies*, vol. 11, no. 1, p. 183, 2018.
[3] D. Sun, B. Ashley, B. Brewer, A. Hughes, and W. Tinney, "Optimal power flow by Newton approach," *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-103, no. 10, pp. 2864–2880, 1984.

[4] O. Alsac, J. Bright, M. Prais, and B. Stott, "Further developments in LP-based optimal power flow," *IEEE Transactions on Power Systems*, vol. 5, no. 3, pp. 697–711, 1990.

[5] G. Opoku, "Optimal power system VAR planning," *IEEE Transactions on Power Systems*, vol. 5, no. 1, pp. 53–60, 1990.

[6] B. Stott and J. Marinho, "Linear programming for power-system network security applications," *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-98, no. 3, pp. 837–848, 1979.

[7] A. M. Chebbo and M. R. Irving, "Combined active and reactive despatch. I. problem formulation and solution algorithm," *IEE Proceedings-Generation, Transmission and Distribution*, vol. 142, no. 4, pp. 393–400, 1995.

[8] S. Granville, "Optimal reactive dispatch through interior point methods," *IEEE Transactions on Power Systems*, vol. 9, no. 1, pp. 136–146, 1994.

[9] E. Rezania and S. M. Shahidehpour, "Real power loss minimization using interior point method," *International Journal of Electrical Power & Energy Systems*, vol. 23, no. 1, pp. 45–56, 2001.

[10] V. H. Quintana and M. Santos-Nieto, "Reactive-power dispatch by successive quadratic programming," *IEEE Transactions on Energy Conversion*, vol. 4, no. 3, pp. 425–435, 1989.

[11] N. Grudinin, "Reactive power optimization using successive quadratic programming method," *IEEE Transactions on Power Systems*, vol. 13, no. 4, pp. 1219–1225, 1998.

[12] F.-C. Lu and Y. Y. Hsu, "Reactive power/voltage control in a distribution substation using dynamic programming," *IEE Proceedings-Generation, Transmission and Distribution*, vol. 142, no. 6, pp. 639–645, 1995.

[13] H. Yoshida, K. Kawata, Y. Fukuyama, S. Takayama, and Y. Nakanishi, "A particle swarm optimization for reactive power and voltage control considering voltage security assessment," *IEEE Transactions on Power Systems*, vol. 15, no. 4, pp. 1232–1239, 2000.

[14] M. A. Abido, "Multiobjective particle swarm optimization for optimal power flow problem," in *Handbook of Swarm Intelligence*, Springer, Berlin, Heidelberg, Germany, 2011.

[15] K. Mahadevan and P. S. Kannan, "Comprehensive learning particle swarm optimization for reactive power dispatch," *Applied Soft Computing*, vol. 10, no. 2, pp. 641–652, 2010.

[16] Z. Sahli, A. Hamouda, A. Bekrar, and D. Trentesaux, "Hybrid PSO-tabu search for the optimal reactive power dispatch problem," in *Proceedings of the IECON 2014-40th Annual Conference of the IEEE Industrial Electronics Society*, Dallas, TX, USA, November 2014.

[17] J. Polprasert, W. Ongsakul, and V. N. Dieu, "Optimal reactive power dispatch using improved pseudo-gradient search particle swarm optimization," *Electric Power Components and Systems*, vol. 44, no. 5, pp. 518–532, 2016.

[18] M. Varadarajan and K. S. Swarup, "Differential evolution approach for optimal reactive power dispatch," *Applied Soft Computing*, vol. 8, no. 4, pp. 1549–1561, 2008.

[19] A. A. A. E. Ela, M. A. Abido, and S. R. Spea, "Differential evolution algorithm for optimal reactive power dispatch," *Electric Power Systems Research*, vol. 81, no. 2, pp. 458–464, 2011.

[20] C.-M. Huang and Y.-C. Huang, "Combined differential evolution algorithm and ant system for optimal reactive power dispatch," *Energy Procedia*, vol. 14, pp. 1238–1243, 2012.

[21] M. Ghasemi, M. M. Ghanbarian, S. Ghavidel, S. Rahmani, and E. Mahboubi Moghaddam, "Modified teaching learning algorithm and double differential evolution algorithm for optimal reactive power dispatch problem: a comparative study," *Information Sciences*, vol. 278, pp. 231–249, 2014.

[22] K. Iba, "Reactive power optimization by genetic algorithm," *IEEE Transactions on Power Systems*, vol. 9, no. 2, pp. 685–692, 1994.

[23] M. S. Kumari and S. Maheswarapu, "Enhanced genetic algorithm based computation technique for multi-objective optimal power flow solution," *International Journal of Electrical Power & Energy Systems*, vol. 32, no. 6, pp. 736–742, 2010.

[24] S. Jeyadevi, S. Baskar, C. K. Babulal, and M. Willjuice Iruthayarajan, "Solving multiobjective optimal reactive power dispatch using modified NSGA-II," *International Journal of Electrical Power & Energy Systems*, vol. 33, no. 2, pp. 219–228, 2011.

[25] P. Subbaraj and P. N. Rajnarayanan, "Optimal reactive power dispatch using self-adaptive real coded genetic algorithm," *Electric Power Systems Research*, vol. 79, no. 2, pp. 374–381, 2009.

[26] S. Duman, Y. Sönmez, U. Güvenç, and N. Yörükeren, "Optimal reactive power dispatch using a gravitational search algorithm," *IET Generation, Transmission & Distribution*, vol. 6, no. 6, pp. 563–576, 2012.

[27] S. Mouassa, T. Bouktir, and A. Salhi, "Ant lion optimizer for solving optimal reactive power dispatch problem in power systems," *Engineering Science and Technology, an International Journal*, vol. 20, no. 3, pp. 885–895, 2017.

[28] B. Mandal and P. K. Roy, "Optimal reactive power dispatch using quasi-oppositional teaching learning based optimization," *International Journal of Electrical Power & Energy Systems*, vol. 53, pp. 123–134, 2013.

[29] C.-x. Guo and B. Zhao, "A pooled-neighbor swarm intelligence approach to optimal reactive power dispatch," *Journal of Zhejiang University-Science A*, vol. 7, no. 4, pp. 615–622, 2006.

[30] A. Khorsandi, A. Alimardani, B. Vahidi, and S. H. Hosseinian, "Hybrid shuffled frog leaping algorithm and Nelder-Mead simplex search for optimal reactive power dispatch," *IET Generation, Transmission & Distribution*, vol. 5, no. 2, pp. 249–256, 2011.

[31] A. Mukherjee and V. Mukherjee, "Solution of optimal reactive power dispatch by Chaotic Krill Herd algorithm," *IET Generation, Transmission & Distribution*, vol. 9, no. 15, pp. 2351–2362, 2015.

[32] K. Rayudu, G. Yesuratnam, and A. Jayalaxmi, "Artificial Bee Colony algorithm for optimal reactive power dispatch to improve voltage stability," in *Proceedings of the 2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT)*, pp. 1–7, Nagercoil, India, March 2016.

[33] A. Rajan and T. Malakar, "Exchange market algorithm based optimum reactive power dispatch," *Applied Soft Computing*, vol. 43, pp. 320–336, 2016.

[34] A. M. Shaheen, R. A. El-Sehiemy, and S. M. Farrag, "Optimal reactive power dispatch using backtracking search algorithm," *Australian Journal of Electrical and Electronics Engineering*, vol. 13, no. 3, pp. 200–210, 2016.

[35] A. H. Khazali and M. Kalantar, "Optimal reactive power dispatch based on harmony search algorithm," *International Journal of Electrical Power & Energy Systems*, vol. 33, no. 3, pp. 684–692, 2011.

[36] H. Salimi, "Stochastic fractal search: a powerful metaheuristic algorithm," *Knowledge-Based Systems*, vol. 75, pp. 1–18, 2015.

[37] H. V. Tran, T. V. Pham, L. H. Pham, N. T. Le, and T. T. Nguyen, "Finding optimal reactive power dispatch solutions by using a novel improved stochastic fractal search optimization algorithm," *Telecommunication Computing Electronics and Control*, vol. 17, no. 5, pp. 2517–2526, 2019.

[38] MATPOWER 4.1 IEEE 30-bus and 118-bus test system, http://www.pserc.cornell.edu/matpower.