Stochastic fractal search based method for economic load dispatch

Ly Huu Pham¹, Thang Trung Nguyen², Lam Duc Pham³*, Nam Hoang Nguyen⁴¹,²
¹,²Power System Optimization Research Group, Faculty of Electrical and Electronics Engineering,
Ton Duc Thang University, Ho Chi Minh City, Vietnam
³Faculty of Mechanical, Electrical, Electronic and Automotive Engineering,
Nguyen Tat Thanh University, Ho Chi Minh City, Vietnam
⁴Modeling Evolutionary Algorithms Simulation and Artificial Intelligence, Faculty of Electrical & Electronics
Engineering, Ton Duc Thang University, Ho Chi Minh City, Vietnam
*Corresponding author, e-mail: phamhuuly@tdtu.edu.vn¹, nguyentrungthang@tdtu.edu.vn², pdlam@ntt.edu.vn³, nguyenhoangnam@tdtu.edu.vn⁴

Abstract

This paper presents a nature-inspired meta-heuristic, called a stochastic fractal search based
method (SFS) for coping with complex economic load dispatch (ELD) problem. Two SFS methods are
introduced in the paper by employing two different random walk generators for diffusion process in which
SFS with Gaussian random walk is called SFS-Gauss and SFS with Levy Flight random walk is called
SFS-Levy. The performance of the two applied methods is investigated comparing results obtained from
three test system. These systems with 6, 10, and 20 units with different objective function forms and
different constraints are inspected. Numerical result comparison can confirm that the applied approach has
better solution quality and fast convergence time when compared with some recently published standard,
modified, and hybrid methods. This elucidates that the two SFS methods are very favorable for solving
the ELD problem.

Keywords: economic load dispatch, power loss, stochastic fractal search, valve point effect

Nomenclature

| Symbol | Description |
|--------|-------------|
| mᵢ, nᵢ, oᵢ | fuel cost coefficients of the ᵢth unit |
| pᵢ, qᵢ | valve point effects coefficients of the ᵢth unit |
| mᵢl, nᵢl, oᵢl | fuel cost coefficients for fuel type  ᵇ of the ᵢth unit |
| pᵢl, qᵢl | valve point effects coefficients for fuel type  ᵇ of the ᵢth unit |
| Cᵢ₀, Cᵢ₀l, Cᵢ₀j | coefficients of transmission power loss matrix |
| PᵢTL | losses of the transmission line |
| Pᵢmax | maximum power output of the ᵢth unit |
| Pᵢmin | minimum power output of the ᵢth unit |
| PᵢMinᵢ | minimum power output for fuel Mi of the ᵢth unit |
| Pᵢ | power output of the ᵢth unit |
| N | total number of generators |
| PSLD | total system load demand |

1. Introduction

Economic load dispatch (ELD) problem is becoming more important in power system
operation and control. The prime objective of the ELD problem is to minimize the total fuel cost
by economically distributing power of generating units to electric load. In addition, load demand,
all physical and operational constraints are required to be within predetermined bounds.
In traditional ELD problem, a fuel cost function of generators is considered as the single
quadratic cost function with linear constraint [1]. In practical, realistic ELD problem must take
complex and nonlinear characteristic with many equality and inequality constraints into account
to provide the completeness for the ELD problem formulation.

Received March 2, 2019; Revised April 12, 2019; Accepted May 9, 2019
Thus, fuel cost curve of thermal units should be presented as non-smooth presented form, a piecewise function when thermal units are supplied by multi-fuel sources like coal, natural gas, and oil [2]. Also, to get more precise cost model, the valve-point effects and the prohibited zones must also be taken consideration [3, 4]. The complexity of the problem dramatically increases once both multi-fuel option and valve-point effects are considered simultaneously. Generally, the ELD problem on later is more and more difficult when taking many power systems and generator constraints into account. Over the past decades, there were many applied methods with the task of solving ELD problem such as Lambda Iteration method [5], Dynamic Programming (DP) [6], Gradient Method [7], Lagrangian Relaxation algorithm [8], and Hopfield neural network based numerical method (HNNNM) [9]. For the classical methods above, parameters of these algorithms are surveyed and selected after many trial run times, which helps to find a global solution in a short time. However, the process of setting parameters takes much time. As complex ELD problem has non-convex features and various nonlinear constraints, the mentioned classical methods cannot afford to handle and result in low convergence. A series of novel methodologies have been born called meta-heuristic method to deal with these disadvantages such as Genetic algorithm (GA) [10], Firefly algorithm [11], Particle Swarm Optimization (PSO) [12], Differential Evolution (DE) algorithm [13], Anti-predatory particle swarm optimization (APPSO) [14], Biogeography-Based Optimization (BBO) [15], and Ant Lion algorithm (ALO) [16]. Because of their outstanding characteristics, such meta-heuristic methods proved their efficiency for solving the aforementioned difficulties. Consequently, meta-heuristic methods have been received much more curiosity by researchers. Besides, a large number of scientists in many engineering fields have been constantly strived and selected the strong points of methods to modify/improve them into the promising methods such as Colonial Competitive Differential Evolution (CCDE) [17], Efficient Real-Coded Genetic algorithm (ERCGA) [18], Improved Real-Coded Genetic algorithm (IRCGA) [19], and Modified Cuckoo Search algorithm (MCSA) [20]. As known, obtained results of the meta-heuristic family are better than that of the standard methods although they may still exist some weaknesses. Hence, improving meta-heuristic ones is the expectation of researchers with the goal of finding the best solution quality in exploring and exploiting search space effectively.

In addition, the combination between two or more methods is also known as a unique way to create powerful hybrid algorithms such as Genetic Algorithm with an ant colony approach (GAAPI) [21], Particle Swarm Optimization based Differential Evolution (PSODE) [22], Distributed Sobol Particle Swarm Optimization and Tabu Search algorithm (DSPSO-TSA) [23], Differential Evolution-Particle Swarm Optimization-Differential Evolution (DPD) [24], Biogeography-Based Optimization, and modified Differential Evolution (aBBOMDE) [25]. In recent years, various optimization methods have been successfully applied to deal with the realistic ELD problem in large-scale power system including Crisscross Optimization (CSO) [26], Dimensional Steepest Decline method (DSD) [27], an Improved Orthogonal Design Particle Swarm Optimization (AIODPSO) algorithm [28], Double Weighted Particle Swarm Optimization (DWPSO) [29], and Modified Crow Search algorithm (MCSA) [30].

In this paper, a nature-inspired Stochastic Fractal Search (SFS) algorithm is applied to determine the minimum cost of the ELD problem. SFS was first recommended by Salimi [31] and applied to optimize twenty-three benchmark functions with a quite good solution quality. In the paper, our purpose is to investigate the efficacy and robustness of the SFS method on various standard IEEE systems through using two different random walk generators for diffusion process. Firstly, SFS with Gaussian random walk is called SFS-Gauss and secondly, SFS with Levy Flight random walk is called SFS-Levy. In addition, the achievement of SFS method has also competed against other ones available in the literature.

2. Problem Formulation
2.1. Formulation of the Smooth ELD Problem

The traditional fuel cost function is often represented as a single quadratic polynomial function polynomial function in (1):

$$F_i(P_i) = a_i P_i^2 + b_i P_i + c_i$$

(1)
the (1) can indicate that the fuel cost for each MWh is different for different power output of thermal generating unit. Thus, the major target of the ELD problem is to reduce the total fuel cost of all thermal generating units and it can be described as the following model:

$$\min F = \sum_{i=1}^{N} F_i(P_i)$$

(2)

2.2. Formulation of the Non-smooth ELD Problem

2.2.1. ELD Problem Considering Valve-point Effects

In the practical power system, thermal units often use many valve for adjusting their power output. This makes the fuel cost function become discontinuous form as shown in (3).

$$F(P_i) = \frac{1}{2}P_i^2 + n_iP_i + m_i + |\frac{1}{2}P_i^2 - P_i^\min|$$

(3)

2.2.2. ELD Problem Considering Multi-fuel Options

Since the generators are supplied by various fuel sources such as coal, natural gas, oil etc., the total fuel cost function of each unit can be represented by a piecewise quadratic cost function as follows:

$$F_i(P_i) = \begin{cases} \frac{1}{2}P_i^2 + n_{1i}P_i + m_{1i}, & \text{for fuel 1, } P_i^\min \leq P_i \leq P_i^\max_{1i} \\ \frac{1}{2}P_i^2 + n_{2i}P_i + m_{2i}, & \text{for fuel 2, } P_i^\min \leq P_i \leq P_i^\max_{2i} \\ K \left[ n_{Mi}P_i^2 + n_{Mi}P_i + m_{Mi}, & \text{for fuel } Mi, P_i^\min \leq P_i \leq P_i^\max_{Mi} \right] \end{cases}$$

(4)

2.2.3. ELD Problem Considering Both Valve-point Effects and Multiple Fuel Options

The ELD problem will be practical and more accurate if both valve-point effects and multiple fuel options are considered as the following [22].

$$F_i(P_i) = \begin{cases} \frac{1}{2}P_i^2 + n_{1i}P_i + m_{1i} + \frac{1}{2}q_1(P_i^\min - P_i), & \text{for fuel 1, } P_i^\min \leq P_i \leq P_i^\max_{1i} \\ \frac{1}{2}P_i^2 + n_{2i}P_i + m_{2i} + \frac{1}{2}q_2(P_i^\min - P_i), & \text{for fuel 2, } P_i^\min \leq P_i \leq P_i^\max_{2i} \\ K \left[ n_{Mi}P_i^2 + n_{Mi}P_i + m_{Mi} + \frac{1}{2}q_{Mi}(P_i^\min - P_i), & \text{for fuel } Mi, P_i^\min \leq P_i \leq P_i^\max_{Mi} \right] \end{cases}$$

(5)

2.3. Constrains

2.3.1. Generating Capacity Limit

A real power output of units is generated that must be lied in the range of their lower and their upper limit as:

$$P_i^\min \leq P_i \leq P_i^\max$$

(6)

2.3.2. Power Balance Constraint

The formula of the generator power balance constraint with considering the total transmission power losses are presented by:

$$\sum_{i=1}^{N} (P_i - P_{SLD} - P_{TTL}) = 0$$

(7)

where $P_L$ is calculated by the Kron’s loss formula expressed as:
\[ P_L = \sum_{i=1}^{N} \sum_{j=1}^{N} P_{C_{ij}} P_j + \sum_{i=1}^{N} C_{i0} P_i + BC_{00} \quad (8) \]

3. Stochastic Fractal Search

The SFS algorithm that was formulated by Salimi in 2014, is a variant of Fractal Search by adding two update processes. So, the structure of SFS comprises of three update phases such as diffusion phase, the first update phase and the second update phase. As results, three new solution generations are created by SFS in each iteration. In SFS, the task of diffusion phase is to find solutions in small search space whilst the task of two update phases is to search solutions in large search space. Basically, SFS has a population corresponding to the number of points where each point \( Y_d \) is represented as an optimal solution \( d = 1, N_p \). At the beginning, all the points are randomly created and their fitness function are calculated to find the best solution \( Y_{best} \) among all solutions in the population. Then, SFS continues to perform the iterative search process with three phases above. The detail of three phase is described as follows:

3.1. Diffusion Phase

Based on the previous points, the first new solutions are produced by using one of two random walks as Levy flight and Gaussian. In this phase, each solution (point) \( Y_d \) diffuses around its position into a number of new diffusion solutions \( Y_{di} \) where \( di = 1, \ldots, N_{df} \). The diffusion can be mathematically formulated as follows:

3.1.1. Diffusion Phase with Levy Flight

The equation of the diffusion phase using Levy flight random walk is performed in (9):

\[ Y_{d\text{Levy}} = Y_d + \alpha \times \epsilon \times \Delta Y_{d\text{Levy}} \quad (9) \]

where \( \alpha > 0 \) is scale factor; \( \epsilon \) is a normally distributed random numbers restricted to (0,1); \( Y_d \) denoted the \( d \)th solution in the current population and \( \Delta Y_{d\text{Levy}} \) is described by [20]:

\[ \Delta Y_{d\text{Levy}} = v \times \frac{\omega_x (\lambda)}{\omega_y (\lambda)} (Y_d - Y_{best}) \quad (10) \]

\[ v = \frac{\text{rand}_x}{\left[ \text{rand}_y \right]^{1/\lambda}} \quad (11) \]

where \( \text{rand}_x \) and \( \text{rand}_y \) are two normally distributed stochastic variables.

3.1.2. Diffusion Phase with Gaussian Walk

The process of creating solution following Gaussian random walk is performed by:

\[ Y_{d\text{Gau}} = b_d \times GW_{d1} + (1 - b_d) \times GW_{d2} \quad (12) \]

where \( b_d \) is a binary number (0 or 1) dependent on comparison of a random number \( \text{rand}_d \) and walk factor \( w \) (0≤\( w \leq 1 \)) as follows:

\[ b_d = \begin{cases} 1 & \text{if } \text{rand}_d < w \\ 0 & \text{otherwise} \end{cases} \quad (13) \]

and one out of \( GW_{d1} \) and \( GW_{d2} \) is used to create solutions, described (14):

\[ GW_{di} = \text{normrnd}(Y_{best}, \sigma_d) + \delta \times (Y_{best} - Y_d) \quad (14) \]
\[ GW_{d2} = \text{normrnd}(Y_d, \sigma_d) \]  

(15)

where \( \delta \) is uniformly distributed random numbers; \( \sigma_d \) is the standard deviation.

At the end of creating the new solutions, all new solutions are appraised by computing the fitness function and then the evaluation between each old solution and \( N_d \) new solutions at each point is done in order to retain a better solution with the best fitness, named \( Y_d \).

### 3.2. The First Update Phase

First of all, all current points are assigned to a value of probability \( P_{d}\alpha \), which is determined by:

\[ P_{d\alpha} = \frac{\text{Rank}_d}{N_p} \]  

(16)

according to (16), the point with the best fitness has the highest probability and ranks at the last position otherwise stands at the first position. After ranking for all points, each point \( Y_d \) in group is updated by the comparison of the probability \( P_{d\alpha} \) and a random number \( \alpha_1 \) (0<\( \alpha_1 \)<1). If \( P_{d\alpha} < \alpha_1 \), the \( d \)th point is updated like (17), otherwise it doesn’t remain changed.

\[ Y_{new}^{d} = Y_{1} - \text{rand} \times (Y_{2} - Y_{d}) \]  

(17)

where \( Y_{d}^{new} \) the new modified position of \( Y_d \); \( Y_{1} \) and \( Y_{2} \) is symbolize randomly selected points in the group. Through the first update phase, it is easily seen that all of points with a bad quality are often updated while the points with a better quality have low possibility to be newly updated. After performing the second generation, once again, mechanism of the comparison is re-performed to select the better solution between old solution and new solution at each point, named \( Y_d \).

### 3.3. The Second Update Phase

Similar to the first update stage, the first step in the second update stage is also to determine \( \text{rank}_d \) and \( P_{d\alpha} \) for each solution \( d \) and then \( P_{d\alpha} \) is compared to a random number within 0 and 1 for determining if the solution is newly updated. In case that considered solution \( d \) is accepted to be newly updated, there are two models to be used as follows:

\[ Y_{d2}^{new} = Y_{d} - \varepsilon' \times (Y_{1} - Y_{best}) \quad \text{if} \quad \text{rand}_d \leq 0.5 \]  

(18)

\[ Y_{d4}^{new} = Y_{d} + \varepsilon' \times (Y_{i} - Y_{d}) \quad \text{if} \quad \text{rand}_d > 0.5 \]  

(19)

where random selected points \( Y_{3}, Y_{4} \) and the best point \( Y_{best} \) are obtained from the first phase; \( \varepsilon' \) is random number in the range (0,1).

### 4. Implementation of SFS for Solving ELD Problem

#### 4.1. Constraint Violation Handling Technique

A punishment function technique is employed in the ELD problem to deal with constraint violations by using two variable types such as dependent variable and control variable. From (7), \( P_{1d} \) corresponding to power output of the \( 1^{st} \) unit of the \( d^{th} \) solution is selected to be a dependent variable. Other variables from \( P_{2d} \) to \( P_{Nd} \) are control variables and included in each solution \( d \). Thus, these control variables are supposed to be given and then the dependent variable needs to be determined. The value of \( P_{1d} \) obtained has no assurance that satisfies its limit power as (6). Therefore, the violation of the dependent variable must be penalized if one occurs and is calculated by:
\[
PUN_d = \begin{cases} 
    P_{\text{dmin}} - P_{\text{id}} & \text{if } P_{\text{dmax}} > P_{\text{id}} \\
    P_{\text{id}} - P_{\text{dmax}} & \text{if } P_{\text{id}} < P_{\text{dmin}} \\
    0 & \text{if } P_{\text{dmin}} \leq P_{\text{id}} \leq P_{\text{dmax}} 
\end{cases}
\]  \tag{20}

where \(PUN_d\) is the violation of solution \(d\).

Thus, in order to optimize the ELD problem related to the cost function (1)-(5), punishment function \(PUN_d\) must be considered in fitness function in the case of occurring violation of value of \(P_{\text{id}}\). Accordingly, the fitness function is determined as the following (21):

\[
F_d = \sum_{i=1}^{N} F_i(P_i) + K \times (PUN_d)^2
\]  \tag{21}

where \(K\) is factor for handling constraint violation.

4.2. The Detail of SFS’s Procedure of the ELD Problem

Step 1: Set parameters including the number of population \(N_p\), the number of diffusion population \(N_{df}\) and the maximum number of iterations \(MI\).

Step 2: Initializing population \(Y_d\) (\(d=1, \ldots, N_p\)). The maximum and minimum of each point are \(Y_{\text{max}} = [P_{\text{dmax}}]\) and \(Y_{\text{min}} = [P_{\text{dmin}}]\) where \(i=2, \ldots, N\). Thus, each point \(Y_d\) is randomly initialized based on the constraint: \(Y_{\text{min}} \leq Y_d \leq Y_{\text{max}}\).

Step 3: Calculate fitness function \(F_d\) following (21) and find the best point \(Y_{\text{best}}\) in group.

- Set \(\text{Iter}_{\text{cur}} = 1\).

Step 4:
- The diffusion phase is executed by using either Levy Flight or Gaussian walk.
- Check bounds for new solutions and correct them if violated.
- Calculate fitness function.
- Compare old solution and new solutions at each point to keep the best one, called \(Y_d\).

Step 5:
- The new solutions are produced by using the first update phase.
- Check bounds for new solutions and correct them if violated.
- Calculate fitness function.
- Compare old solution and new solutions at each point to keep better one, called \(Y_d\).
- Select the current best solution in group.

Step 6:
- The new solutions are produced by using the second update phase.
- Check bounds for new solutions and correct them if violated.
- Calculate fitness function.
- Compare old solution and new solutions at each point to keep better one.

Step 7: Save the best point \(Y_{\text{best}}\) for the current iteration.

Step 8: Check stopping condition.

If \(\text{Iter}_{\text{cur}} < MI\), \(\text{Iter}_{\text{cur}} = \text{Iter}_{\text{cur}} + 1\) and back to step 4. Otherwise, stop the procedure.

5. Numerical Results

In this section, we present two issues as follows: 1) Analysis of the efficiency of the SFS method based on the simulation results applying Levy Flight or Gauss walk for the diffusion phase; 2) Comparing results from three various standard IEEE test systems with 6 units, 10 units and 20 units to evaluate performance of SFS methods. Three test systems are solved by running SFS on Matlab 2016 a and a computer with 2.4 GHz processor and 4 GB of RAM.

5.1. Applying Levy Flight or Gaussian Random Walk for the Diffusion Phase

In [31], author described that the diffusion phase of SFS could use Levy Flight or Gaussian random walk. However, all applications for solving benchmark functions, Gaussian random walk was only selected. In order to fully investigate the characteristics of SFS,
we examine its characteristics via applying Levy Flight or Gauss walk in the diffusion phase via three test systems. They comprise of 6-unit test system considering with and without line transmission losses, 10-unit test system with both multi fuels and valve point effect, and 20 unit test system with transmission losses. Such investigation process of using Levy Flight or Gauss walk has a very important role because it helps researchers easily to know the effectiveness of SFS to application for different systems. This investigation will be implemented based on two cases including the influence of walk factor $\omega$ on the obtained results of SFS_Gauss and impact of the number of population and the number of iterations on the results of SFS_Gauss and SFS_Levy.

5.1.1. Survey 1: the Influence of the Walk Factor

For the first case, walk factor of SFS_Gauss is set from 0 to 1 with a step of 0.25 to analyze its impact on the tested results from convex or non-convex test systems. If $\omega$ is select to 0, (15) is used to create the new solutions in the diffusion phase. If is select to 1, (14) is employed to produce the new solutions. Otherwise, both. (14-15) are utilized for the solution creating process. To see the changes clearly, some parameters like the number of populations and number of iterations need to be established suitably for different test systems. Particularly, the number of populations and number of iterations are respectively set to 5 and 30 for 6-unit system, 10 and 50 for 20-unit system, and 20 and 500 for 10-unit system. The obtained results from these systems are summarized in Tables 1, 2 and 3.

As shown in these tables, when the value of is varied from 0 to 1 with the step size of 0.25, the minimum costs of various three test systems decreased from a high value to a low value. Specifically, for 6-unit system without transmission losses, the minimum costs are respectively 31446.8981 $/h, 36003.5396 $/h and 40679.0466 $/h for cases 1.1, 1.2, and 1.3 corresponding to $\omega=0$. These costs could be minimized and equal to 31445.6233 $/h, 36003.1278 $/h, and 40675.9824 $/h as setting $\omega=1$. Similarly, when the value of is 0, costs of 10-unit system and 20-unit system are 31446.8981$/h and 62460.87 $/h, respectively. When the value of is 1, those of 10-unit and 20-unit test systems are 331445.6233 $/h and 62456.91 $/h, respectively. From such analysis, it points out that (14) has better performance than (15) on result obtained from the method. Furthermore, if only (14) is applied, the obtained results are the most effective.

Table 1. The Obtained Results from 6-Unit System without Transmission Losses with Different $\omega$

| $\omega$ | Case 1.1: $P_D=600W$ | Case 1.2: $P_D=700W$ | Case 1.3: $P_D=800W$
|---------|------------------|------------------|------------------|
| 0       | Min. cost ($/h$) | Min. cost ($/h$) | Min. cost ($/h$) |
| 31446.8981 | 36003.5396 | 40679.0466 |
| 0.25    | 31445.6404      | 36003.2043      | 40676.0080      |
| 0.5     | 31445.6400      | 36003.2040      | 40676.0200      |
| 0.75    | 31445.6270      | 36003.2046      | 40675.9860      |
| 1       | 31445.6233      | 36003.1279      | 40675.9824      |

Table 2. The Obtained Results from 10-Unit System with Multi Fuels and Valve Point Effect with Different $\omega$

| $\omega$ | Min. cost ($/h$) | Aver. cost ($/h$) | Max. cost ($/h$) |
|---------|------------------|------------------|------------------|
| 0       | 623.8911         | 624.4027         | 626.6973         |
| 0.25    | 623.8327         | 624.0381         | 626.381          |
| 0.5     | 623.8386         | 623.981          | 626.398          |
| 0.75    | 623.8287         | 624.0505         | 626.4194         |
| 1       | 623.8274         | 624.1806         | 631.1825         |

Table 3. The Obtained Results from 20-Unit System with Transmission Losses with Different $\omega$

| $\omega$ | Min. cost ($/h$) | Aver. cost ($/h$) | Max. cost ($/h$) |
|---------|------------------|------------------|------------------|
| 0       | 62460.87         | 62477.9943       | 62504.9535       |
| 0.25    | 62457.05         | 62459.1811       | 62484.3307       |
| 0.5     | 62457.06         | 62458.7540       | 62473.4330       |
| 0.75    | 62456.95         | 62459.0491       | 62466.2881       |
| 1       | 62456.91         | 62460.9454       | 62487.6297       |
5.1.2. Survey 2: the Impact of the Number of Population and Number of Iterations on Obtained Results

For the second survey, we scrutinize the impact of \( N_p \) and \( MI \) on the results of SFS_Gauss and SFS_Levy for 10-unit and 20-unit test system. \( N_p \) is fixed and chosen to be 20 for 10-unit system and 10 for 20-unit system while \( MI \) is altered from 100 to 550 for the former, and from 50 to 250 for the latter. Moreover, according to the survey 1, SFS_Gauss had good solutions if only (14) is used for producing new solutions. So, we only use (14) corresponding to \( \omega=1 \). And statistical results from 10-unit system and 20-unit system are shown in Tables 4 and 5.

In accordance with Tables 4 and 5, the minimum cost of SFS_Gauss and SFS_Levy are more and more changeable when \( MI \) is changed. For the case with non-smooth objective function, the best cost of SFS-Gauss and SFS-Flevy are 623.8252 $/h and 623.8235 $/h, respectively. For the case with smooth objective function, those of SFS-Gauss and SFS-Flevy are 62456.6331 $/h and 62456.6338 ($/h), respectively. It is clearly recognized that the best cost of SFS-Gauss is always better than that of SFS-Flevy at the same number of iterations for 20-unit system. In contrast to the case above, the best cost of SFS-Flevy outperforms than that of SFS-Gauss with the same manner for 10-unit system.

This implies that SFS_Levy is suitable for solving non-convex economic load dispatch problem with many local optimum solutions because its strong characteristic is to search solutions in large space, while SFS_Gauss is appropriate for disentangling convex one as it is powerfully capable for finding solutions in small space.

### Table 4. Statistical Results of Survey 2 for 10-Unit System

| Min. cost ($/h) | SFS_Gauss | SFS_Levy | \( N_p \) | \( MI \) | SFS_Gauss | SFS_Levy | \( N_p \) | \( MI \) |
|----------------|-----------|-----------|---------|--------|-----------|-----------|---------|--------|
| 623.9072       | 623.9348  | 20        | 100     | 623.8360| 623.8264  | 20        | 350     |
| 623.8474       | 623.8888  | 20        | 150     | 623.8340| 623.8272  | 20        | 400     |
| 623.8340       | 623.8442  | 20        | 200     | 623.8270| 623.8285  | 20        | 450     |
| 623.8318       | 623.8316  | 20        | 250     | 623.8252| 623.8240  | 20        | 500     |
| 623.8269       | 623.8279  | 20        | 300     | 623.8293| 623.8235  | 20        | 550     |

### Table 5. Statistical Results of Survey 2 for 20-Unit System

| Min. cost ($/h) | SFS_Gauss | SFS_Levy | \( N_p \) | \( MI \) |
|----------------|-----------|-----------|---------|--------|
| 62456.7717     | 62458.3038| 10        | 50      |
| 62456.6343     | 62456.6841| 10        | 100     |
| 62456.6331     | 62456.6338| 10        | 150     |
| 62456.6331     | 62456.6331| 10        | 200     |
| 62456.6331     | 62456.6331| 10        | 250     |

5.2. Comparison and Discussion

In section, the SFS method performance is evaluated by comparing the minimum costs with other available methods. For fair comparison, some parameters such as \( N_p \) and \( MI \) along with the number of function evaluations \( Fes \) are also reported in tables.

5.2.1. Case Study 1: 6-Unit Test System

This study solves 6-generating unit taking with or without line transmission losses into account. Load demand level of 600, 700, and 800 MVA in turn for both test circumstances are scrutinized. Problem data for various load demand levels of the first test system can be reached in Moustafa et al. [11]. In such study, we set \( N_p \) to 10 and \( MI \) to 50 for testing all the cases with or without transmission losses. Tables 6 and 7 report the numerical results achieved by FFA [11], MFA [11], VSSFA [11], MFFA [11], SFS_Gauss and SFS_Levy.

From Table 6, it can be seen that minimum fuel and standard cost values attained by SFS_Gauss and SFS_Levy are much lower than those of other methods. In addition, SFS_Gauss, and SFS_Levy only use \( N_p=10 \), \( MI=50 \) and \( Fes=1500 \) while other ones need to employ \( N_p=50 \) \( MI=150 \) and \( Fes=3750 \). It easily confirms that SFS_Gauss and SFS_Levy are faster than those of variants of FA. Even as with the case including transmission losses.
exhibited in Table 7, SFS_Gauss and SFS_Levy still display its supremacy about the best cost with the shortest execution time. Consequently, it can clinch that SFS_Gauss and SFS_Levy are the best methods for these cases.

Table 6. Numerical Analysis for the 6-Unit Test System without Transmission Losses

| Methods     | Case 1.1: P_D=600W | Case 1.2: P_D=700W | Case 1.3: P_D=800W | Np | Ml | Fes |
|-------------|---------------------|---------------------|---------------------|----|----|-----|
| FFA [11]    | 31489               | 243.8401            | 36075               | 243.8401 | 40739 | 121.8807 | 25 | 150 | 3750 |
| MFA [11]    | 31447               | 2.928535            | 36006               | 2.928535 | 40676 | 2.696777 | 25 | 150 | 3750 |
| VSSFA [11]  | 31576               | 244.0893            | 36036               | 244.0893 | 40701 | 77.10171 | 25 | 150 | 3750 |
| MFFA [11]   | 31481               | 95.84878            | 36021               | 95.84878 | 40740 | 110.8561 | 25 | 150 | 3750 |
| SFS-Gauss   | 31445.62            | 0.005524            | 36003.12            | 0.005524 | 40675.97 | 0.057181 | 10 | 50  | 1500 |
| SFS-Levy    | 31445.62            | 0.146154            | 36003.12            | 0.146154 | 40675.97 | 0.02478  | 10 | 50  | 1500 |

Table 7. Numerical Analysis for the 6-Unit Test System with Transmission Losses

| Methods     | Case 1.4: P_D=600W | Case 1.5: P_D=700W | Case 1.6: P_D=800W | Np | Ml | Fes |
|-------------|---------------------|---------------------|---------------------|----|----|-----|
| FFA [11]    | 32122               | 159.5433            | 37004               | 186.8881 | 41939 | 167.7257 | 25 | 150 | 3750 |
| MFA [11]    | 32098               | 4.706938            | 36914               | 3.322966 | 41898 | 2.347077 | 25 | 150 | 3750 |
| VSSFA [11]  | 32159               | 159.4889            | 36960               | 96.77619 | 41976 | 68.85434 | 25 | 150 | 3750 |
| MFFA [11]   | 32109               | 103.4451            | 36978               | 35.98293 | 41930 | 33.53627 | 25 | 150 | 3750 |
| SFS-Gauss   | 32094.68            | 0.000006            | 36912.14            | 0.000138 | 41896.63 | 0.000563 | 10 | 50  | 1500 |
| SFS-Levy    | 32094.68            | 0.013828            | 36912.14            | 0.057937 | 41896.63 | 0.020958 | 10 | 50  | 1500 |

5.2.2. Case Study 2: 10-unit Test System

This portion applied 10-unit system with valve-point loading, multiple fuel options and without transmission to size up the real performance of the SFS on non-convex problem. The data such as upper and lower powers of the units and fuel cost coefficients are come from the previous study as in [17, 23]. In such study, the load of all thermal units is 2700 MW.

Table 8 describes the comparison of results performed by SFS method and other thirteen ones in terms of minimum cost, population, the maximum iterations, and the number of function evaluations. As seen from the table, the best fuel cost of DWPSO [29] is 622.74 $/h and is better than those of other methods. After rechecking this value by substituting the optimum dispatch solutions of DWPSO into function objective, the exact cost is 624.23 $/h.

Table 8. Comparison of Results in Case Study 2

| Methods     | Min. cost ($/h) | Np  | Ml  | Fes  | Methods     | Min. cost ($/h) | Np  | Ml  | Fes  |
|-------------|-----------------|-----|-----|------|-------------|-----------------|-----|-----|------|
| APPSO(1) [14] | 624.16         | 20  | 200 | 4.000 | CSO [26]    | 623.82          | 100 | 1000 | 100.000 |
| APPSO(2) [14] | 624.01         | 20  | 200 | 4.000 | DSD [27]    | 623.83          | -   | -   | -    |
| CCDE [17]    | 623.83         | 35  | 200 | 7.000 | AIODPSO-Global [28] | 623.83 | 40 | -   | 15.000 |
| ERCGA [18]   | 623.83         | -   | -   | -    | AIODPSO-Local [28] | 623.83 | 40 | -   | 15.000 |
| IRCGA [19]   | 623.83         | -   | -   | -    | DWPSO [29]  | 622.74         | 200 | 1000 | 200.000 |
| PSODE [22]   | 623.83         | 50  | 500 | 25.000 | MCIA [30]   | 623.83         | 100 | 1000 | 10.000 |
| DSPS-TSA[23] | 623.83         | 30  | 100 | 3.000 | SFS-Gauss   | 623.8252       | 20  | 500  | 30.000 |
| DPD [24]     | 623.83         | 99  | 250 | 24.750 | SFS-Levy    | 623.8240       | 20  | 500  | 30.000 |

It is clearly seen that such cost is higher than that reported in Samir Sayah [29]. For this aspect, 623.8252 $/h and 623.8240 $/h are the best costs of SFS-Gauss and SFS-Levy and these values are approximate or smaller than those of other methods. If considering the number of function evaluations, we see that the Fes value of almost all methods is completely different. The Fes value of SFS_Gauss and SFS_Levy is 30.000 which is smaller than that of CSO [26] and DWPSO [29] and higher than those of remaining ones. Moreover, the convergence features of SFS_Gauss and SFS_Levy illustrated in Figure 1 reveal that SFS-Gauss converges to local optimal solutions faster than SFS-Levy from the 1st iteration to the 150th iteration, and then its...
fitness does not change much to the 500\textsuperscript{th} iteration. Otherwise, SFS-Levy can decreases fuel cost gradually from the 280\textsuperscript{th} iteration to the last one. Clearly, SFS-Levy is more effective for nonconvex objective function problem.

![Graph showing convergence curves of SFS-Gauss and SFS-Levy](image)

**Figure 1.** The convergence curves of SFS_Gauss and SFS_Levy in case study 2

### 5.2.3. Case Study 3: 20-Unit Test System

In the study, the twenty generating units with transmission line losses has been deliberated to value the effectiveness of SFS_Gauss and SFS_Levy for clinching the ELD problem. The input data for the case is accessible from [16] and the total load demand is 2500 MW. The result comparison summary is presented in Table 9.

| Methods          | Min. cost ($/h) | \( N_p \) | \( M_I \) | \( Fes \) |
|------------------|-----------------|-----------|------------|----------|
| BBO [15]         | 62456.793       | 50        | 400        | 20000    |
| ALO [16]         | 62456.633       | 30        | 500        | 15000    |
| MCSA [20]        | 62456.633       | 10        | 500        | 10000    |
| aBBOMDE [25]     | 62456.701       | -         | -          | 35000    |
| SFS-Gauss        | 62456.633       | 10        | 150        | 4500     |
| SFS-Levy         | 62455.634       | 10        | 150        | 4500     |

The best fuel cost achieved by SFS-Gauss is 62456.633 $/h, which is better than that of SFS-Levy. In Table 9, it is also seen that SFS_Gauss and SFS_Levy completely devastate other ones such as BBO [15] and aBBOMDE [25] but share the standing position with ALO [16] and MCSA [20]. Besides, SFS-Gauss and SFS-Levy only use \( Fes=4500 \) and the value is smaller than that of the four mentioned methods. So, this underlines that SFS_Gauss and SFS_Levy are very favorable tool for this case. Figure 2 delineates the convergence features of SFS algorithms. As perceived in the figure, SFS-Gauss can improve the fuel cost significant from the 115\textsuperscript{th} iteration to the last one but SFS_Levy just reduce gradually.
6. Conclusions

In this paper, efficient optimum solutions of the ELD problem have been uncovered by executing two forms of the stochastic fractal search algorithm. The focal contributions presented in the paper can be encapsulated as the following aspects:

- Analyze the performance of the SFS method when applying two distribution random walk in the diffusion phase which is one out of three basic mechanisms of the original SFS method.
- Point out the selection of the most effective formula for producing new SFS solutions using Gauss walk.
- Reveal the most suitable form of the SFS for applying the non-smooth or smooth ELD problem.

Moreover, the SFS methods have been investigated via three case studies with different objective function forms and different constraints. The results obtained by the SFS method has been compared to those of other existing techniques. These comparative results show that the SFS method is an effective optimization tool for addressing convex or non-convex ELD problem in a power system.

Reference

[1] Wood AJ, Wollenberg BF. Power generation, operation, and control. John Wiley & Sons. 2012.
[2] Vo DN, Ongsakul W. Economic dispatch with multiple fuel types by enhanced augmented Lagrange Hopfield network. Applied energy. 2012; 91(1): 281-289.
[3] Mandal B, Roy PK, Mandal S. Economic load dispatch using krill herd algorithm. International journal of electrical power & energy systems. 2014; 57: 1-10.
[4] Selvakumar Al, Thanushkodi K. A new particle swarm optimization solution to nonconvex economic dispatch problems. IEEE transactions on power systems. 2007; 22(1): 42-51.
[5] Chen CL, Wang SC. Branch-and-bound scheduling for thermal generating units. IEEE transactions on energy conversion. 1993; 8(2): 184-189.
[6] Liang ZX, Glover JD. A zoom feature for a dynamic programming solution to economic dispatch including transmission losses. IEEE Transactions on Power Systems. 1992; 7(2): 544-550.
[7] Dodu JC, Martin P, Merlin A, Pouget J. An optimal formulation and solution of short-range operating problems for a power system with flow constraints. Proceedings of the IEEE. 1972; 60(1): 54-63.
[8] Bard JF. Short-term scheduling of thermal-electric generators using Lagrangian relaxation. Operations Research. 1988; 36(5): 756-766.
[9] Park JH, Kim YS, Eom IK, Lee KY. Economic load dispatch for piecewise quadratic cost function using Hopfield neural network. IEEE transactions on power systems. 1993; 8(3): 1030-1038.
[10] Walters DC, Sheble GB. Genetic algorithm solution of economic dispatch with valve point loading. *IEEE transactions on Power Systems*. 1993; 8(3): 1325-1332.

[11] Moustafa FS, El-Rafei A, Badra NM. Abdelaziz A Y. Application and performance comparison of variants of the firefly algorithm to the economic load dispatch problem. In Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB), 2017 Third International Conference on IEEE. 2017: 147-151.

[12] Kumar MJ, Singh MHD. Relevance of Particle Swarm Optimization Technique for the Solution of Economic Load Dispatch Problem. 2017; 4(11): 633-639.

[13] Noman N, Iba H. Differential evolution for economic load dispatch problems. Electric Power Systems Research. 2008; 78(8): 1322-1331.

[14] Selvakumar AI, Thanushkodi K. Anti-predatory particle swarm optimization: solution to nonconvex economic dispatch problems. *Electric Power Systems Research*. 2008; 78(1): 2-10.

[15] Bhattacharya A, Chattopadhyay PK. Biogeography-based optimization for different economic load dispatch problems. *IEEE transactions on power systems*. 2010; 25(2): 1064-1077.

[16] Nischal MM, Mehta S. Optimal load dispatch using ant lion optimization. *Int. J Eng Res Appl*. 2015; 5(8): 10-19.

[17] Ghasemi M, Taghizadeh M, Ghavidel S, Abbasian A. Colonial competitive differential evolution: an experimental study for optimal economic load dispatch. *Applied Soft Computing*. 2016; 40: 342-363.

[18] Amjadi N, Nasiri-Rad H. Economic dispatch using an efficient real-coded genetic algorithm. *IET generation, transmission & distribution*. 2009; 3(3): 266-278.

[19] Amjadi N, Nasiri-Rad H. Solution of nonconvex and nonsmooth economic dispatch by a new adaptive real coded genetic algorithm. *Expert Systems with Applications*. 2010; 37(7): 5239-5245.

[20] Nguyen TT, Vo DN. The application of one rank cuckoo search algorithm for solving economic load dispatch problems. *Applied Soft Computing*. 2015; 37: 763-773.

[21] Ciornei I, Kyriakides E. A GA-API solution for the economic dispatch of generation in power system operation. *IEEE Transactions on power systems*. 2012; 27(1): 233-242.

[22] Sayah S, Hamouda A. A hybrid differential evolution algorithm based on particle swarm optimization for nonconvex economic dispatch problems. *Applied Soft Computing*. 2013; 13(4): 1608-1619.

[23] Khamsawang S, Jiriwibhakom S. DSPSO–TSA for economic dispatch problem with non-smooth and noncontinuous cost functions. *Energy Conversion and Management*. 2010; 51(2): 365-375.

[24] Parouha RP, Das KN. A novel hybrid optimizer for solving Economic Load Dispatch problem. *International Journal of Electrical Power & Energy Systems*. 2016; 78: 108-126.

[25] Lohokare MR, Panigrahi BK, Pattnaik SS, Devi S, Mohapatra A. Neighborhood search-driven accelerated biogeography-based optimization for optimal load dispatch. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*. 2012; 42(5): 641-652.

[26] Meng A, Li J, Yin H. An efficient crisscross optimization solution to large-scale non-convex economic load dispatch with multiple fuel types and valve-point effects. *Energy*. 2016; 113: 1147-1161.

[27] Zhan J, Wu QH, Guo C, Zhou X. Economic dispatch with non-smooth objectives—Part II: dimensional steepest decline method. *IEEE Transactions on Power Systems*. 2015; 30(2): 722-733.

[28] Qin Q, Cheng S, Chu X, Lei X, Shi Y. Solving non-convex/non-smooth economic load dispatch problems via an enhanced particle swarm optimization. *Applied Soft Computing*. 2017; 59: 229-242.

[29] Khesti M, Liu H, Zhao B. Double weighted particle swarm optimization to non-convex wind penetrated emission/economic dispatch and multiple fuel option systems. *Renewable Energy*. 2018; 125: 1021-1037.

[30] Mohammadi F, Abdi H. A modified crow search algorithm (MCSA) for solving economic load dispatch problem. *Applied Soft Computing*. 2018; 71: 51-65.

[31] Salimi H. Stochastic fractal search: a powerful metaheuristic algorithm. *Knowledge-Based Systems*. 2015; 75: 1-18.