RESEARCH ARTICLE

WILEY

# Efficient method for estimation of smooth and nonsmooth fuel cost curves for thermal power plants

Samir Sayah[1] | Abdellatif Hamouda[2]

[1] QUERE Laboratory, Department of Electrical Engineering, Ferhat Abbas University, Setif 1, Algeria

[2] QUERE Laboratory, Department of Optics and Fine Mechanics, Ferhat Abbas University, Setif 1, Algeria

**Correspondence**
Samir Sayah, QUERE Laboratory, Department of Electrical Engineering, Ferhat Abbas University, Setif 1, Algeria.
Email: samir_pg04@yahoo.fr; samir.sayah@univ-setif.dz

**Summary**

For accurate economic dispatch solution, it is necessary to periodically estimate the parameters of fuel cost function. This paper proposes an improved differential evolution algorithm for computing the optimal parameters of fuel cost functions for thermal power plants. A new mutation strategy is suggested to enhance convergence rate and improve solution quality of original differential evolution. The proposed approach is examined on different test systems with several generator cost curve models involving smooth and nonsmooth/nonconvex functions. The results using the proposed approach are compared to those available in recent literature. The results show the efficiency of the proposed estimation approach for obtaining accurate fuel cost parameters without any restriction on the mathematical model of the generator cost curve.

KEYWORDS

generation output of thermal units and thus ensure optimum economic operation of electric power generation systems. To this end, it is important to use accurate and reliable estimation method. Various mathematical models are used for representing thermal generating unit input-output curve. They are of 2 types, smooth and nonsmooth models. These 2 models are discussed in Section 2.

So far, only few works devoted to the cost function parameters estimation have been reported in literature, because research on the subject has only recently begun to gain interest. In previous studies, several traditional techniques were proposed for solving such estimation problems using smooth polynomial functions. These methods are based either on static estimation techniques like least error square and least absolute value[2-4] or dynamic estimation techniques that use Kalman filter.[5] Unlike static approaches, where the optimal solution is obtained using the whole set of data, dynamic filters are recursive algorithms that update the estimates at each new measurement. Moreover, the recursive propriety of dynamic filters requires a large amount of data to reach steady-state solution.

In practice, actual input-output characteristics of large steam turbine generators are inherently nonsmooth and nonconvex functions with higher order nonlinearities and discontinuities because of the valve-point effect.[6] This phenomenon is due to throttling losses that are caused by partial valve opening that produces ripples in the cost function.[1] Thus, estimating this kind of cost curves is very difficult, if not impossible, using traditional methods because of nondifferentiable and nonconvex regions.

As an alternative to conventional methods, the recently developed evolutionary techniques, like genetic algorithms,[7] particle swarm optimization (PSO),[8,9] artificial bee colony,[10] and cuckoo search (CS) algorithm,[11] have been used to estimate the fuel cost parameters. Through the previous researches, modern approaches have shown promising success in this topic. Although the above-mentioned methods were devoted to the parameter estimation problem, the great economic impact of the task requires efficient algorithms that accurately estimate the actual input-output curve. In view of the above, the main focus of this paper is to propose an efficient modified differential evolution (DE) algorithm for

## 2.2 | Nonsmooth fuel cost function

If cost curves of small steam power plants can be well represented by smooth polynomial functions, for large thermal power plants, the smooth model is no longer valid. In practice, for large steam turbine generators with multiple admission valves, actual input-output characteristics are nonsmooth, nonconvex, and present higher order nonlinearities and discontinuities because of valve-point effect. This effect introduces ripples in the input-output curve that are the result of throttling losses that occur as each steam admission valve is first opened.

For more accurate modeling of valve-point effect, an additional sine term is added to the smooth polynomial function 1 in this manner[11]:

$$F_i(P_i) = \left(a_{0\,i} + \sum_{j=1}^{M} a_{j\,i}\, P_i^j\right) + \left|e_i \sin\left(f_i\left(P_i^{\min} - P_i\right)\right)\right| + \varepsilon_i, \qquad i = 1, 2, \ldots, N \tag{2}$$

where $e_i$ and $f_i$ are the valve-point effects factors of generator $i$ and $P_i^{\min}$ is the minimum power output limit of generator $i$.

For the purpose of this study, smooth and nonsmooth models are used to test the effectiveness of the proposed technique. The 3 models under consideration are as follows:

- Smooth second-order model

$$F_i(P_i) = a_{0\,i} + a_{1\,i}\, P_i + a_{2\,i}\, P_i^2 + \varepsilon_i \tag{3}$$

- Smooth third-order model

$$F_i(P_i) = a_{0\,i} + a_{1\,i}\, P_i + a_{2\,i}\, P_i^2 + a_{3\,i}\, P_i^3 + \varepsilon_i \tag{4}$$

- Nonsmooth model with valve-point loading effects[6]

$$F_i(P_i) = a_{0\,i} + a_{1\,i}\, P_i + a_{2\,i}\, P_i^2 + \left|e_i \sin\left(f_i\left(P_i^{\min} - P_i\right)\right)\right| + \varepsilon_i \tag{5}$$

# 3 | PROBLEM FORMULATION

Considering measurement data relating fuel cost input and active power output for a generator $i$ and using 1 of the 3 models (3-5), a set of nonlinear equations can be formulated as follows[8]:

$$Z_i = F_i(P_i, X_i) + E_i \tag{6}$$

where $Z_i$ is the vector of measured or actual values of generation costs for $i$th generator, $P_i$ is the vector including real power output associated with $Z_i$, $X_i$ are the model cost parameters to estimate for $i$th thermal unit, and $E_i$ is the vector of errors associated with $Z_i$. The error vector is obtained by subtracting actual and estimated values of fuel costs:

$$E_i = Z_i - F_i(P_i, X_i) \tag{7}$$

The problem can be formulated as find an estimate of the cost parameter vector $X_i$ that minimizes the error vector $E_i$. Each error vector element $\varepsilon_i$ is computed by the difference between actual value $F_{i(\mathrm{actual})}$ and estimated value $F_{i(\mathrm{estimated})}$[8]:

$$\varepsilon_i = F_{i(\mathrm{actual})} - F_{i(\mathrm{estimated})} \tag{8}$$

At each step, a cost function estimate $F_{i(\mathrm{estimated})}$ can be calculated using the estimated parameters $X_i$.

The objective of this study is then to minimize the total estimation error, which can be computed by the sum of the absolute values of the deviations in the error vector. The total estimation error is then formulated by

$$\text{Total error} = \sum |\varepsilon_i| \tag{9}$$

In the stated optimization problem, each parameter is constrained between minimum and maximum bounds, as follows:

$$C_i^{\min} \leq X_i \leq C_i^{\max} \tag{10}$$

where $C_i^{\min}$ and $C_i^{\max}$ are the lower and upper bounds of cost parameters for unit $i$.

## 4 | DIFFERENTIAL EVOLUTION ALGORITHM

Differential evolution (DE) algorithm is a type of evolutionary search techniques that evolves a set of candidate solutions by performing 3 main operators: mutation, crossover, and selection, which are presented in this section. Differential evolution requires a number $N_P$ of candidate solutions or individuals that may be represented as $X_{i^{(G)}}$, $i = 1, \ldots, N_P$, where $i$ index denotes the population and $G$ represents the current generation. Each solution is encoded using a real valued vector of $D$ variables, depending on the problem dimension:

$$X_i^{(G)} = \left[x_{1,i}^{(G)}, \ldots, x_{D,i}^{(G)}\right]^{\mathrm{T}} \tag{11}$$

The set of these individuals form a population $P^{(G)}$:

$$P^{(G)} = \left[X_1^{(G)}, \ldots, X_{N_P}^{(G)}\right] \tag{12}$$

The optimization procedure of the DE technique is given below:

a) Initialization. The first phase of DE algorithm is to generate randomly an initial population containing $\mathrm{N_P}$ D-dimensional vectors, in this manner:

$$x_{j,i}^{(0)} = x_j^{\min} + \mu_j\left(x_j^{\max} - x_j^{\min}\right) \tag{13}$$

where $i = 1, 2, \ldots, N_P$ and $j = 1, 2, \ldots, D$; $\mu_j$ denotes a random variable within the range [0, 1]; and $x_j^{\max}$ and $x_j^{\min}$ are the maximum and minimum value for parameter $j$.

b) Mutation. A mutated solution $X_i^{'}$ is obtained by a linear combination of 3 randomly selected individuals ($X_a$, $X_b$, and $X_c$) in the current population using the following formula[14]:

$$X_i^{'(G)} = X_a^{(G)} + \alpha\left(X_b^{(G)} - X_c^{(G)}\right), \quad i = 1, 2, \quad \ldots, N_P \tag{14}$$

where $\alpha$ is the mutation constant within the range [0, 2] used to control the algorithm convergence speed.

c) Crossover. To enhance the population diversity, the crossover operator is applied. In this stage, the parameters of the produced mutated vector $X_i^{'}$ are recombined with its parent vector $X_i$ to form the final trial vector $X_i^{''}$ according to a predefined crossover factor $C_R \in [0, 1]$. Crossover operation is implemented as follows:

$$x_{j,i}^{''(G)} = \begin{cases} x_{j,i}^{'(G)} & \text{if } \rho_j \le C_R \text{ or } j = s \\ x_{j,i}^{(G)} & \text{otherwise} \end{cases}, \quad i = 1, \; 2, \; \ldots, N_P, \; j = 1, 2, \; \ldots, \; D \tag{15}$$

where $\rho_j$ denotes a randomly chosen variable between 0 and 1 and $s$ is a randomly selected integer within the set $\{1, \ldots, N_P\}$ that ensures that the final trial vector gets at least 1 parameter from the mutated vector.

d) Selection. In this last stage, the fitness value of the final trial vector $X_i^{''}$ is compared with that of the parent vector $X_i$, and the vector with better fitness value of the two is kept for the new population. The selection operator is implemented in the following way[14]:

$$X_i^{(G+1)} = \begin{cases} X_i^{''(G)}, & \text{if } fit\left(X_i^{''(G)}\right) \le fit\left(X_i^{(G)}\right) \\ X_i^{(G)}, & \text{otherwise} \end{cases}, \quad i = 1, \; 2, \; \ldots, N_P \tag{16}$$

where $fit\left(X^{''}_{i^{(G)}}\right)$ and $fit\left(X_{i^{(G)}}\right)$ are the fitness values of the final trial vector and parent vector, respectively.

In short, excluding the initialization stage, the above-mentioned steps are continued until a stopping criterion is attained.

## 5 | IMPROVED DIFFERENTIAL EVOLUTION ALGORITHM

In the mutation step of standard DE, the base vector is selected from 3 random vectors. On the one hand this strategy explores the solution space, but on the other hand the DE convergence is slowed down. Furthermore, the mutation factor $\alpha$ of the standard DE is a positive constant, which restrains its exploration ability.

The first proposed modification to DE is applied to the mutation rule 14. The best vector, named $X_{tb}^{(G)}$, among the 3 randomly vectors ($X_a$, $X_b$, and $X_c$) is selected as the base vector. The 2 remaining individuals are used to compute the difference vector. In this way, exploration is performed in regions around the best individuals $X_{tb}^{(G)}$. This approach preserves the exploratory feature and speed ups the convergence.[15]

The second proposed modification in DE is to adopt a random mutation factor $\alpha$ within the range $[-1, -0.4] \cup [0.4, 1]$ for each mutated vector, instead of using a constant factor through the DE procedure.[15] This random search strategy

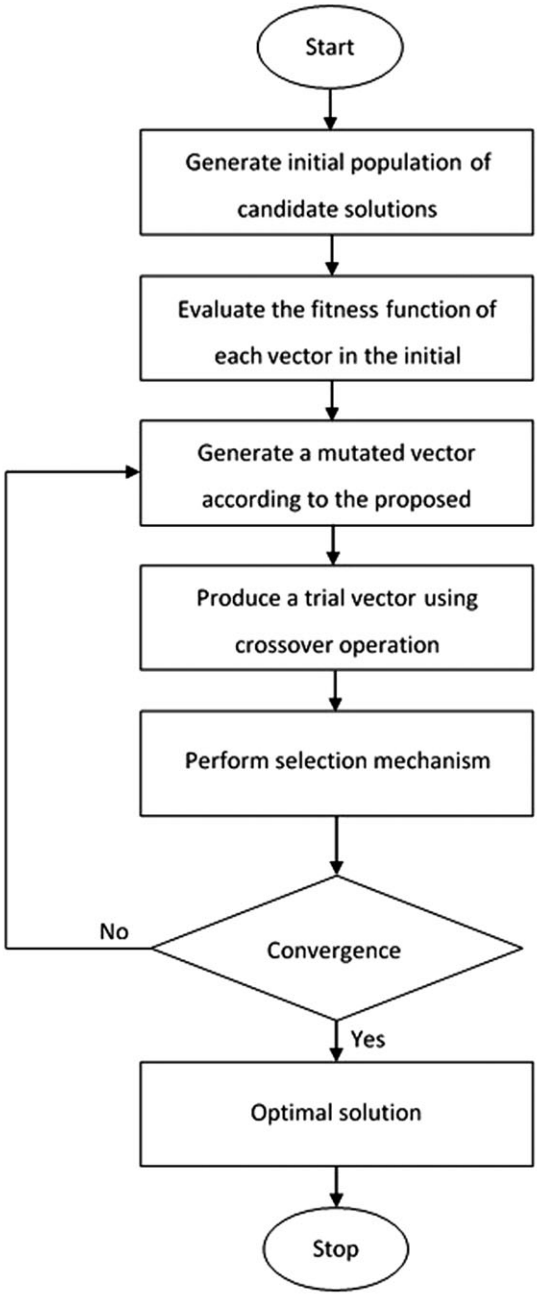


**FIGURE 1** General flowchart of the proposed IDE estimation algorithm

progressively transforms itself into the search intensification feature for rapid convergence when the points in the solution space form a cluster around the global minimizer.

The general flowchart the proposed IDE estimation algorithm for fuel cost parameters is depicted in Figure 1.

**TABLE 1** Tuned control parameters of improved differential evolution algorithm

| Parameter | Value |
|---|---|
| Population size ($N_p$) | 50 |
| Crossover factor ($C_R$) | 0.99 |
| Maximum number of generations ($G^{\max}$) | 1000 |

**TABLE 2** Estimated fuel cost coefficients and estimation errors for case study 1 (quadratic model)

| | | Coefficients | | | | | |
|---|---|---|---|---|---|---|---|
| **Unit** | **$P$ (MW)** | | **Actual** | **IDE** | **$F_{actual}$ (\$/h)** | **Error** | **Total Error** |
| 1 | 100 | $a_0$ | 150 | 149.9966 | 389.00 | 0.000003 | **0.005167** |
| | 400 | $a_1$ | 1.89 | 1.8900 | 1706.00 | 0.004518 | |
| | 700 | $a_2$ | 0.005 | 0.0050 | 3923.00 | 0.000646 | |
| 2 | 100 | $a_0$ | 115 | 115.0005 | 370.00 | 0.000593 | **0.005139** |
| | 325 | $a_1$ | 2.00 | 2.0000 | 1345.94 | 0.002612 | |
| | 550 | $a_2$ | 0.0055 | 0.0055 | 2878.75 | 0.001933 | |
| 3 | 10 | $a_0$ | 40 | 40.0009 | 75.60 | 0.000530 | **0.003654** |
| | 180 | $a_1$ | 3.50 | 3.5000 | 864.40 | 0.002599 | |
| | 350 | $a_2$ | 0.0060 | 0.0060 | 2000.00 | 0.000525 | |
| 4 | 50 | $a_0$ | 122 | 121.9993 | 293.25 | 0.000593 | **0.006238** |
| | 200 | $a_1$ | 3.15 | 3.1500 | 972.00 | 0.000577 | |
| | 350 | $a_2$ | 0.0055 | 0.0055 | 1898.25 | 0.005068 | |
| 5 | 100 | $a_0$ | 125 | 124.9925 | 480.00 | 0.0000099 | **0.000102** |
| | 275 | $a_1$ | 3.05 | 3.0501 | 1341.88 | 0.0000598 | |
| | 450 | $a_2$ | 0.0050 | 0.0050 | 2510.00 | 0.0000327 | |
| 6 | 50 | $a_0$ | 120 | 120.0013 | 275.00 | 0.00061 | **0.001149** |
| | 200 | $a_1$ | 2.75 | 2.7500 | 950.00 | 0.00036 | |
| | 350 | $a_2$ | 0.0070 | 0.0070 | 1940.00 | 0.00018 | |
| 7, 8 | 50 | $a_0$ | 70 | 69.9902 | 260.00 | 0.00452 | **0.012486** |
| | 175 | $a_1$ | 3.45 | 3.4501 | 888.13 | 0.00391 | |
| | 300 | $a_2$ | 0.0070 | 0.0070 | 1735.00 | 0.00405 | |
| 9 | 50 | $a_0$ | 70 | 70.0000 | 260.00 | 0.000023 | **0.000313** |
| | 200 | $a_1$ | 3.45 | 3.4500 | 1040.00 | 0.000035 | |
| | 350 | $a_2$ | 0.0070 | 0.0070 | 2135.00 | 0.000256 | |
| 10, 11 | 50 | $a_0$ | 130 | 129.9975 | 265.00 | 0.000019 | **0.000101** |
| | 275 | $a_1$ | 2.45 | 2.4501 | 1181.88 | 0.000007 | |
| | 500 | $a_2$ | 0.0050 | 0.0050 | 2605.00 | 0.000074 | |
| 12 | 50 | $a_0$ | 135 | 135.0000 | 266.25 | 0.000023 | **0.000294** |
| | 300 | $a_1$ | 2.35 | 2.3500 | 1335.00 | 0.000169 | |
| | 550 | $a_2$ | 0.0055 | 0.0055 | 3091.25 | 0.000102 | |
| 13 | 100 | $a_0$ | 200 | 200.0036 | 405.00 | 0.001452 | **0.004859** |
| | 450 | $a_1$ | 1.60 | 1.6000 | 1831.25 | 0.001840 | |
| | 800 | $a_2$ | 0.0045 | 0.0045 | 4360.00 | 0.001567 | |
| 14 | 0 | $a_0$ | 45 | 45.0006 | 45.00 | 0.000617 | **0.003817** |
| | 100 | $a_1$ | 3.89 | 3.8900 | 494.00 | 0.000222 | |
| | 200 | $a_2$ | 0.0060 | 0.0060 | 1063.00 | 0.002978 | |

The bold values show the total estimation errors achieved with IDE.

## 6 | SOLUTION METHODOLOGY

The implementation of the IDE technique for estimation of the fuel cost function parameters is summarized in the following steps.

Step 1. Generate initial population of candidate solutions between lower and upper bounds of fuel cost parameters, using Equation 13
Step 2. Calculate the objective function value (ie, total estimation error) for each vector in the initial population using Equation 9.
Step 3. Generate a mutated vector according to the proposed strategy described in Section 5.
Step 4. Produce a trial vector according to Equation 15 of crossover operation.
Step 5. Calculate the objective function value of the trial vector.
Step 6. Perform selection mechanism using Equation 16.
Step 7. Check the convergence criteria. If one of them is satisfied, then stop, else go back to step 3.

In this work, the search procedure will terminate whenever the predetermined maximum number of iterations $G^{max}$ is reached, or whenever the global best solution is not improved over a predefined number of generations.

## 7 | SIMULATION

The performance of the proposed method has been evaluated by estimating 3 types of cost functions. The first test case considered is the smooth second-order model of generators defined with Equation 3, using the 14-unit system taken from Venkatesh et al.[16] In the second test example, the smooth third-order model given by Equation 4 is studied using the 3-unit system obtained from the previous studies.[7,8] In the third test case, the nonsmooth/nonconvex model defined with Equation 5 is examined using the 2-unit system available in Alrashidi et al.[11] For each case, IDE was executed 50 times with different random initial solutions to assess its robustness. The suitable tuned control parameters of IDE used for all test cases are listed in Table 1. The IDE algorithm has been implemented in MATLAB package, and simulations were performed on a 2.10-GHz Intel Core 2 Duo laptop computer having 3 GB memory.

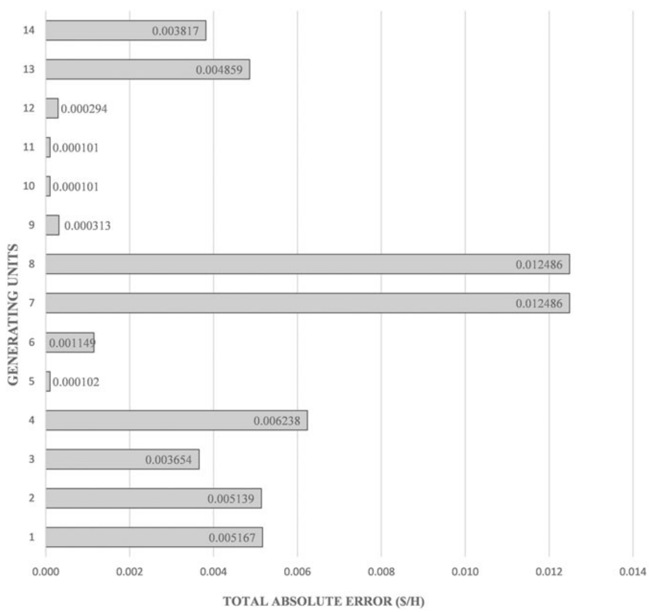


**FIGURE 2** Total estimation errors by IDE for case study 1 (quadratic model)

## 7.1 | Case study 1: second-order model

In this test case, the proposed approach is implemented for estimating the cost curve parameters of the IEEE 118 bus test system consisting of 14 thermal units using quadratic model. The system data comprising the actual fuel cost parameters and active power outputs are given in Venkatesh et al[16] and are reported in Table 2. It should be noted that actual fuel costs displayed in Table 2 were determined on the basis of actual cost parameters. The IDE algorithm has been applied for each generating unit to estimate again its fuel cost coefficients and compare them to the actual values. For 50 trial runs, and for each unit, IDE has given the same total absolute estimation errors shown in Table 2. For this case study, the average execution time was 0.10 seconds. Table 2 shows also the estimated coefficients of the cost function achieved with IDE, which are almost identical to the actual coefficients.

The total estimation error achieved with IDE for each generator is illustrated in Figure 2. From this figure, it is observed that estimation errors are close to zero, which confirms the accuracy of the estimates.

**TABLE 3** Estimated fuel cost parameters for case study 2 (cubic model)

| | | IDE | ABC[10] | PSO[8] | LES[8] |
|---|---|---|---|---|---|
| Unit 1 (coal) | $a_0$ | 127.0666667 | 124.5362 | 120.241 | 123.180 |
| | $a_1$ | 3.11866666 | 3.4859 | 3.979 | 3.535 |
| | $a_2$ | 0.19993337 | 0.1872 | 0.184 | 0.193 |
| | $a_3$ | −0.00162667 | −0.0015 | −0.002 | −0.002 |
| Unit 2 (oil) | $a_0$ | 132.5000000 | 129.2351 | 130.278 | 128.640 |
| | $a_1$ | 3.3325000 | 3.4859 | 3.542 | 3.746 |
| | $a_2$ | 0.2058750 | 0.1872 | 0.200 | 0.199 |
| | $a_3$ | −0.0016625 | −0.0015 | −0.002 | −0.002 |
| Unit 3 (gas) | $a_0$ | 132.3333295 | 126.0143 | 128.376 | 128.400 |
| | $a_1$ | 3.6250006 | 3.8044 | 4.146 | 4.046 |
| | $a_2$ | 0.2024166 | 0.1896 | 0.188 | 0.195 |
| | $a_3$ | −0.0016250 | −0.0015 | −0.002 | −0.002 |

Abbreviations: ABC, artificial bee colony; IDE, improved differential evolution; LES, least error square; PSO, particle swarm optimization.

**TABLE 4** Estimation errors for case study 2 (cubic model)

| | | | Error | | | |
|---|---|---|---|---|---|---|
| | **$P$ (MW)** | **$F_{actual}$ (GJ/h)** | **IDE** | **ABC[10]** | **PSO[8]** | **LES[8]** |
| Unit 1 (coal) | 10 | 176.62 | 0.0000 | 0.0048 | 0.186 | 0.393 |
| | 20 | 256.40 | 0.0000 | 0.7342 | 4.157 | 1.874 |
| | 30 | 361.50 | 4.8533 | 4.4068 | 0.451 | 1.779 |
| | 40 | 467.60 | 0.0000 | 0.1078 | 3.846 | 3.368 |
| | 50 | 579.50 | 0.0000 | 0.1688 | 0.000 | 2.915 |
| Total error | | | **4.8533** | 5.4224 | 8.641 | 10.329 |
| Unit 2 (oil) | 10 | 184.75 | 0.0000 | 0.0109 | 0.674 | 0.449 |
| | 20 | 268.20 | 0.0000 | 0.9631 | 0.000 | 1.362 |
| | 30 | 377.70 | 4.8249 | 4.1929 | 4.690 | 3.477 |
| | 40 | 488.80 | 0.0000 | 0.0289 | 0.063 | 0.716 |
| | 50 | 606.00 | 0.0000 | 0.0449 | 0.119 | 5.055 |
| Total error | | | **4.8249** | 5.2407 | 5.547 | 11.059 |
| Unit 3 (gas) | 10 | 187.20 | 0.0000 | 0.0167 | 0.099 | 0.396 |
| | 20 | 272.80 | 0.0000 | 1.8323 | 1.526 | 1.888 |
| | 30 | 384.30 | 4.9167 | 3.7387 | 3.300 | 1.848 |
| | 40 | 497.20 | 0.0000 | 0.0297 | 0.874 | 3.296 |
| | 50 | 616.50 | 0.0000 | 0.1593 | 0.000 | 2.720 |
| Total error | | | **4.9167** | 5.7767 | 5.799 | 10.148 |

The bold values show the total estimation errors achieved with IDE.

Abbreviations: ABC, artificial bee colony; IDE, improved differential evolution; LES, least error square; PSO, particle swarm optimization.

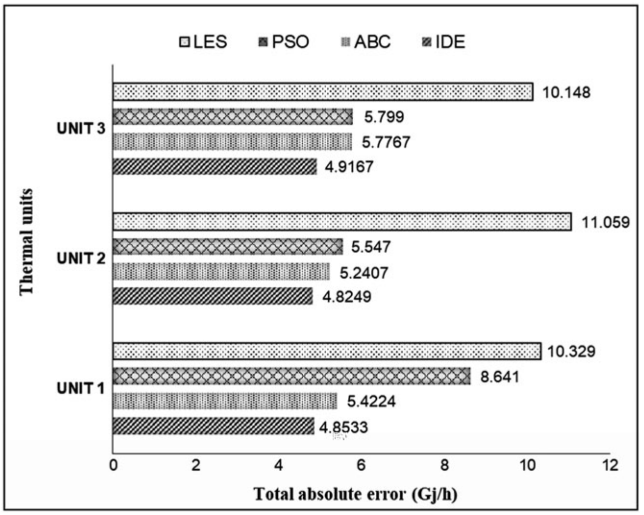


**FIGURE 3** Estimation errors for case study 2 (cubic model)

**TABLE 5** Estimation errors for unit 1 using nonsmooth model (case study 3)

| $P$ (MW) | $F_{actual}$ (GJ/h) | Error | | |
|---|---|---|---|---|
| | | IDE | CS[11] | PSO[9] |
| 0 | 550.000 | 0.0000 | 1.129 | 1.079 |
| 25 | 982.938 | 0.0002 | 0.357 | 0.068 |
| 50 | 1250.896 | 0.0001 | 0.113 | 0.123 |
| 75 | 1307.251 | 0.0001 | 0.558 | 0.715 |
| 100 | 1468.035 | 0.0000 | 0.752 | 0.811 |
| 125 | 1849.962 | 0.0002 | 0.087 | 0.080 |
| 150 | 2028.980 | 0.0004 | 0.065 | 0.376 |
| 175 | 2023.333 | 0.0002 | 0.667 | 1.514 |
| 200 | 2378.296 | 0.0000 | 0.262 | 0.468 |
| 225 | 2686.609 | 0.0003 | 0.238 | 0.162 |
| 250 | 2779.917 | 0.0000 | 0.039 | 0.907 |
| 275 | 2858.341 | 0.0000 | 0.536 | 0.989 |
| 300 | 3269.109 | 0.0003 | 0.317 | 0.137 |
| 325 | 3490.738 | 0.0001 | 0.610 | 0.324 |
| 350 | 3512.636 | 0.0006 | 0.212 | 1.393 |
| 375 | 3785.882 | 0.0002 | 0.228 | 0.307 |
| 400 | 4131.982 | 0.0000 | 0.964 | 0.126 |
| 425 | 4265.053 | 0.0000 | 1.028 | 0.529 |
| 450 | 4264.307 | 0.0000 | 0.102 | 0.621 |
| 475 | 4698.816 | 0.0004 | 1.105 | 0.373 |
| 500 | 4962.688 | 0.0006 | 1.663 | 0.303 |
| Total error | | **0.0037** | 11.032 | 11.405 |
| Total %error | | **$1.41 \times 10^{-6}$** | 0.572 | 0.619 |

The bold values show the total estimation errors achieved with IDE.

Abbreviations: CS, cuckoo search; IDE, improved differential evolution; PSO, particle swarm optimization.

## 7.2 | Case study 2: third-order model

The IDE algorithm is used to estimate the cost function parameters of a test system including 3 identical thermal units but functioning with different fuels (coal, oil, and gas).[8] The fuel cost characteristics are expressed as cubic polynomials. After 50 repeated trials and for each unit the developed algorithm obtained almost similar solution with a mean computation time of 0.80 seconds. The total absolute estimation error was 4.8533 GJ/h for unit 1, 4.8249 GJ/h for unit 2, and 4.9167 GJ/h for unit 3. The cost coefficients given by IDE method are presented in Table 3. Moreover, the absolute errors achieved by the proposed estimation method for each unit are shown in Table 4.

As displayed in Tables 3 and 4, the obtained results are compared to those reported using artificial bee colony,[10] PSO,[8] and least error square.[8] Table 4 shows that the total error is significantly reduced using IDE compared to the previously mentioned approaches. Compared with the above-cited methods, the total error is reduced from 10.50% to 53.02% for the first unit, from 7.93% to 56.37% for the second unit, and from 14.89% to 51.55% for the third unit. These reduction rates show that the IDE algorithm is promising and gives a better fuel cost curve approximation for the nonconvex third-order model.

The different absolute errors are summarized in Figure 3. This figure clearly proves the suitability of the proposed estimation technique.

**TABLE 6** Estimation errors for unit 2 using nonsmooth model (case study 3)

| | | Error | | |
|---|---|---|---|---|
| ***P*** **(MW)** | $F_{actual}$ **(GJ/h)** | **IDE** | **CS[11]** | **PSO[9]** |
| 0 | 309.000 | 0.0004 | 0.002 | 0.590 |
| 18 | 592.185 | 0.0002 | 0.332 | 0.039 |
| 36 | 800.980 | 0.0004 | 0.386 | 0.239 |
| 54 | 901.361 | 0.0002 | 0.040 | 0.138 |
| 72 | 918.567 | 0.0003 | 0.584 | 0.237 |
| 90 | 1161.719 | 0.0005 | 0.176 | 0.260 |
| 108 | 1387.229 | 0.0001 | 0.051 | 0.465 |
| 126 | 1505.826 | 0.0000 | 0.330 | 0.285 |
| 144 | 1533.617 | 0.0002 | 0.895 | 0.172 |
| 162 | 1735.414 | 0.0000 | 0.407 | 0.273 |
| 180 | 1976.564 | 0.0002 | 0.205 | 0.410 |
| 198 | 2113.785 | 0.0005 | 0.417 | 0.152 |
| 216 | 2153.829 | 0.0003 | 0.928 | 0.393 |
| 234 | 2313.540 | 0.0000 | 0.361 | 0.000 |
| 252 | 2569.061 | 0.0002 | 0.077 | 0.075 |
| 270 | 2725.078 | 0.0001 | 0.221 | 0.259 |
| 288 | 2778.892 | 0.0000 | 0.681 | 0.896 |
| 306 | 2896.388 | 0.0000 | 0.041 | 0.562 |
| 324 | 3164.832 | 0.0000 | 0.333 | 0.542 |
| 342 | 3339.577 | 0.0005 | 0.259 | 0.950 |
| 360 | 3408.509 | 0.0003 | 0.151 | 1.681 |
| Total error | | **0.0044** | 6.877 | 8.618 |
| Total %error | | $\mathbf{4.13 \times 10^{-6}}$ | 0.444 | 0.575 |

The bold values show the total estimation errors achieved with IDE.

Abbreviations: CS, cuckoo search; IDE, improved differential evolution; PSO, particle swarm optimization.

### 7.3 | Case study 3: nonsmooth model with valve-point effects

The IDE algorithm is used to estimate the parameters of a more realistic fuel cost function featuring valve-point effects. It is important to point out that estimating this kind of curves is hard, if not impossible, using traditional methods because of nondifferentiable and nonconvex regions. The application system of this case study contained 2 large thermal power plants with a rated power of 500 MW for the first unit and 360 MW for the second unit. The input-output data relating the fuel cost to active power output are obtained from Alrashidi et al[11] and are reported in Tables 5 and 6. A wide range of possible loading conditions, including all rippling effects, is covered by this set of data for both units. These data are exploited for estimating the nonsmooth model parameters.

For 50 trial runs, the total absolute error given by the proposed IDE algorithm is almost the same. It is equal to 0.0037 GJ/h for unit 1 and 0.0044 GJ/h for unit 2. The total estimation error is close to zero for each unit, which demonstrates the accuracy of the developed estimation approach. The average simulation time for this example was 0.91 seconds. The fuel cost parameters estimated by IDE are given in Table 7. Moreover, we have reported in Tables 5 and 6 the absolute errors provided by the proposed estimation method for both units.

For comparison purposes, the results obtained from CS[11] and particle swarm optimization (PSO)[9] are included in Tables 5, 6, and 7. Obviously, from Tables 5 and 6 the total estimation error given by our approach is negligible compared to CS[11] and PSO[9] for both units. These results validate the reliability and accuracy of the proposed IDE estimation technique even in the presence of generating units involving the effects of multiple valve-points.

### 7.4 | Convergence characteristic

Recently, Saber et al[17] have recommended the third-order polynomial for modeling fuel cost function because it provides a better representation of actual generator operation. Moreover, Theerthamalai and Maheswarapu[18] have shown that the economic dispatch solution can be greatly improved by using the cubic function model. On the other hand, when practical operating conditions such as valve-point loading effects are considered, real fuel cost characteristic should be

**TABLE 7** Fuel cost parameters for case study 3 (nonsmooth model)

| | | IDE | CS[11] | PSO[9] |
|---|---|---|---|---|
| Unit 1 (500 MW) | $a_0$ | 550.000000004 | 551.129 | 548.92129 |
| | $a_1$ | 8.099999462 | 8.100 | 8.09575 |
| | $a_2$ | 0.000279999 | 0.000 | 0.00029 |
| | $e$ | 300.000184360 | 298.990 | 301.42437 |
| | $f$ | 637.079990147 | 99.561 | 0.03500 |
| Unit 2 (360 MW) | $a_0$ | 308.999625883 | 309.015 | 308.41028 |
| | $a_1$ | 8.100003011 | 8.090 | 8.10701 |
| | $a_2$ | 0.000559990 | 0.00586 | 0.00053 |
| | $e$ | 200.000481965 | 200.7088 | 200.61149 |
| | $f$ | 421.189948503 | 382.6182 | 0.04201 |

Abbreviations: CS, cuckoo search; IDE, improved differential evolution; PSO, particle swarm optimization.

**TABLE 8** Statistical results of IDE and DE using cubic model (case 2)

| Method | | Minimum Total Absolute Error | Maximum Total Absolute Error | Mean Total Absolute Error | Mean Iteration | Standard Deviation | Average CPU Time, s |
|---|---|---|---|---|---|---|---|
| DE | Unit 1 | 4.8535 | 4.8642 | 4.8551 | 722 | 0.01319 | 1.46 |
| **IDE** | | **4.8533** | **4.8533** | **4.8533** | **274** | $\mathbf{2.7018 \times 10^{-7}}$ | **0.81** |
| DE | Unit 2 | 4.8252 | 4.8289 | 4.8263 | 740 | 0.00541 | 1.24 |
| **IDE** | | **4.8249** | **4.8250** | **4.8250** | **268** | $\mathbf{2.0234 \times 10^{-7}}$ | **0.74** |
| DE | Unit 3 | 4.9167 | 4.9233 | 4.9183 | 726 | 0.01003 | 1.28 |
| **IDE** | | **4.9167** | **4.9167** | **4.9167** | **280** | $\mathbf{1.1184 \times 10^{-6}}$ | **0.83** |

The bold values show the statistical results achieved with IDE.

Abbreviations: DE, differential evolution; IDE, improved differential evolution.

expressed by quadratic curve model with sine term.[6,11] Hence, for this research we have examined the smooth cubic model studied in case 2 and the nonsmooth model used in case 3.

To assess the performance of the proposed IDE, we have also solved cases 2 and 3 with standard DE implemented under MATLAB package on the same laptop computer. Both optimization techniques (IDE and DE) have been executed 50 times using the same control parameters of IDE given in Table 1. Note that for the DE algorithm the

**TABLE 9** Statistical results of IDE and DE using nonsmooth model with valve-point effects (case 3)

| Method | | Minimum total absolute error | Maximum total absolute error | Mean total absolute error | Mean iteration | Standard deviation | Average CPU time (s) |
|---|---|---|---|---|---|---|---|
| DE | Unit 1 | 0.0036928 | 0.0426981 | 0.0047328 | 952 | 0.038539 | 1.33 |
| **IDE** | | **0.0036805** | **0.0036807** | **0.0036806** | **302** | **$2.5818 \times 10^{-7}$** | **0.91** |
| DE | Unit 2 | 0.0044672 | 0.0375239 | 0.0053243 | 922 | 0.032550 | 1.48 |
| **IDE** | | **0.0044474** | **0.0044476** | **0.0044474** | **300** | **$1.5035 \times 10^{-7}$** | **0.92** |

The bold values show the statistical results achieved with IDE.

Abbreviations: DE, differential evolution; IDE, improved differential evolution.

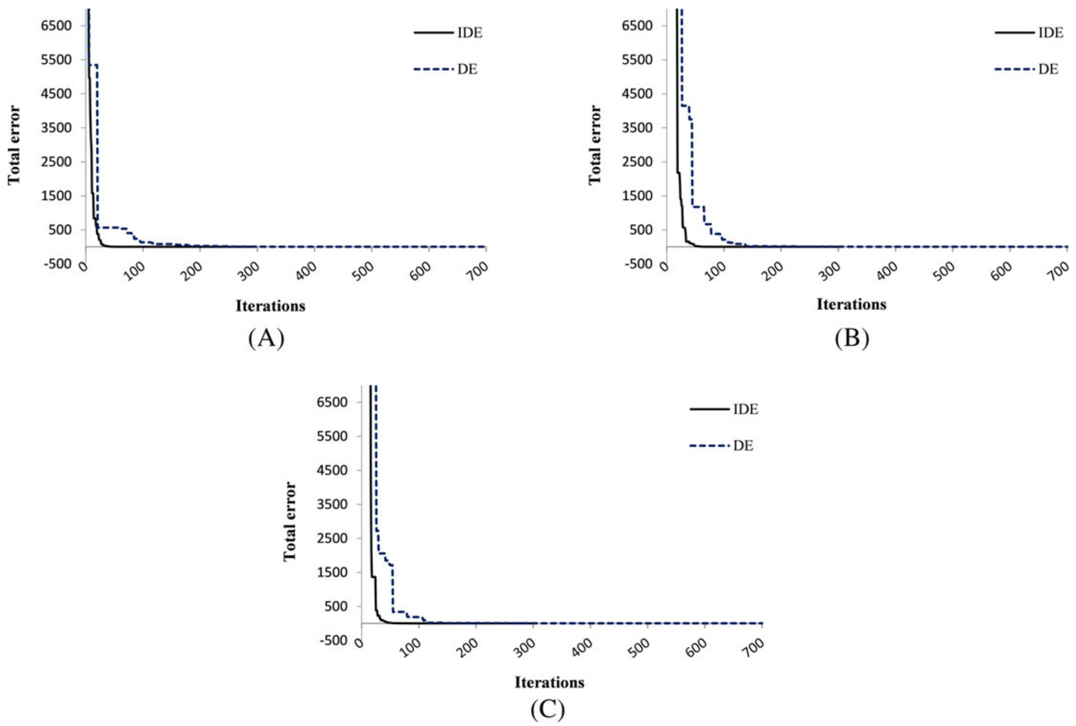


**FIGURE 4** Convergence characteristics of IDE and DE for cubic model (case 2): A, unit 1, B, unit 2, and C, unit 3

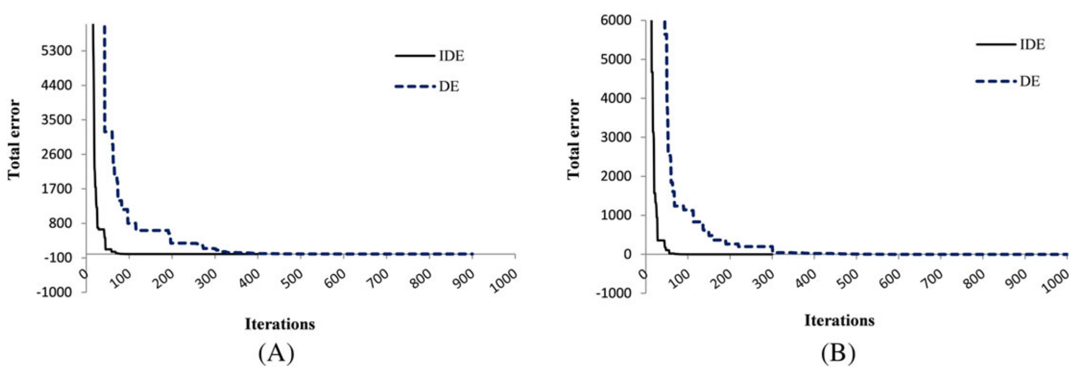


**FIGURE 5** Convergence characteristics of IDE and DE for nonsmooth model (case 3): A, unit 1 and B, unit 2

mutation factor $\alpha$ was set to 0.5. Tables 8 and 9 compare the minimum error, maximum error, mean error, mean iteration, standard deviation, and average CPU time for 50 trials. As seen from these tables, the best, worst, and average solutions achieved with the IDE technique are better than those of standard DE. Furthermore, we note that the worst solution of the IDE strategy outperforms the best solution achieved with the conventional DE approach. Moreover, we observe that the IDE is computationally more efficient than classical DE, which confirms the reliability of the proposed estimation algorithm. The small standard deviation of IDE indicates the stability of the method and accuracy of the estimates.

In Figures 4 and 5, we have compared the convergence characteristics of the best solution of IDE and DE. This figure shows that IDE has a better convergence behavior than standard DE. This can be explained by the inclusion of the new mutation strategy that enhances the convergence rate and improve solution quality.

## 8 | CONCLUSION

An IDE approach is introduced and applied to estimate the optimal parameters of the fuel cost functions. These parameters play essential role in accuracy of economic dispatch calculations. The parameter estimation problem has been expressed as an optimization problem where the goal is the minimization of total estimation error. The effectiveness of the developed IDE algorithm has been illustrated by its application through 3 case studies involving convex models and nonsmooth/nonconvex model with valve-point loading effects. Numerical results show that the proposed technique is computationally efficient and able to converge to accurate solutions with stable convergence characteristics. Compared to some recent methods, the proposed approach achieves better estimates. Overall, the IDE approach has been found to be an effective and viable technique for power utilities to solve such estimation problem without restrictions on the mathematical model of fuel cost function.

### ORCID

*Samir Sayah* http://orcid.org/0000-0002-3407-189X

### REFERENCES

1. Wood AJ, Wollenberg BF. *Power Generation, Operation and Control*. 2nd ed. New York: John Wiley and Sons; 1996.
2. Chen HYK, Postel CE. On-line parameter identification of input–output curves for thermal units. *IEEE Trans Power Syst*. 1986;1(2):221-224. https://doi.org/10.1109/TPWRS.1986.4334933
3. El-Shibini M, Osman ZH. A novel technique to estimate the fuel cost functions for economic operation of power systems. *Electr Power Energy Syst*. 1989;11(2):109-114. https://doi.org/10.1016/0142-0615(89)90018-5
4. Soliman SA, Emam SA, Christensen GS. Optimization of the optimal coefficients of non-monotonically increasing incremental cost curves. *Electr Pow Syst Res*. 1991;21(2):99-106. https://doi.org/10.1016/0378-7796(91)90023-G
5. Soliman SA, Al-Kandari AM. Kalman filtering algorithm for on-line identification of input–output curves for thermal power plant. *8th IEEE Mediterranean Electrotechnical Conference*. Italy, 1996;1588–1593.
6. Beigvand SD, Abdi H, La Scala M. Combined heat and power economic dispatch problem using gravitational search algorithm. *Electr Pow Syst Res*. 2016;133:160-172. https://doi.org/10.1016/j.epsr.2015.10.007
7. Al-Kandari AM, El-Naggar KM. A genetic-based algorithm for optimal estimation of input–output curve parameters of thermal power plants. *Electr Eng*. 2007;89(8):585-590. https://doi.org/10.1007/s00202-006-0047-x
8. EL-Naggar KM, AlRashidi MR, Al-Othman AK. Estimating the input–output parameters of thermal power plants using PSO. *Energ Conver Manage*. 2009;50(7):1767-1772. https://doi.org/10.1016/j.enconman.2009.03.019
9. Alrashidi MR, EL-Naggar KM, Al-Othman AK. Particle swarm optimization based approach for estimating the fuel-cost function parameters of thermal power plants with valve loading effects. *Electr Power Compon Syst*. 2009;37(11):1219-1230. https://doi.org/10.1080/15325000902993589
10. Sonmez Y. Estimation of fuel cost curve parameters for thermal power plants using the ABC algorithm. *Turk J Electr Eng Comput Sci*. 2013;21:1827-1841. https://doi.org/10.3906/elk-1203-10
11. Alrashidi MR, EL-Naggar KM, Alhajri MF. Convex and non-convex heat curve parameters estimation using cuckoo search. *Arab J Sci Eng*. 2015;40(3):873-882. https://doi.org/10.1007/s13369-014-1547-z

12. Storn R, Price K. Differential evolution—a simple and efficient adaptive scheme for global optimization over continuous spaces. *J Glob Optim*. 1997;11(4):341-359. https://doi.org/10.1023/A:1008202821328
13. Storn R. On the usage of differential evolution for function optimization. *Biennial Conference of the North American Fuzzy Information Processing Society, NAFIPS*. Berkley, 1996;519–523.
14. Sayah S, Hamouda A. A hybrid differential evolution algorithm based on particle swarm optimization for nonconvex economic dispatch problems. *Appl Soft Comput*. 2013;13(4):1608-1619. https://doi.org/10.1016/j.asoc.2012.12.014
15. Sayah S, Zehar K. Modified differential evolution algorithm for optimal power flow with non-smooth cost functions. *Energ Conver Manage*. 2008;49(11):3036-3042. https://doi.org/10.1016/j.enconman.2008.06.014
16. Venkatesh P, Gnanadass R, Padhy NP. Comparison and application of evolutionary techniques to combined economic emission dispatch with line flow constraints. *IEEE Trans Power App Syst*. 2003;18(2):688-697. https://doi.org/10.1109/TPWRS.2003.811008
17. Saber AY, Chakraborty S, Abdur Razzak SM, Senjyu T. Optimization of economic load dispatch of higher order general cost polynomials and its sensitivity using modified particle swarm optimization. *Electr Pow Syst Res*. 2009;79(1):98-106. https://doi.org/10.1016/j.epsr.2008.05.017
18. Theerthamalai A, Maheswarapu S. An effective non-iterative “λ-logic based” algorithm for economic dispatch of generators with cubic fuel cost function. *Electr Power Energy Syst*. 2010;32(5):539-542. https://doi.org/10.1016/j.ijepes.2009.11.002

**How to cite this article:** Sayah S, Hamouda A. Efficient method for estimation of smooth and nonsmooth fuel cost curves for thermal power plants. *Int Trans Electr Energ Syst*. 2018;28:e2498. https://doi.org/10.1002/etep.2498