A comparative study of economic load dispatch using sine cosine algorithm

N. Patel and K. Bhattacharjee

Department of Electrical Engineering, Institute of Technology, Nirma University, SG Highway, Gota Ahmedabad - 382481, India.

Received 15 March 2018; received in revised form 28 July 2018; accepted 15 October 2018

KEYWORDS
Economic Load Dispatch (ELD); Optimization; Prohibited operating zone; Sine Cosine Algorithm (SCA); Valve-point loading.

Abstract. Economic Load Dispatch (ELD) is an important part of cost minimization procedure in power system operation. Different derivative and probabilistic methods are used to solve ELD problems. This paper proposes a powerful Sine Cosine Algorithm (SCA) to explain the ELD issue including equality and inequality restrictions. The main aim of ELD is to satisfy the entire electric load at minimum cost. The SCA is a population-based probabilistic method, which guides its search agents that are randomly placed in the search space towards an optimal point using their fitness functions and keeps a track of the best solution achieved by each search agent. SCA was used to solve the ELD problem due to its favorable exploration and local optima escaping technique. This algorithm confirmed that promising areas of the search space were exploited to have a smooth transition from exploration to exploitation using sine and cosine functions. Simulation results proved that the proposed algorithm surpassed other existing optimization techniques in terms of quality of the solution obtained and computational efficiency. The final results also proved the robustness of the SCA.

© 2018 Sharif University of Technology. All rights reserved.

1. Introduction

Economic Load Dispatch (ELD) is considered as one of the valuable optimization problems in the field of power system operations. The ELD fulfills the total load demand by economically allocating the load demand to each and every generator while satisfying their operation and physical constraints. The main aim of the ELD is to make the entire system reliable and to minimize the total generation cost of the thermal power plant. Also, it satisfies all the constraints on each and every generator that is considered for the ELD problem.

There are many classical optimization methods, e.g., gradient method [1], Quadratic Programming (QP) [2], Lagrangian relaxation [3], Hopfield modeling framework [4], Linear Programming (LP) [5], and Dynamic Programming (DP) [6], which assume a linear increasing cost function. The application of such methods to solving the ELD problem has generally been successful. However, the main problem with the classical approach is that it tends to converge on a local optimum and then, begins to diverge from the global optimal solution. The problem with the DP approach is that it requires very large dimensions and so many programming efforts. These classical methods are not able to locate the global optimum solution because of the presence of many non-linear equations like the non-smooth cost function, ramp rate limit, and discontinues Prohibited Operating Zones (POZs). Also, due to non-linearity of the ELD problem, many of the classical optimization techniques cannot reach the
global optimal solution and tend to diverge at a local optimum. Therefore, it is imperative to develop an optimization technique that can overcome drawbacks of the classical methods and give the global optimum solution in the lowest computational time. Many artificial intelligence algorithms like the Hopfield neural network [7] have been used in solving the ELD problem to overcome the mentioned drawbacks. The problem with the artificial intelligence algorithms is that they take a huge number of iterations to reach the global optimum solution. Hence, more time is required to reach the global solution. The computer technology has helped to develop many population based heuristic optimization techniques, e.g., Differential Evolution (DE) [8], Evolutionary Programming (EP) [9], Hybrid Evolutionary Programming (HEP) [10], Particle Swarm Optimization (PSO) [11], Civilized Swarm Optimization (CSO) [12], Craziness based PSO (CRPSO) [13], Hybrid PSO (HPSO) [14], Modified PSO (MPSO) [15], Genetic Algorithm (GA) [16], Hybrid GA (HGA) [17], Adaptive Real Coded GA (ARCGA) [18], Bacteria Foraging Optimization (BFO) [19], Modified BFO (MBFO) [20], modified Artificial Bee Colony (ABC) [21], Seeker Optimization Algorithm (SOA) [22], Ant Colony Optimization (ACO) [23], Tabu Search (TS) [24], Backtracking Search Algorithm (BSA) [25], and Teaching Learning Based Optimization (TLBO) [26] for solving ELD problems. Roy and Bhattacharjee [27] and Zarei et al. [28] solved the problem of unit commitment. Also, an optimization technique based on trigonometric functions, called Sine Cosine Algorithm (SCA), was used to solve the problem of unit commitment [29]. Apart from electrical problems, SCA has also been used to solve the engineering design problems [30]. Even the problem of short-term hydrothermal scheduling has been solved using the SCA technique [31]. An enhanced version of the PSO was proposed to solve the problem of ELD [32]. A new maximum likelihood optimism technique was presented to solve the ELD problem [33]. Group Leader Optimization (GLO) [34] with special ability to solve the non-linear and the non-quadric equations with greater ease was proposed. However, some of the above-mentioned algorithms have difficulties in finding the local optimum and some have problems in finding the global optimum solution. Therefore, to overcome such kind of problems, a new powerful optimization technique is needed.

The SCA [35] was proposed based on the trigonometric functions sine and cosine to find the fitness function of a search agent. In this method, the search agent having the maximum fitness is made to move towards the global optimum. The superiority of this method is the exploration and exploitation property it utilizes to reach the global optimal value in the lowest computational time. This characteristic helps the method to avoid the local optima and move directly towards the global optimum value.

To give a better solution to the ELD problem by implementing the trigonometric functions in the algorithm, we applied SCA to solving the problem. ELD is also a problem related to power system optimization in which the fuel cost has to be minimized. These are elaborately described in the following sections.

Section 2 formulates various ELD problems with different feasible constraints. The conception of the SCA is described in Section 3. The performance of the SCA under various test systems and the simulation studies are discussed in Section 4. Finally, the conclusions are presented in Section 5.

2. Problem formulation

The ELD problems are either convex or non-convex with some linear and nonlinear constraints in different applications.

The objective function of ELD with quadratic cost function is given as follows [36]:

\[
F_{Cost} = \min \sum_{a=1}^{N} (\alpha_a + \beta_a P_a + \gamma_a P_a^2).
\]  

(1)

For more realistic and practical application of ELD problems, the smooth quadratic cost function can be modified by adding sinusoidal terms of ripples input-output curve with valve point effects. The valve point effect based cost function of ELD is given below [36]:

\[
F_{Cost} = \min \sum_{a=1}^{N} (\alpha_a + \beta_a P_a + \gamma_a P_a^2 + \left| \delta_a \right| \\
\times \sin \left\{ \varepsilon_a \left( P_{a_{\text{min}}} - P_a \right) \right\}).
\]  

(2)

where \(\alpha_a, \beta_a, \gamma_a, \delta_a, \) and \(\varepsilon_a\) are the constant values of fuel cost function. \(N\) is the total number of thermal generators. Power generation by each generator is indicated by \(P_a\). Lower limit and higher limit of power generation are characterized by \(P_{a_{\text{min}}}\) and \(P_{a_{\text{max}}}\), respectively. Power generation by each unit is determined by the capacity constraint of the following generator:

\[
P_{a_{\text{min}}} \leq P_a \leq P_{a_{\text{max}}}.
\]  

(3)

After identifying the inequality constraint of ELD problem, its equality or real power balance constraint can be formulated as:

\[
\sum_{a=1}^{N} P_a - P_D - P_{Loss} = 0,
\]  

(4)

where \(P_D\) is total active power demand of the system...
and total transmission loss $P_{Loss}$ is calculated using the B-matrix loss coefficients, which are expressed as [36]:

$$P_{Loss} = \sum_{a=1}^{N} \sum_{b=1}^{N} P_a B_{ab} P_b + \sum_{a=1}^{N} B_{0a} P_a + B_{00}. \quad (5)$$

Ramp rate limit is another constraint considered in ELD problems to increase the life of generators as given below:

$$P_a - P_{a0} \leq UR_a \quad \text{(as generation rises)}, \quad (6)$$

$$P_{a0} - P_a \leq DR_a \quad \text{(as generation declines)}, \quad (7)$$

and:

$$\max (P^\text{min}_a, P_{a0} - DR_a) \leq P_a \leq \min (P^\text{max}_a, P_{a0} + UR_a), \quad (8)$$

where $P_{a0}$ is power generation of the $a$th previous interval. Also, $UR_a$ and $DR_a$ are the up-ramp and down-ramp limits, respectively.

For different faults in the operation of the machines, boilers, feed pumps, and steam valve as well as the vibration in the bearing, the POZ constraint is considered in the ELD problems. Mathematically, POZ can be expressed as given below:

$$P^\text{min}_a \leq P_a \leq P^\text{max}_a, \quad j = 1, 2, \ldots, n_a, \quad (9)$$

where $P^\text{min}_a$ and $P^\text{max}_a$ are respectively the upper and lower limits of the $j$th POZ of the $a$th unit. The total number of generators under POZ is denoted by $n_a$.

Specifying slack generator is one of the important parts of ELD problem formulation. Let $N$ be the total number of number of generators. Initially calculate the number of $(N - 1)$ power generations randomly based on Eq. (3) and Eqs. (6)-(9). The remaining generator ($N$th), which is called slack generator, has to be identified using Eq. (4). The value of slack generator is given below:

$$P_N = P_D - \sum_{a=1}^{N-1} P_a \quad \text{(without transmission losses)}, \quad (10)$$

$$P_N = P_D + P_{Loss} - \sum_{a=1}^{N-1} P_a \quad \text{(with transmission losses)}. \quad (11)$$

Transmission loss ($P_{Loss}$) is also dependent on power generation based on Eq. (5). Therefore, Eq. (11) is further modified as follows:

$$BNP_N^2 + P_N \left( \frac{2}{N-1} \sum_{a=1}^{N-1} B_{Na} P_a + \sum_{a=1}^{N-1} B_{0N} - 1 \right) \quad (12)$$

$$+ \left( P_D + \sum_{a=1}^{N-1} \sum_{b=1}^{N-1} P_a B_{ab} P_b \right) \quad (12)$$

$$+ \left( \sum_{a=1}^{N-1} B_{0a} P_a - \sum_{a=1}^{N-1} P_a + B_{00} \right) = 0. \quad (12)$$

3. Sine Cosine Algorithm (SCA)

SCA [35] is a population-based optimization technique. This technique starts with a random number of search agents. The optimization process is divided into two phases, namely exploration and exploitation. In the phase of exploration, SCA combines all the random numbers of solutions in a set of solutions quickly with a higher rate of randomness so that it can find those regions of search space where there is a higher probability to find the global solution. On the other hand, in the phase of exploitation, there are slow changes in the random solutions and low random variations compared to the exploration phase.

In the SCA, there are four main parameters, namely $e_1$, $e_2$, $e_3$, and $e_4$. The parameter $e_1$ indicates the next position, which could be between the solution and the destination or even outside it. The parameter $e_2$ decides the distance that the search agents have to cover in the direction of the solution. The parameter $e_3$ helps to decide the weighting factor for the destination. Weighting factors greater than one indicate increased emphasis on a destination and lower than one represent decreased emphasis. The parameter $e_4$ equally switches between the sine and cosine components. Due to the involving property of switching between the sine and cosine functions, the algorithm is known as the SCA. The sine and the cosine functions have the tendency to re-position themselves around the global solution.

To update the result in every iteration, the following two equations are used:

$$X_a^{t+1} = X_a^t + e_1 \sin(e_2) \times |e_3 \times PO_a^d - X_a^d|, \quad (13)$$

$$X_a^{t+1} = X_a^t + e_1 \cos(e_2) \times |e_3 \times PO_a^d - X_a^d|, \quad (14)$$

where $e_1$, $e_2$, and $e_3$ are constant variables. The modification is done using variable $e_4$ given a random value within $[0, 1]$ through the following equation:

$$X_a^{t+1} =$$

$$\begin{cases} X_a^t + e_1 \sin(e_2) \times |e_3 \times PO_a^d - X_a^d|; & e_4 \leq 0.5 \\
X_a^t + e_1 \cos(e_2) \times |e_3 \times PO_a^d - X_a^d|; & e_4 > 0.5 \end{cases} \quad (15)$$

$X_a^{t+1}$ is the position of the search agent in the current $(t+1)$th iteration and 8th dimension, and $X_a^t$ is the position of the search agent in the previous $t$th iteration and 8th dimension. $PO_a^d$ is the position
of the destination location up to the $t$th iteration. The main benefits of SCA over other present effective optimization techniques are the following:

1. This algorithm works upon the set of solutions that it has created randomly, so that it can avoid the local optima and benefit from the favourable exploration property. Such feature cannot be found in other classical algorithms;

2. When the sine and cosine functions give a value lower, or greater, than 1, then different regions of the entire search space are explored for finding the global solution;

3. When the sine and cosine functions give a value between 1 and −1, then the search agents will exploit the present regions;

4. The entire range of the sine and cosine functions is utilized to make a smooth transition from the exploration to the exploitation phase;

5. The global solution obtained by the SCA is stored in a variable at a known destination point; thus, the global solution is never lost.

The authors [29–31] have already proved the versatile advantages of SCA algorithm in different domains. The sequential steps of SCA are given below:

3.1. Sequential steps of SCA

(i) In the initialization process, the lower bound and the upper bound values are assigned to each search agent randomly. Also, the total number of iterations is decided and then, the number of search agents to be used in the algorithm is identified;

(ii) The objection function of the system is calculated. This function depends on the independent variables given by the user;

(iii) If the value of the fitness function obtained in the present iteration is lower than that in the previous iteration, then it can be considered as the local best. Then, the sine and cosine functions start processing. Initially, the parameters of SCA are assigned fixed values and as the iterations increase, the values of these four parameters keep on changing. Here, the parameter $e_1$ decides the direction of movement of the search agent in the search space while the parameter $e_2$ decides the distance that a particular search agent will move in a particular direction given by parameter $e_1$. The parameter $e_3$ assigns a random weighting factor to a particular search agent, which decides its importance among the searching criteria. Finally, the parameter $e_4$ equally switches between the sine and cosine functions;

(iv) Using SCA algorithm, the changed values of the search agents have to be checked with regard to different constraints. If there is any violation, then their values are fixed with their boundary conditions;

(v) As the iteration changes, the values of the four parameters also change and the search agents move towards the global optimum value together. After every iteration, the fitness values of the search agents also change. The nearest search agent to the global optimum value has the highest fitness. In this way, the search agents will move in the search space and explore it entirely for the optimized value. Once the location of the optimized value is known to the search agent, the exploitation phase will be started. Now, the search agents, instead of moving in the entire search space, will exploit the regions where the results are promising. In this way, the search agents tend to move towards the global optimum value. Once the identified number of iterations is reached or the value of the cost function is obtained within the tolerance limit, the iteration is terminated. The result obtained is considered as the sub-global value;

(vi) Once the final iteration is performed, the algorithm is terminated and the search agents having the highest fitness are considered as the nearest to the global optimum value.

3.1.1. Consecutive steps of SCA algorithm for the ELD problem

In this subsection, the steps to solve the ELD problem by the implementation of SCA are explained. The flowchart for the implementation is shown in Figure 1. The steps for solving the ELD problem are the following:

(i) Initialization of various parameters takes place in the first step. Different variables such as lower bound, upper bound, total power demand $P_D$, etc. are initialized. The total number of generators is denoted by $N$ and total number of search agents is denoted by $P_{size}$.

The search agent matrix is represented as:

$$ X_{ij} = [X_1, X_2, X_3, \ldots, X_{P_{size}}] $$

where $i = 1, 2, 3, \ldots, P_{size}$. In the ELD problem, the search agent matrix is considered for active power generation and represented as follows:

$$ [X_{ij}] = [X_{1i}, X_{2i}, X_{3i}, \ldots, X_{Ni}] = [P_{1i}, P_{2i}, P_{3i}, \ldots, P_{Ni}] = [P_{ij}], $$

where $N$ is the number of generators;
begin at this point. The values for the main four parameters of the algorithm are assigned to the concerned variables, i.e., $e_1$ to $e_4$. These values help the movement of the search agent ($X_{ij}$) (i.e., power generation ($P_{ij}$)) in the search space. Using Eqs. (13)–(15), the movement of search agents takes place in the search space;

(v) If the value of parameter $e_1$ is greater than 1, then the search agent has to move in the direction opposite to its current one; but if the value of $e_1$ is less than 1, then the search agent has to move in the same specified direction. Similarly, the parameter $e_2$ will determine how much distance a particular search agent has to move in the specified direction. Also, the parameter $e_3$ will give the weighting factor to the search agent based upon its proximity to the optimized value;

(vi) Now, the new values of power generation are obtained. These new values are checked for the constraints given in Eq. (3) and Eqs. (6)–(9). If various effects like the ramp rate limit and the POZ are considered, then the equation should be satisfied based on Eqs. (6)–(9), respectively. If any variable violates any of these constraints, then its upper or lower value is considered. The slack value of power generation can be calculated based on Eqs. (11) and (12). If there are any violations of any inequality constraint among Eq. (3) and Eqs. (6)–(9) that are valid for the slack generator, then step (ii) onward is repeated. This process will continue until the ultimate set of power generation matrix is formed;

(vii) The new objective function of fuel cost can be calculated based on the newly generated power generation matrix;

(viii) Now, the current objective values are compared with the values obtained in the previous iterations. If the present objective value is lower than the previous one, it is treated as the best local optimal value. Otherwise, i.e., if it is not lower than the previous value, then the previous value takes the position of the newly generated value in the power generation matrix. Now, the objective function value obtained in the present iteration will be compared with all other values obtained in various iterations and, finally, the minimum value will be made the global optimum value. This global optimum value will be stored in a different memory location;

(ix) In the next iteration, step (ii) and the following ones are repeated. When a predetermined

---

**Figure 1.** Sine Cosine Algorithm (SCA) Flowchart.
Table 1. Comparison of the optimum power output and fuel costs obtained by Sine Cosine Algorithm (SCA) and other techniques for 13-unit test system.

| Unit   | Power output (MW) |
|--------|-------------------|
|        | SCA   | BSA [25] | SDE [37] | ORCCRO [39] | OFWO [40] | FPSOOGSA [41] |
| $P_1$  | 628.3179 | 628.3178 | 628.32 | 628.32 | 628.3185 | 628.3185 |
| $P_2$  | 299.1992 | 299.1947 | 299.20 | 299.20 | 299.1989 | 299.1993 |
| $P_3$  | 297.4468 | 297.4764 | 299.20 | 299.20 | 299.1991 | 299.1993 |
| $P_4$  | 159.7327 | 159.7322 | 159.73 | 159.73 | 159.7331 | 159.7331 |
| $P_5$  | 159.7327 | 159.7330 | 159.73 | 159.73 | 159.7331 | 159.7331 |
| $P_6$  | 159.7328 | 159.7328 | 159.73 | 159.73 | 159.7331 | 159.7331 |
| $P_7$  | 159.7331 | 159.7318 | 159.73 | 159.73 | 159.7330 | 159.7331 |
| $P_8$  | 159.7325 | 159.7329 | 159.73 | 159.73 | 159.7331 | 159.7331 |
| $P_9$  | 159.7328 | 159.7286 | 159.73 | 159.73 | 159.7330 | 159.7331 |
| $P_{10}$ | 77.3095 | 77.3045 | 77.40 | 77.40 | 77.3953 | 76.9368 |
| $P_{11}$ | 114.7993 | 114.7992 | 113.12 | 112.14 | 113.1079 | 114.2795 |
| $P_{12}$ | 92.3097 | 92.3062 | 92.40 | 92.40 | 92.3594 | 92.4248 |
| $P_{13}$ | 92.4000 | 92.3919 | 92.40 | 92.40 | 92.3911 | 92.2007 |
| Power generation (MW) | **2559.8000** | 2560.3641 | 2560.4300 | 2559.43 | 2560.3686 | 2560.7765 |
| Transmission loss (MW) | **39.8000** | 39.8006 | 40.43 | 39.43 | 40.3686 | 40.7765 |
| Fuel cost ($/hr.) | **2451.26085** | 2451.26654 | 2451.848 | 2451.91 | 2451.83 | 2451.53543 |

The algorithm of SCA is presented in Figure 1.

4. Simulations and results

To prove the effectiveness of the SCA, four sets of experiments have been conducted and the final results are compared with the results of various existing methods in tabular and graphical forms.

The SCA algorithm has been applied to four different test systems with varying degrees of complexity to verify its effectiveness and feasibility. The program is written in MATLAB-2017B language and executed on a 1.7 GHz Intel core i3 personal computer with 4-GB RAM.

**Test case 1**: 13 generator units have been considered in test system 1 with transmission losses. The input data for the transmission loss is taken from the study by Srinivasa and Vaisakh [37]. The total power demand is 2520 MW. The input data is taken from Sinha et al. [38] and the system runs for 400 iterations. The number of search agents used is 50 in this case. In test case 1, the results of the SCA algorithm are compared with those of the Oppositional Real Coded Chemical Reaction Optimization (ORCCRO) [39] and Stochastic Differential Equation (SDE) [37] optimization techniques. It can be seen from the graph and the table that the minimum cost is first reached by using SCA algorithm and the rest of the algorithms take more time as compared to SCA. According to Table 1, the minimum fuel cost for 13 generator units in the proposed algorithm is 24512.6085 $/hr$, which is better than the those in SDE [37] and ORCCRO [39]. The minimum, maximum, and average fuel costs obtained after 50 trials are presented in Table 2. The comparison of the convergence characteristics of the SCA with those of SDE and ORCCRO is presented in Figure 2. The comparison of the convergence characteristics of the SCA with those of SDE and ORCCRO is presented in Figure 2. The comparison of the convergence characteristics of the SCA with those of SDE and ORCCRO is presented in Figure 2.
Table 2. Minimum, maximum, and average costs obtained by Sine Cosine Algorithm (SCA) and various optimization techniques for 13 generator units (50 trials).

| Method   | Maximum   | Minimum   | Average   | Time/iteration (s) | Number of hits to minimum solution |
|----------|-----------|-----------|-----------|-------------------|-----------------------------------|
| SCA      | 24512.6085 | 24512.6085 | 24512.6085 | 0.041             | 50                                |
| BSA [25] | 24512.6654 | 24512.6654 | 24512.6654 | 0.035             | 50                                |
| ORCRO [39]| 24513.91   | 24513.91   | 24513.91   | 0.04              | 50                                |
| SDE [37] | 24519.74   | 24514.68   | 24516.23   | NA*               | 21                                |
| BBO [39] | 24516.09   | 24515.21   | 24515.32   | 0.15              | 44                                |
| DE/BBO [39]| 24515.98  | 24514.97   | 24515.05   | 0.11              | 46                                |

*NA: Not Available.

Figure 3. Comparison of SCA, BBO, and NEW PSO for 38 generating units.

Test case 2: In this system, 38 units of generators are considered and transmission loss is neglected. The total load demand is 6000 MW. The minimum fuel cost has been calculated using SCA. The input data is taken from Sydul [42] and the system is run in 400 iterations. Fifty search agents are used in this case. The final results obtained by SCA are compared with those by Biogeography Based Optimization (BBO) [43], DE/BBO [43], New PSO [43], and PSO Time Varying Acceleration Coefficients (TVAC) [43]. It is clear from the tabular and graphical data that the best result is obtained by SCA in the minimum computational time. The best solutions obtained by various optimization techniques are presented in Table 3. The minimum, maximum, and average fuel costs by other optimization techniques after 50 trials are stated in Table 4. The comparison of the convergence characteristics of SCA with those of BBO and NEW PSO is given in Figure 3. The comparison of the convergence characteristics of SCA with those of GA/PSO [39] and SDE [37] is illustrated in Figure 4. Looking at the tabular and graphical data, it is clear that the minimum fuel cost obtained by the SCA is better than those by other techniques like GA/PSO [39], DE/BBO [43], SDE [37], and BBO [43]. The net power delivered to the system is 10499.9999 MW. Hence, the accuracy of the result is 99.9999% based on Eq. (4).

Figure 4. Comparison of Sine Cosine Algorithm (SCA), GA/PSO, and Stochastic Differential Equation (SDE) for 40 generating units.

Test case 3: In this case, 40 generator units are considered and their transmission losses have been taken into consideration. The total power demand is 10500 MW. The input data is taken from Sinha et al. [38] and the system is run in 400 iterations. Fifty search agents are used in this case. Only valve-point loading effect is considered as a constraint for this test case. The B-coefficients for the transmission losses in this system have been taken from the B-coefficients of the 6-generator test system [44] by multiplication of rows and columns up to 40 units. The comparison of the optimum fuel costs obtained using various optimization techniques is given in Table 5. Table 6 illustrates the minimum, maximum, and average fuel costs of various optimization techniques after 50 trials. The comparison of the convergence characteristics of the SCA with those of GA/PSO [39] and SDE [37] is illustrated in Figure 4. Looking at the tabular and graphical data, it is clear that the minimum fuel cost obtained by the SCA is better than those by other techniques like GA/PSO [39], DE/BBO [43], SDE [37], and BBO [43]. The net power delivered to the system is 10499.9999 MW. Hence, the accuracy of the result is 99.9999% based on Eq. (4).
Table 3. Comparison of the optimum power output and fuel costs obtained by Sine Cosine Algorithm (SCA) and other techniques for 38-unit test system.

| Unit | SCA     | BBO [43] | DE/BBO [43] | NEW PSO [43] | PSO TVAC [43] |
|------|---------|----------|-------------|--------------|--------------|
| $P_1$ | 426.69880 | 422.2935 | 426.6060    | 550.000      | 443.659      |
| $P_2$ | 426.360334 | 422.1179 | 426.6060    | 512.2033     | 342.9656     |
| $P_3$ | 429.671911 | 435.7794 | 429.6631    | 485.7333     | 433.1177     |
| $P_4$ | 429.740739 | 445.4819 | 429.6631    | 391.0833     | 500.000      |
| $P_5$ | 429.674882 | 428.4757 | 429.6631    | 443.8466     | 410.3399     |
| $P_6$ | 429.673000 | 428.6492 | 429.6631    | 358.3088     | 492.2864     |
| $P_7$ | 429.668089 | 428.1912 | 429.6631    | 415.7298     | 409.4835     |
| $P_8$ | 429.618641 | 429.9006 | 429.6631    | 320.8165     | 446.0795     |
| $P_9$ | 114.000000 | 115.9049 | 114.0000    | 115.3478     | 119.5663     |
| $P_{10}$ | 114.000000 | 114.1153 | 114.0000    | 204.4222     | 137.2724     |
| $P_{11}$ | 119.76933  | 115.4186 | 119.7680    | 114.0000     | 138.9332     |
| $P_{12}$ | 127.08847  | 127.5114 | 127.0728    | 249.1978     | 155.4010     |
| $P_{13}$ | 110.000000 | 110.0099 | 110.0000    | 118.8866     | 121.7199     |
| $P_{14}$ | 90.000000  | 90.0217  | 90.0000     | 102.8023     | 90.9246      |
| $P_{15}$ | 82.000000  | 82.0000  | 82.0000     | 89.0390      | 97.9411      |
| $P_{16}$ | 120.000000 | 120.0384 | 120.0000    | 120.0000     | 128.1066     |
| $P_{17}$ | 159.61791  | 160.3038 | 159.5980    | 156.5620     | 189.1086     |
| $P_{18}$ | 65.000000  | 65.0001  | 65.0000     | 81.2657      | 65.0000      |
| $P_{19}$ | 65.000000  | 65.0001  | 65.0000     | 65.0411      | 65.0000      |
| $P_{20}$ | 271.999999 | 271.9995 | 272.0000    | 151.1041     | 267.4222     |
| $P_{21}$ | 271.99998  | 271.8726 | 272.0000    | 226.3414     | 221.3834     |
| $P_{22}$ | 259.99994  | 259.7320 | 260.0000    | 209.2986     | 130.8041     |
| $P_{23}$ | 130.632515 | 125.9930 | 130.6486    | 85.7198      | 124.3659     |
| $P_{24}$ | 10.00008   | 10.4143  | 10.0000     | 10.0000      | 11.535       |
| $P_{25}$ | 113.278756 | 109.4177 | 113.3050    | 60.0000      | 77.1030      |
| $P_{26}$ | 88.092495  | 89.3772  | 88.0669     | 90.4890      | 55.0181      |
| $P_{27}$ | 37.511273  | 36.4110  | 37.5051     | 30.9670      | 75.0000      |
| $P_{28}$ | 20.000000  | 20.0098  | 20.0000     | 20.0000      | 21.6280      |
| $P_{29}$ | 20.000000  | 20.0098  | 20.0000     | 20.0095      | 29.8299      |
| $P_{30}$ | 20.000000  | 20.0000  | 20.0000     | 22.8100      | 20.3264      |
| $P_{31}$ | 20.000000  | 20.0000  | 20.0000     | 20.0000      | 20.0000      |
| $P_{32}$ | 20.000000  | 20.0033  | 20.0000     | 20.4100      | 21.8400      |
| $P_{33}$ | 25.000000  | 25.0066  | 25.0000     | 25.0000      | 25.6200      |
| $P_{34}$ | 18.000000  | 18.0222  | 18.0000     | 21.3499      | 24.3610      |
| $P_{35}$ | 8.000000   | 8.0000   | 8.0000      | 9.1220       | 9.6670       |
| $P_{36}$ | 25.000000  | 25.0000  | 25.0000     | 25.1840      | 25.0000      |
| $P_{37}$ | 21.787463  | 22.0005  | 21.7820     | 20.0000      | 31.6412      |
| $P_{38}$ | 21.059227  | 20.6076  | 21.0621     | 25.1040      | 29.9350      |

Fuel cost ($/hr.) \textbf{9417235.7919} 9417633.6376 9417235.7863 9516448.312 9500448.307

Table 4. Minimum, maximum, and average, fuel costs obtained by SCA and various optimization techniques for test system 2 (50 trials).

| Method     | Generation cost ($/hr.) | Time/iteration (s) | Number of hits to minimum solution |
|------------|-------------------------|-------------------|-----------------------------------|
| SCA        | \textbf{9417235.7919}   | 5.24              | 50                                |
| BBO [43]   | 9417658.75              | 12.21             | 41                                |
| DE/BBO [43]| 9417230.83              | 17.75             | 45                                |
| ORCCRO [39]| 9412104.27              | 9.35              | 37                                |
Table 5. Comparison of the optimum power output and fuel costs obtained by Sine Cosine Algorithm (SCA) and other techniques for 40-unit test system.

| Unit | Power output (MW) |         |         |         |
|------|-------------------|---------|---------|---------|
|      | SCA               | GAAP [37]| DE/BBO [43]| SDE [37]| BBO [43]|
| $P_1$| 113.8585          | 114.0000 | 111.0400 | 110.0600 | 112.5400 |
| $P_2$| 114.0000          | 114.0000 | 113.7100 | 112.4100 | 113.2200 |
| $P_3$| 119.3004          | 120.0000 | 118.6400 | 120.0000 | 119.5100 |
| $P_4$| 183.3389          | 190.0000 | 189.4900 | 188.7200 | 188.3700 |
| $P_5$| 91.7652           | 97.0000  | 86.3200  | 85.9100  | 90.4100  |
| $P_6$| 139.9816          | 140.0000 | 139.8800 | 140.0000 | 139.0500 |
| $P_7$| 299.5148          | 300.0000 | 299.8600 | 250.1900 | 294.9700 |
| $P_8$| 299.1356          | 300.0000 | 285.4200 | 290.6800 | 299.1800 |
| $P_9$| 297.6808          | 300.0000 | 296.2900 | 300.0000 | 296.4600 |
| $P_{10}$| 279.1599       | 205.2000 | 285.0700 | 282.0100 | 279.8900 |
| $P_{11}$| 171.4666       | 230.3000 | 164.6900 | 180.8200 | 160.1500 |
| $P_{12}$| 94.4916        | 204.7300 | 94.0000  | 168.7400 | 96.7400  |
| $P_{13}$| 485.0345      | 346.4800 | 486.3000 | 469.9600 | 484.0400 |
| $P_{14}$| 482.8777      | 434.3200 | 480.7000 | 484.1700 | 483.3200 |
| $P_{15}$| 481.0869      | 331.3400 | 480.6600 | 487.7300 | 483.7700 |
| $P_{16}$| 481.9795      | 440.2300 | 485.0500 | 482.3000 | 483.3000 |
| $P_{17}$| 489.6806      | 500.0000 | 487.9400 | 499.6400 | 490.8300 |
| $P_{18}$| 488.7178      | 500.0000 | 491.0900 | 491.3200 | 492.1900 |
| $P_{19}$| 515.9524      | 550.0000 | 511.7900 | 510.4700 | 511.2800 |
| $P_{20}$| 511.6585      | 550.0000 | 544.8900 | 542.0400 | 521.5500 |
| $P_{21}$| 532.3453      | 550.0000 | 528.9200 | 544.8100 | 526.4200 |
| $P_{22}$| 549.9726      | 550.0000 | 540.5800 | 550.0000 | 538.3000 |
| $P_{23}$| 523.9532      | 550.0000 | 524.9800 | 550.0000 | 534.7400 |
| $P_{24}$| 527.3965      | 550.0000 | 521.1200 | 528.1600 | 521.2000 |
| $P_{25}$| 523.3733      | 550.0000 | 534.4900 | 524.1600 | 526.1400 |
| $P_{26}$| 527.6279      | 550.0000 | 529.1500 | 539.1000 | 544.3300 |
| $P_{27}$| 10.0009       | 11.4400  | 10.5100  | 10.0000  | 11.5100  |
| $P_{28}$| 11.1190       | 11.5600  | 10.0000  | 10.3700  | 10.2100  |
| $P_{29}$| 10.1184       | 11.4200  | 10.0000  | 10.0000  | 10.7100  |
| $P_{30}$| 86.9830       | 97.0000  | 90.6600  | 96.1000  | 88.2800  |
| $P_{31}$| 189.9885      | 190.0000 | 189.8200 | 185.3300 | 189.8400 |
| $P_{32}$| 189.9150      | 190.0000 | 187.6900 | 189.5400 | 189.9400 |
| $P_{33}$| 189.9535      | 190.0000 | 189.9700 | 189.9600 | 189.1300 |
| $P_{34}$| 199.9110      | 200.0000 | 199.8300 | 199.9000 | 198.0700 |
| $P_{35}$| 197.9306      | 200.0000 | 199.9300 | 196.2500 | 199.9200 |
| $P_{36}$| 165.3294      | 200.0000 | 163.0300 | 185.8500 | 194.2500 |
| $P_{37}$| 109.4111      | 110.0000 | 109.8500 | 109.7200 | 109.4300 |
| $P_{38}$| 109.9582      | 110.0000 | 109.2600 | 110.0000 | 109.5600 |
| $P_{39}$| 109.9271      | 110.0000 | 109.6000 | 95.7100  | 109.6200 |
| $P_{40}$| 547.6036      | 550.0000 | 543.2300 | 532.4700 | 527.8200 |

Fuel cost ($/hr.) 136653.0219 139864.96 136950.77 138157.46 137026.82
Power generation (MW) 11459.5499 11545.0600 11147.8300 11474.4300 11470.0000
Transmission loss (MW) 959.5500 1045.0600 957.8300 974.4300 970.3700
Table 6. Minimum, maximum, and average fuel costs obtained by Sine Cosine Algorithm (SCA) and various optimization techniques for 40 generator units (50 trials).

| Method     | Generation cost ($/hr) | Time/iteration (s) | Number of hits to minimum solution |
|------------|------------------------|--------------------|-----------------------------------|
| SCA        | 136653.10 136653.02 136653.02 | 0.07               | 48                                |
| BBO [43]   | 137026.82 137116.58     | 0.2                | 41                                |
| DE/BBO [43]| 136950.77 136966.77     | 0.16               | 45                                |
| ORCCRO [39]| 136855.19 136855.19     | 0.07               | 50                                |

Figure 5. Decreasing cost for 140 generator units using Sine Cosine Algorithm (SCA).

Test case 4: In this case, 140 generator units have been considered. The transmission losses have been neglected in this test system. The total load demand is 49342 MW and the input data is taken from Jong-Bae et al. [45]. The large and complicated test system of 140 generating units is considered with valve point loading effects, ramp rate limits, and POZs. The system is made to run for 1000 iterations. Fifty search agents are used in this case. Table 7 shows the power generation of each of the 140 generators using the SCA. Table 8 compares the minimum, maximum, and average fuel costs obtained using various optimization techniques after 50 trials. The results in Table 8 prove that the minimum fuel cost obtained by SCA is much better than those by other algorithms. The net power delivered to the system is 49342.0006 MW. Hence, the accuracy of the result is 99.9999% based on Eq. (4) when transmission loss is neglected. The convergence characteristics for the SCA are shown in Figure 5.

4.1. Tuning of parameters for the SCA
To obtain the optimized solution by the use of SCA, it is imperative to obtain the proper values of parameters $e_1$, $e_2$, and $e_3$. Tuning of these parameters is very important for obtaining the optimized solution. Different values of these parameters give different fuel costs. For one single value of one parameter, other parameters have to be varied in all possible combinations. For a single value of $e_1$, different combinations of $e_2$ and $e_3$ have been tried to obtain the minimum fuel cost. A summary of the results for the 140-generator system is provided in Table 9.

Also, using a too large or too small number of search agents for screening the search space does not lead to the optimized solution. Thus, only a specific number of search agents will help to obtain the optimized solution. For each number of search agents, 50 trials have been run. The trials show that the number of 50 search agents end in achieving the optimized fuel cost. For other numbers of search agents, no significant improvement in the fuel cost is observed. Moreover, beyond the number of 50 search agents, the simulation time also increases. The best output obtained by SCA for each number of search agents in the 140-generator system is presented in Table 10.

The optimum values of the tuned parameters are $P_{st,x} = 50$, $e_1 = 0.55$, $e_2 = 0.15$, $e_3 = 0.72$, and $e_4 = 0.5$.

5. Comparative study
5.1. Quality of solution
Tables 1, 3, 5, and 7 show that the fuel costs obtained by the SCA are the lowest among all optimization techniques. Also, the cost obtained by the SCA is better than the cost obtained by many previously developed algorithms. For example, in test case 1, the minimum fuel cost obtained by the SCA is 24512.6085 $/hr, which is lower than those obtained by SDE and ORCCRO. The comparison has been made in both cases of taking the transmission losses into account and neglecting them. Thus, it is clear that the quality of the solution is the best when SCA is applied.

5.2. Robustness
The robustness of any optimization algorithm cannot be judged by only running in for a single time. A number of trials should be carried out in order to prove the robustness of any optimization technique. It is evident form Tables 2 and 4 that SCA achieves the global optimal solution for all the 50 trials in various
Table 7. Optimum power output and fuel cost obtained by Sine Cosine Algorithm (SCA) for 140-unit test system.

| Unit | Power output (MW) | Unit | Power output (MW) | Unit | Power output (MW) |
|------|-------------------|------|-------------------|------|-------------------|
| $P_1$ | 110.8395 | $P_{36}$ | 499.9997 | $P_{71}$ | 140.7389 | $P_{106}$ | 880.9000 |
| $P_2$ | 163.9999 | $P_{37}$ | 240.9999 | $P_{72}$ | 388.4824 | $P_{107}$ | 873.6068 |
| $P_3$ | 189.9518 | $P_{38}$ | 240.9424 | $P_{73}$ | 230.9036 | $P_{108}$ | 877.4000 |
| $P_4$ | 189.9612 | $P_{39}$ | 773.9925 | $P_{74}$ | 271.6243 | $P_{109}$ | 871.6999 |
| $P_5$ | 168.3794 | $P_{40}$ | 768.9999 | $P_{75}$ | 175.9105 | $P_{110}$ | 864.7967 |
| $P_6$ | 186.3858 | $P_{41}$ | 3.161799 | $P_{76}$ | 293.5256 | $P_{111}$ | 881.9908 |
| $P_7$ | 489.9999 | $P_{42}$ | 3.072809 | $P_{77}$ | 306.7155 | $P_{112}$ | 94.20333 |
| $P_8$ | 489.9997 | $P_{43}$ | 230.2171 | $P_{78}$ | 385.5398 | $P_{113}$ | 95.06407 |
| $P_9$ | 496.0000 | $P_{44}$ | 249.8248 | $P_{79}$ | 530.9998 | $P_{114}$ | 94.32603 |
| $P_{10}$ | 496.0000 | $P_{45}$ | 247.436 | $P_{80}$ | 530.9998 | $P_{115}$ | 244.0719 |
| $P_{11}$ | 495.9894 | $P_{46}$ | 249.2271 | $P_{81}$ | 542.0000 | $P_{116}$ | 245.6768 |
| $P_{12}$ | 495.9999 | $P_{47}$ | 246.1245 | $P_{82}$ | 566.6217 | $P_{117}$ | 245.6193 |
| $P_{13}$ | 505.9871 | $P_{48}$ | 247.803 | $P_{83}$ | 115.1015 | $P_{118}$ | 98.8419 |
| $P_{14}$ | 508.9965 | $P_{49}$ | 246.1036 | $P_{84}$ | 110.5754 | $P_{119}$ | 95.7353 |
| $P_{15}$ | 505.9998 | $P_{50}$ | 246.5329 | $P_{85}$ | 115.9195 | $P_{120}$ | 116.5415 |
| $P_{16}$ | 504.9990 | $P_{51}$ | 165.1967 | $P_{86}$ | 207.117 | $P_{121}$ | 175.1441 |
| $P_{17}$ | 505.9666 | $P_{52}$ | 165.8992 | $P_{87}$ | 207.2333 | $P_{122}$ | 3.6211 |
| $P_{18}$ | 505.9948 | $P_{53}$ | 185.7631 | $P_{88}$ | 176.4165 | $P_{123}$ | 4.0847 |
| $P_{19}$ | 505.0000 | $P_{54}$ | 165.0303 | $P_{89}$ | 175.241 | $P_{124}$ | 15.4299 |
| $P_{20}$ | 504.9951 | $P_{55}$ | 180.1148 | $P_{90}$ | 177.5673 | $P_{125}$ | 9.6570 |
| $P_{21}$ | 504.9971 | $P_{56}$ | 180.9737 | $P_{91}$ | 180.4744 | $P_{126}$ | 16.8826 |
| $P_{22}$ | 504.9874 | $P_{57}$ | 112.9304 | $P_{92}$ | 575.3998 | $P_{127}$ | 10.0005 |
| $P_{23}$ | 504.9936 | $P_{58}$ | 199.5520 | $P_{93}$ | 547.4997 | $P_{128}$ | 112.9878 |
| $P_{24}$ | 504.9997 | $P_{59}$ | 311.9997 | $P_{94}$ | 836.7998 | $P_{129}$ | 4.7148 |
| $P_{25}$ | 537.0000 | $P_{60}$ | 290.2522 | $P_{95}$ | 837.4999 | $P_{130}$ | 5.0210 |
| $P_{26}$ | 536.9998 | $P_{61}$ | 163.5181 | $P_{96}$ | 681.9973 | $P_{131}$ | 5.0062 |
| $P_{27}$ | 548.9997 | $P_{62}$ | 99.08827 | $P_{97}$ | 719.9999 | $P_{132}$ | 50.1757 |
| $P_{28}$ | 548.9996 | $P_{63}$ | 468.563 | $P_{98}$ | 717.9918 | $P_{133}$ | 5.0813 |
| $P_{29}$ | 500.9999 | $P_{64}$ | 510.7641 | $P_{99}$ | 719.9925 | $P_{134}$ | 42.0132 |
| $P_{30}$ | 498.9999 | $P_{65}$ | 489.9999 | $P_{100}$ | 963.9999 | $P_{135}$ | 42.0579 |
| $P_{31}$ | 505.9997 | $P_{66}$ | 201.0382 | $P_{101}$ | 957.9999 | $P_{136}$ | 41.3626 |
| $P_{32}$ | 505.9910 | $P_{67}$ | 488.1348 | $P_{102}$ | 947.8997 | $P_{137}$ | 17.0319 |
| $P_{33}$ | 505.7959 | $P_{68}$ | 485.3448 | $P_{103}$ | 933.9998 | $P_{138}$ | 7.044 |
| $P_{34}$ | 505.9998 | $P_{69}$ | 132.4697 | $P_{104}$ | 934.9996 | $P_{139}$ | 7.0202 |
| $P_{35}$ | 500.0000 | $P_{70}$ | 338.9781 | $P_{105}$ | 876.4997 | $P_{140}$ | 31.3066 |

Total fuel cost = 1658384.8872 $/hr.

Table 8. Minimum, maximum, and average fuel costs obtained by Sine Cosine Algorithm (SCA) and various optimization techniques for 140 generator units (50 trials).

| Method       | Maximum    | Minimum    | Average   | Time/iteration (s) | Number of hits to minimum solution |
|--------------|------------|------------|-----------|--------------------|-------------------------------------|
| SCA          | 1658386.57 | 1658384.88 | 1658385.04 | 50.47              | 45                                  |
| BBO [43]     | 1657809.57 | 1657721.38 | 1657730.72 | 142.5              | 41                                  |
| DE/BBO [43]  | 1657781.72 | 1657736.84 | 1657725.92 | 125.4              | 43                                  |
| RCCRO [36]   | 1657742.97 | 1657690.83 | 1657039.96 | 75.8               | 47                                  |
Table 9. Effect of various parameters on the performance of Sine Cosine Algorithm (SCA).

| $e_1$  | $e_2$  | $e_3$  | $e_4$  | Fuel cost ($/hr$) |
|--------|--------|--------|--------|-------------------|
| 0.16   | 0.41   | 0.14   | 0.5    | 1658479.1876     |
| 0.68   | 0.65   | 0.15   | 0.5    | 1658455.6189     |
| 0.47   | 0.87   | 0.62   | 0.5    | 1658438.3245     |
| 0.57   | 0.54   | 0.25   | 0.5    | 1658420.9452     |
| 0.55   | 0.15   | 0.72   | 0.5    | 1658384.8872     |
| 0.42   | 0.26   | 0.95   | 0.5    | 1658359.5475     |
| 0.94   | 0.32   | 0.84   | 0.5    | 1658456.3225     |
| 0.21   | 0.41   | 0.25   | 0.5    | 1658472.2587     |
| 0.78   | 0.52   | 0.41   | 0.5    | 1658501.3654     |

Table 10. Effect of the number of search agents on the 14-bus generator system.

| No. of search agents | Number of hits to best solution | Simulation time (sec) | Max. cost ($/hr$) | Min. cost ($/hr$) | Average cost ($/hr$) |
|----------------------|---------------------------------|-----------------------|-------------------|-------------------|----------------------|
| 20                   | 32                              | 48.25                 | 1658406.547       | 1658399.254       | 1658401.879          |
| 50                   | 45                              | 50.47                 | 1658386.570       | 1658384.880       | 1658385.04          |
| 100                  | 27                              | 54.36                 | 1658416.235       | 1658406.325       | 1658410.884         |
| 150                  | 19                              | 57.25                 | 1658428.625       | 1658412.658       | 1658422.558         |
| 200                  | 11                              | 62.33                 | 1658468.235       | 1658435.328       | 1658460.995         |

test cases and from Tables 6 and 8 that SCA gives the minimum fuel cost for the maximum number of trials in comparison with other optimization techniques. This proves that the efficiency of the SCA is very high and its performance is superior to other optimization techniques, which in turn confirms the robustness of the algorithm.

5.3. Computational efficiency
The efficiency of any optimization technique is determined by the time the technique takes to reach the global optimal solution. It is clear from Tables 2, 4, 6, and 8 that the computational time taken for one single iteration is the minimum in the SCA as compared to other previously developed optimization techniques. Thus, the SCA gives the global optimal results in the lowest computational time.

6. Conclusion
In this paper, a new algorithm, named Sine Cosine Algorithm (SCA), was proposed to solve Economic Load Dispatch (ELD) problem. To prove the efficiency of the SCA, four test cases were considered and the net fuel cost obtained by SCA was compared with those by other optimization techniques. The results were presented in tabular and graphical forms. They proved that SCA was robust, feasible, and effective as compared to other algorithms in terms of efficiency and computational time. The numerical results also proved that the SCA prevented premature convergence and had a stable convergence characteristic. Hence, by using the exploration and exploitation ability of SCA, the problem of ELD was successfully solved.

References
1. Dhar, R.N. and Mukherjee, P.K. “Reduced-gradient method for economic dispatch”, Electrical Engineers, Proceedings of the Institution, 120(5) pp. 608–610 (1973).
2. Aoki, K. and Satoh, T. “Economic dispatch with network security constraints using parametric quadratic programming”, IEEE Power Engineering Review, 2(12), pp. 37–38 (1982).
3. El-Keib, A.A., Ma, H., and Hart, J.L. “Environmentally constrained economic dispatch using the Lagrangian relaxation method”, IEEE Trans Power Syst, 9(4), pp. 1723–1729 (1994).
4. Ching-Tsong, S. and Chien-Tung, L. “New approach with a Hopfield modeling framework to economic dispatch”, IEEE Transactions on Power Systems, 15(2), pp. 541–545 (2000).
5. Jabr, R.A., Coonick, A.H., and Cory, B.J. “A homogeneous linear programming algorithm for the security constrained economic dispatch problem”, IEEE Transactions on Power Systems, 15(3) pp. 930–936 (2000).
6. Shailti Swamp, K. and Natarajan, A. “Constrained
optimization using evolutionary programming for dynamic economic dispatch”, Proceedings of 2005 International Conference on Intelligent Sensing and Information Processing, pp. 314–319 (2005).

7. Farooqi, M.R., Jain, P., and Niazi, K.R. “Using Hopfield neural network for economic dispatch of power systems”, Proceedings, National Power Engineering Conference, pp. 5–10 (2003).

8. Sheng-Kuan, W., Chih-Wen, L., and Ji-Pyung, C. “Ant direction hybrid differential evolution for solving economic dispatch of power system”, 2006 IEEE International Conference on Systems, Man and Cybernetics, 2(1) pp. 1154–1159 (2006).

9. Ali Ahmad, K., Amir, M., Nasir, S., et al “Fuzzy economic dispatch and spinning reserve allocation using evolutionary programming”, 2008 4th North American Power Symposium Year, pp. 1–5 (2008).

10. Bava, M., Monsef, H., and Navidi, N. A new hybrid approach for unit commitment using Lagrangian relaxation combined with evolutionary and quadratic programming”, 2009 Asia-Pacific Power and Energy Engineering Conference, pp. 1–6 (2009).

11. Yongqiang, W., Jianzhong, Z., Wen, X., et al “Economic load dispatch of hydroelectric power plants using a hybrid particle swarm optimization combined simulation annealing algorithm”, 2010 Second WRI Global Congress on Intelligent Systems, 2(1), pp. 231–234 (2010).

12. Himanshu, A. and Nitin, N. “Civilized swarm optimization for combined heat and power economic emission dispatch”, 2016 7th India International Conference on Power Electronics (IICPE), pp. 1–6 (2016).

13. Chaturvedi, K.T., Pandit, M., and Srivastava, L. “Particle swarm optimization with crazy particles for nonconvex economic dispatch”, Appl Soft Comput, 9(1), pp. 962–969 (2009).

14. Lu, H., Siyanmng, P., Song, Y.H., et al “Experimental study of a new hybrid PSO with mutation for economic dispatch with non smooth cost function”, Int J Electr Power Energy Syst, 32(1) pp. 921–935 (2010).

15. Taher, N., Faramarz, G., and Bahman, B. “Modified adaptive PSO algorithm to solve dynamic economic dispatch”, 2011 IEEE Power Engineering and Automation Conference, 1(1), pp. 108–111 (2011).

16. King, D.J. and Ozveren Warnuso, C.S. “A genetic algorithm based economic dispatch (GAE) with environmental constraint optimisation universities”, Power Engineering Conference (UPEC), Proceedings of 2011 4th International, pp. 1–6 (2011).

17. Jamal, S.A and Jan, K.S. “Solving economic dispatch problem using hybrid GA-PS-SQP method”, IEEE EUROCON, pp. 333–338 (2009).

18. Amjadi, N. and Nasiri-Rad, H. “Solution of non-convex and non-smooth economic dispatch by a new adaptive real coded genetic algorithm”, Expert Syst Appl, pp. 3290–3245 (2009).

19. Hazra, J. and Sinha, A.K. “Environmental constrained economic dispatch using bacteria foraging optimization”, 2008 Joint International Conference on Power System Technology and IEEE Power India Conference, pp. 1–6 (2008).

20. Deblina, M., Sumit, B., and Chandan, K.C. “Multi-objective economic emission load dispatch using modified biogeography based optimization algorithm”, 2016 IEEE 7th Power India International Conference (PIICON), pp. 1–6 (2016).

21. Nagur, P.N., Shubham, R., and Jadhav, H.T. “Modified artificial bee colony algorithm for non-convex economic dispatch problems”, 2012 International Conference on Green Technologies (ICGT), pp. 258–262 (2012).

22. Shaw, B., Mukherjee, V., and Ghoshal, S.P. “Seeker optimisation algorithm: application to the solution of economic load dispatch problems”, IET Generation, Transmission & Distribution, 5(1), pp. 81–91 (2011).

23. Rahmat, N.A., Musirin, I., Abidiin, A.F., et al “Economic load dispatch with valve-point loading effect by using differential evolution immunized ant colony optimization technique”, 2014 Australasian Universities Power Engineering Conference (AUPEC), pp. 1–6 (2014).

24. Khamasawang, S., Pothiya, S., and Boonseng, C. “Distributed tabu search algorithm for solving the economic dispatch problem”, 2004 IEEE Region 10 Conference TENCON 2004, 3(1), pp. 484–487 (2004).

25. Bhattacharjee, K. “Economic dispatch problems using backtracking search optimization”, International Journal of Energy Optimization and Engineering, IGI-Global, 7(2) pp. 39–60 (2018).

26. Bhattacharjee, K., Bhattacharya, A., and Hukler, S. “Teaching learning based optimization for different economic dispatch problems”, International Journal of Science and Technology, 21(3), pp. 870–884 (2014).

27. Roy, S. and Bhattacharjee, K. “The use of krill herd based optimization to solve complex economic load dispatch problems”, 2017 Conference on Emerging Devices and Smart Systems (ICEDSS), pp. 72–78 (2017).

28. Zarei, E., Hemmatpour, M., and Mohammadian, M. “The effects of demand response on security-constrained unit commitment”, Scientia Iranica, 26(3), pp. 1627–1636 (2019). DOI: 10.24200/sci.2017.4536

29. Sandeep, K. and Shushil, P. “A novel sine cosine algorithm for the solution of unit commitment problem”, International Journal of Science, Engineering and Technology Research (IJESETR), 5(12), pp. 3298–3310 (2016).

30. Mohamed, A.T. and Vimal, S. “Multi-objective sine-cosine algorithm (MO-SCA) for multi-objective engineering design problems”, Neural Computing and Applications, 31(4), pp. 1–15 (2017).
31. Sujoy, D., Bhattacharya, A., and A Kumar, C. “Solution of short-term hydrothermal scheduling using sine cosine algorithm”, Soft Computing, 22, pp. 6109–6127 (2018). https://doi.org/10.1007/s00500-017-2695-3
32. Quande, Q., Cheng, S., and Xianghua, C. “Solving non-convex/non-smooth economic load dispatch problems via an enhanced particle swarm optimization”, Applied Soft Computing, 59(1), pp. 229–242 (2017).
33. Sanjoy, R. “The maximum likelihood optima for an economic load dispatch in presence of demand and generation variability”, Energy, 147(1), pp. 915–923 (2018).
34. Roy, S., Bhattacharjee, K., and Bhattacharya, A. “A modern approach to solve of economic load dispatch using group leader optimization technique”, International Journal of Energy Optimization and Engineering, IGI-Global, 6(1), pp. 66–85 (2016).
35. Seyedali, M. “SCA: A sine cosine algorithm for solving optimization problems”, Knowledge-Based Systems, 96, pp. 120–133 (2016).
36. Bhattacharjee, K., Bhattacharya, A., and Halder, S. “Chemical reaction optimisation for different economic dispatch problems”, IET Generation Transmission Distribution, 8(3), pp. 530–541 (2014).
37. Srinivasa, R.A. and Vaisakh, K. “Shuffled differential evolution for large scale economic dispatch”, Electric Power Syst Res, 96, pp. 237–245 (2013).
38. Sinha, N., Chakralarti, R., and Chattopadhyay, P.K. “Evolutionary programming techniques for economic load dispatch”, IEEE Trans Evol Comput, 7(1), pp. 83–94 (2003).
39. Bhattacharjee, K., Bhattacharya, A., and Sumita, H. “Oppositional real coded chemical reaction optimization for different economic dispatch problems”, Electrical Power and Energy Systems, 55, pp. 378–391 (2014).
40. Barisal, A.K. and Prusty, R.C. “Large scale economic dispatch of power systems using oppositional invasive weed optimization”, Applied Soft Computing, 29, pp. 122–137 (2015).
41. Duman, S., Yorulmazer, N., and Altas, I.H. “A novel modified hybrid PSOGSA based on fuzzy logic for non-convex economic dispatch problem with valve-point effect”, Electrical Power and Energy Systems, 64, pp. 121–135 (2015).
42. Syedlu, M. “A very fast and effective non-iterative lambda logic based”, Algorithm for Economic Dispatch of Thermal Units, in Proc. IEEE Region 10 Conf. TENCON, 2(1), pp. 1431–1437 (1999).
43. Bhattacharya, A. and Chattopadhyay, P.K. “Hybrid differential evolution with biogeography-based optimization for solution of economic load dispatch”, IEEE Trans. Power Syst., 25(4), pp. 1955–1964 (2010).
44. Ciorni, I. and Kyriakides, E. “Efficient hybrid optimization solution for the economic dispatch with nonsmooth cost function”, Proc IEEE Power Tech, Bucharest, Romania, 2(1), pp. 1–7 (2009).
45. Jong-Bae, P., Yun-Won, J., and Joong-Rin, S. “An improved particle swarm optimization for nonconvex economic dispatch problems”, IEEE Transactions on Power Systems, 25(1), pp. 156–166 (2010).

Biographies
Nitish Patel received his BE degree from Babaria Institute of Technology under Gujarat Technological University and is currently pursuing his MTech in Electrical Power Systems in the Institute of Technology, Nirma University, Ahmedabad, India. His research interests include artificial intelligence, power system optimization, economic load dispatch, and hydro-thermal applications.

Kuntal Bhattacharjee received a BE degree from BIET, Suri Private College, Burdwan University, and MTech degree from NIT, Durgapur, India, in 2003 and 2005, respectively, both in Electrical Engineering. He is currently in the Electrical Engineering Department, Institute of Technology, Nirma University, India. His research interests include power system optimization, ELD, EELD, and hydro-thermal applications.