Electrical Power and Energy Systems 79 (2016) 298–311

Contents lists available at ScienceDirect

Electrical Power and Energy Systems

journal homepage: www.elsevier.com/locate/ijepes

A new rooted tree optimization algorithm for economic dispatch with valve-point effect

Yacine Labbi [a,b,*], Djilani Ben Attous [a], Hossam A. Gabbar [b], Belkacem Mahdad [c], Aboelsood Zidan [b]

[a] University of El-Oued, Department of Electrical Engineering, El-Oued, Algeria
[b] Faculty of Energy Systems and Nuclear Science, University of Ontario Institute of Technology, UOIT, ON, Canada
[c] Department of Electrical Engineering, University of Biskra, Biskra, Algeria

A R T I C L E I N F O

Article history:
Received 28 September 2015
Received in revised form 31 December 2015
Accepted 14 January 2016
Available online 1 February 2016

Keywords:
Root tree optimization algorithm
RTO
Metaheuristic optimization
Economic dispatch
Valve-point effect

A B S T R A C T

This work proposes a new optimization method called root tree optimization algorithm (RTO). The robustness and efficiency of the proposed RTO algorithm is validated on a 23 standard benchmark non-linear functions and compared with well-known methods by addressing the same problem. Simulation results show effectiveness of the proposed RTO algorithm in term of solution quality and convergence characteristics. In order to evaluate the effectiveness of the proposed method, 3-unit, 30 Bus IEEE, 13-unit and 15-units are used as case studies with incremental fuel cost functions. The constraints include ramp rate limits, prohibited operating zones and the valve point effect. These constraints make the economic dispatch (ED) problem a non-convex minimization problem with constraints. Simulation results obtained by the proposed algorithm are compared with the results obtained using other methods available in the literature. Based on the numerical results, the proposed RTO algorithm is able to provide better solutions than other reported techniques in terms of fuel cost and robustness.

1. Introduction

Economic dispatch (ED) is one of the essential operational planning problems in power engineering [1]. ED aims to schedule the committed generating units' outputs in order to meet the load demand at minimum operating cost while satisfying system constraints [2–4]. In the past twenty years, many researchers have been used heuristic optimization and conventional mathematical techniques to solve ED problem in power systems [5–15]. Artificial intelligence started to be oriented to the simulation of nature (e.g., to the way how the human brain functions and the human operations thinking). Consequently, a new branch of artificial intelligence has emerged which studies and designs intelligent implements in order to adapt intelligently with their environment and to show a cognitive behavior. Thus, a decision can be taken through the recuperation of the acquired information. This intelligence considers the human beings as an example of these implements. The arithmetic intelligence includes the evaluating computing, fuzzy computing and the neural computing [16,17].

By the beginning of ninetieth, researches started to simulate other creatures with limited capacities such as pants, birds, and fishes that show a clever social behavior. In 1990, Dorigo suggested an algorithm of ant colony optimization (ACO) which simulates the ants settlements [18]. In 1995, Kennedy and Eberhart suggested an algorithm of practical swarm optimization (PSO) that depends totally on the simulation of birds swarms [19]. PSO and ACO were a starting point for a new branch of the swarm intelligence (SI). The most important characteristics of these new branches are their dependence on digital treatment as they are not based on mathematical knowledge. They are considered as complex algorithms composed of specific steps with known start and end points which lead to problem solving. Even with the great enhancement of computing capacities, still there are challenging problems. Fortunately, many sensitive research algorithms are developed to find suitable solutions for those problems at a reasonable time. They are developed according to the evolution of physiology and biology. Examples are genetic algorithm (GA) and simulated annealing (SA), these techniques are used to solve many problems widely [20–22].

This work proposes a new algorithm that is called rooted tree optimization (RTO) as its concept is extracted from the movement of plant roots while looking for the nearest place of water. RTO algorithm mimics the behavior of desert plants where the water resources are lacked. If vegetal/biologists scientists allow, we can say that a desert plant roots smell the places of water (intuitive behavior) around it, where these places present the optimal solution. To determine water places, a group of roots which oriented

* Corresponding author at: University of El-Oued, Department of Electrical Engineering, P.O. Box 789, 39000, El-Oued, Algeria.
E-mail addresses: yacine-labbi@univ-eloued.dz, yacinelabbi@gmail.com (Y. Labbi).

http://dx.doi.org/10.1016/j.ijepes.2016.01.028
0142-0615/

by a special conducting are used. Therefore, this work attempts to introduce an algorithm (RTO) based on that intuitive behavior which leads to water locations and has an oriented movement while looking for the best solution.

RTO starts its search with a random set of solutions (group of roots). Each population member is evaluated based on a given objective function and assigned with its fitness value. Best candidate solutions are forwarded for the next generation/iteration while others are discarded and compensated by new set of random solutions in each generation. Far solutions from water places are omitted and replaced by new roots oriented randomly (they can be replaced by roots closer to the best root of the previous generation). The stopping criterion is the completion of a maximum number of cycles or generations. At the end of the cycles, the solution with the best fitness will be the desired solution.

The main aim of this study is to present the use of the RTO algorithm to the subject of ED in power systems. Thus, the RTO algorithm has been proposed and applied to solve the ED problem for 3, 6, 13 and 15-units test systems. Furthermore, the valve-point effect, the prohibited operating zones, ramp rate limits constraints, and transmission network losses have been considered. The results obtained with the proposed RTO algorithm were analyzed and compared other with optimization results reported in literature. The reminder of this paper is organized as follows: Section 2 presents the mathematical formulation of ED problem considering various constraints. RTO concept and its application is explained in Section 3. The parameter settings for the test system to evaluate the performance of RTO and the results are discussed in Sections 4 and 5. The conclusion is drawn in Section 6.

2. Formulation of ED problem

ED problem is an optimization problem to determine the schedule of real power outputs of generating units subjected to the real power balance with load demand as well as limits on generators' outputs. The mathematical formulation of the problem can be defined as follows:

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

where F_i is the total fuel cost of the generator units, which is defined by:

$$F_i(P_i) = a_i \times P_i^2 + a_i \times P_i + c_i \tag{2}$$

where $a_i,\ b_i$ and c_i are cost coefficients of generator i and P_i is subject to power balance constraints:

$$\sum_{i=1}^{N} P_i - P_D - P_L = 0 \tag{3}$$

where P_D is the system load demand and P_L is the transmission network loss. The transmission loss may be expressed using β-coefficients as

$$P_L = \sum_{i=1}^{N}\sum_{j=1}^{N} P_i \cdot \beta_{ij} \cdot P_j + \sum_{i=1}^{N} \beta_{oi} \cdot P_i + \beta_{oo} MW \tag{4}$$

where β_{ij}, β_{oi}, β_{oo} are the elements of loss coefficient, the output of each generator must be within its limits:

$$P_i^{\min} \leqslant P_i \leqslant P_i^{\max} \quad \forall i = 1, 2, 3, \ldots, N \tag{5}$$

2.1. ED with valve-point loading effects

The inclusion of valve-point loading effect makes the modeling of the incremental fuel cost function of generators more practical. It increases the non-linearity as well as the number of local optima in the solution space. The incremental fuel cost function of the generating units with valve-point loadings are represented [2,7,8] as follows:

$$\tilde{F}_i(P_i) = F_i(P_i) + \left| e_i \times \sin(f_i \times (P_i^{\min} - P_i)) \right| \tag{6}$$

$$\tilde{F}_i(P_i) = a_i \times P_i^2 + a_i \times P_i + c_i + \left| e_i \times \sin(f_i \times (P_i^{\min} - P_i)) \right| \tag{7}$$

where e_i and f_i are constants of the valve-point effect of generators. Therefore, the total fuel cost to be minimized is represented in Eq. (7) [11]:

$$\min\ F = \sum_{i=1}^{N} \tilde{F}_i(P_i) \tag{8}$$

where $\tilde{F}_i$ is the cost function of ith generator in (\$/h).

2.2. ED with prohibited operating zones and ramp rate limits constraints

Here the objective function is to be minimized subject to the following constraints.

- *Real Power Balance Constraint/Generator Capacity Constraints:* The real power balance constraint remains the same as in (3). Also, the output of each generator must be within its limits as given in (5).
- *Ramp Rate Limit Constraints:* The power generated, $P_{i,(t-1)}$, by the ith generator in certain interval may not exceed that of previous interval by $P_{i,(t-1)}$ more than a certain amount UR_i, the up-ramp limit and neither may it be less than that of the previous interval by more than some amount DR_i the down-ramp limit of the generator. These give rise to the following constraints.
  Generating unit ramp-rate limits:

$$\begin{aligned} P_{i,t} - P_{i,(t-1)} &\leqslant UR_i, \quad i = 1, 2, \ldots, N \\ P_{i,(t-1)} - P_{i,t} &\leqslant DR_i, \quad i = 1, 2, \ldots, N \end{aligned} \tag{9}$$

  where UR_i and DR_i are ramp-up and ramp-down rate limits of ith unit, respectively and are expressed in MW/h.
- *Prohibited operating zones constraint:* In practical operations, generated output P_i of unit i must avoid operations in prohibited zones. The feasible operating zones of unit i can be described as

$$\begin{aligned} & P_{\min,i} \leqslant P_i \leqslant P_{i,l}^{L} \text{ or} \\ & P_{i,k-1}^{U} \leqslant P_i \leqslant P_{i,k}^{L}\ k = 2, \ldots, n_i \text{ or} \\ & P_{i,n_i}^{U} \leqslant P_i \leqslant P_{\max,i} \end{aligned} \tag{10}$$

  where n_i is the number of prohibited zones for unit i, k index of prohibited zones of a unit and $P_{i,k-1}^{U/L}$ are the lower/upper bounds of the kth prohibited zones of unit i.

3. Rooted tree optimization algorithm (RTO)

3.1. Roots look for water

One root has limited capacity, but a group of roots can find together the best place to get water, and the majority of them are located around this place or around the way that links the plant with the resource of water. To create the algorithm, a hypothetical behavior has been added which is the way how roots decide together to choose their orientation according to the wetness degree where the root head is located. These roots move randomly but when one or more find(s) the wetness; they call other roots to intensify their existence around this way/location to become a new starting point for the majority of root groups to get the original place of water (i.e., optimal solution). Fig. 1 presents how roots of a plant behave when they looking for water (i.e., a solution). According to RTO algorithm, far solutions from water place (which has a less wetness degrees) are omitted or replaced by new roots oriented randomly. Furthermore, far solutions from water place can be replaced by roots near the best root of the previous generation. Roots which have a considerable wetness degree preserve their orientation.

3.2. Rooted tree optimization algorithm

Similar to other methods, the proposed algorithm begins by creating an initial population randomly. This section introduces some terms to determine how the RTO algorithm is moving from an initial population to a new one.

- *Root:* is a candidate or the suggested solution.
- *Wetness degree* (D_w): it is a term that evaluates a candidate and gives its fitness degree between the rests of population.

3.2.1. The rate of the nearest root to water (R_n)

It is the rate that represents the number of candidates according to the total population that should gather around the wetter place (the best solution). It will be the successor of the roots which were in dry places (wetness is so weak) from the previous generation. The new population of the nearest root to water is calculated according to:

$$x^{new}(i, It+1) = x^{best}(It) + c_1 \times D_w(i) \times \text{randn} \times l/(N \times It) \tag{11}$$

where It is the iteration step, $x^{new}(It+1)$ is the new candidate for the iteration $(It+1)$, $x^{best}(It)$ is the best solution from the previous generation, i is the candidate number, N is the population scale, l is the upper limit of the parameter and randn is a normal random number between $[-1,1]$. Then a new point x^{new} is upper and lower bounded.

3.2.2. The rate of the continuous root in its orientation (R_c)

It is the rate of members that continued/forwarded from the previous way because they appear near water. The new population of *the random root* is calculated according to:

$$x^{new}(i, It+1) = x(i, It) + c_2 \times D_w(i) \times \text{rand} \times (x^{best}(It) - x(i, It)) \tag{12}$$

where $x(It)$ is the previous candidate for the iteration It and rand is random number between $[0,1]$.

3.2.3. The rate of the random root (R_r)

It is the rate that represents the number of candidates according to the total population that they will spread randomly in the research field in order to increase the rate of getting the global solution. They also replace the roots with a weak wetness degree from the last generation. The new population of *the random root* is calculated according to:

$$x^{new}(i, It+1) = x_r(It) + c_3 \times D_w(i) \times \text{randn} \times l/It \tag{13}$$

where x_r is individual randomly selected from the previous generation, c_1, c_2 and c_3 are adjustable parameters.

The rates R_n, R_r and R_c are determined experimentally according to the problem under study. These rates are considered as variables which affect the convergence. The rate R_r is always small compared to others because it aims to reserve the randomness to be far from local solutions, e.g. its role is similar to mutation in genetic algorithm. D_w value has been inserted in research functions of roots to determine a research space according to the candidate power.

3.3. The RTO algorithm

The steps of RTO algorithm can be summarized as follows:

Step 1: create primary generation randomly which is composed of N candidates within the variables' limits in the research space, and the determine the numerical values of R_n, R_r and R_c rates.

Step 2: Evaluate all population members to measure their wetness degree (D_w) as following:

$$D_w(i) = \begin{cases} \frac{f_i}{\max(f_i)} & \text{for the maximum objective} \\ 1 - \frac{f_i}{\max(f_i)} & \text{for the minimum objective} \end{cases}, \quad i = 1, 2, \ldots, N \tag{14}$$

Or directly use the fitness regardless of the suitable formula.

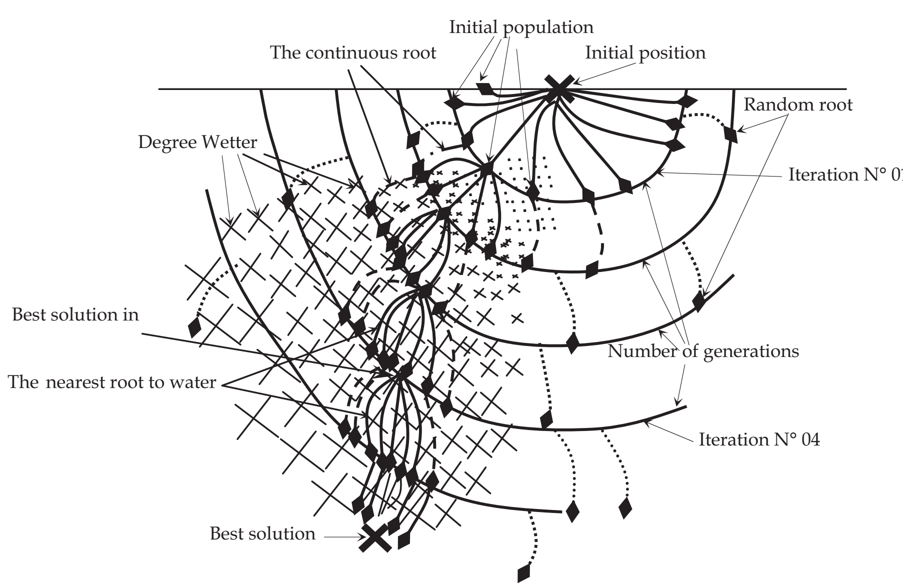

Fig. 1. Roots of plant behavior when they look for water (solution).

Step 3: Reproduce and replace by the new population. Reorder the population according to the wetness degree (D_w) in order to replace them by the new population according to R_n, R_r and R_c using Eqs. (11)–(13) as following:

$D_w = 0$	$x_{new}(i, it+1) = x_r + c_3 * D_W(i) * randn * l / it$	For $i = 1, \ldots, N*R_r$
↑	$x_{new}(i, it+1) = x^{best}(it) + c_1 * D_W(i) * randn * l / (N*it)$	For $i = N*R_r + 1, \ldots, N*(R_n + R_r)$
$D_w = 1$	$x_{new}(i, it+1) = x(i, it) + c_2 * D_W(i) * rand * (x^{best}(it) - x(i, it))$	For $i = N*(R_n + R_r) + 1, \ldots, N$

where $R_n + R_r + R_c = 1$. We start by the candidate with the lowest (D_w) till we get at the one with wetness degree equivalent

Step 4: Return to *step* 2 if the stopping criteria is not realized.

Fig. 2 shows the pseudocode of the proposed RTO algorithm, which summarizes steps 1–4 that previously presented. It comprises two main procedures: (a) Initialization, and (b) Loop with the repeated body while the stopping criterion is not satisfied.

Rastrigin function is a non-convex function used as a performance test problem for optimization algorithms. It is a typical example of non-linear multimodal function. It was proposed by Rastrigin as a two-dimensional function. Finding the minimum of this function is a fairly challenging task due to its large search space and its large number of local minima. This function has been used as an example to test the proposed RTO algorithm. Fig. 3 represents the change of the two-dimensional parameters (x_1 and x_2) with the generation number. Three kinds of roots appear: random roots, roots which meet together till they reach the solution, and the rest represent those roots which cease when they become weak in comparison with other roots of the same generation. Fig. 4 shows that the majority of roots gather in the solution -

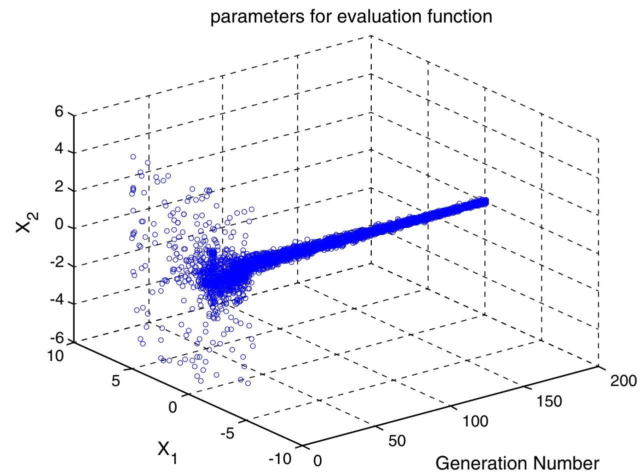

Fig. 3. The parameters and the roots for 200 iterations (the Rastrigin function ($N = 2$)).

```
Algorithm RTO
Begin
    // Initialization:
    Set the rates Rn, Rr and Rc parameters
    Give the maximum number of iterations : MaxIte, and the population scale : theRTOsize
    Set iteration counter it =1
    Generate the initial population X(1) randomly within the search range of (Xmin, Xmax)
    // Loop
    Repeat
        Evaluate the DWi for each root   // DW : Fitness,  root : Individual
        Reorder the population according to the witness degree
        Identify the candidate according the wetness place Xbest   // the global best in whole population
        For i=1 to Rr×theRTOsize do
            Selected individual Xr(it) randomly from the current population
            Xi(it+1)= Xr(it)+c1×DWi×randn×|Xmax+Xmin|/it
        End for
        For i= Rr×theRTOsize +1 to (Rr+Rn)×theRTOsize do
            Xi(it+1)= Xbest+c3×DWi×randn×|Xmax+Xmin|/(it×theRTOsize)
        End for

        For i=(1-Rc) ×theRTOsize +1 to theRTOsize do
            Xi(it+1)= Xi(it)+c1×DWi×rand×(Xbest- Xi(it))
        End for

        Update Xbest
        it = it + 1
    until for a stop criterion is not satisfied  // & it < MaxIte
End
```

Fig. 2. Pseudocode for the proposed RTO algorithm.

resource of water (optimal solution). Fig. 5 shows the concentration of the roots (candidate solutions) with different rates R_n, R_c and R_r where we can see how every kind looks for water through changing rates' values. Therefore, convergence to a solution strongly depends on the behavioral movements of roots.

4. Applying RTO to the ED problem

In this section the proposed RTO algorithm has been applied to solve the economic dispatch problem. To apply the RTO, the following steps have to be taken.

Step 1. Define the input data.
In this step, the input data including the cost coefficients of generators, output power constraints, transmission loss matrix coefficients and loads, maximum number of iterations (Iter_{max}), size of the population (candidates), the adjustable parameters, and the difference rates R_n, R_r and R_c.

Step 2. Generate the initial population.
Initialize randomly the individuals of the population according to the limit of each unit including individual dimensions. These initial individuals must be feasible candidate solutions that satisfy the practical operation constraints.

Step 3. For each individual P_{Gi} of the population, employ the β-coefficient loss formula to calculate the transmission loss P_L.

Step 4. Calculate the evaluation value (fitness) of each individual P_{Gi} in the population using the evaluation function given by (11), (*Evaluate fitness D_{Wi} for each candidate*).

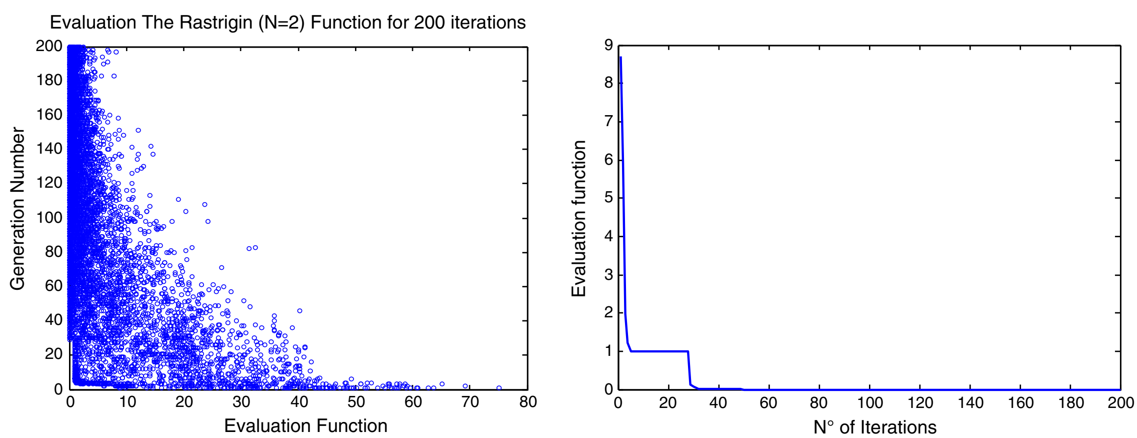

Fig. 4. Evaluation the Rastrigin function ($N = 2$) for 200 iterations.

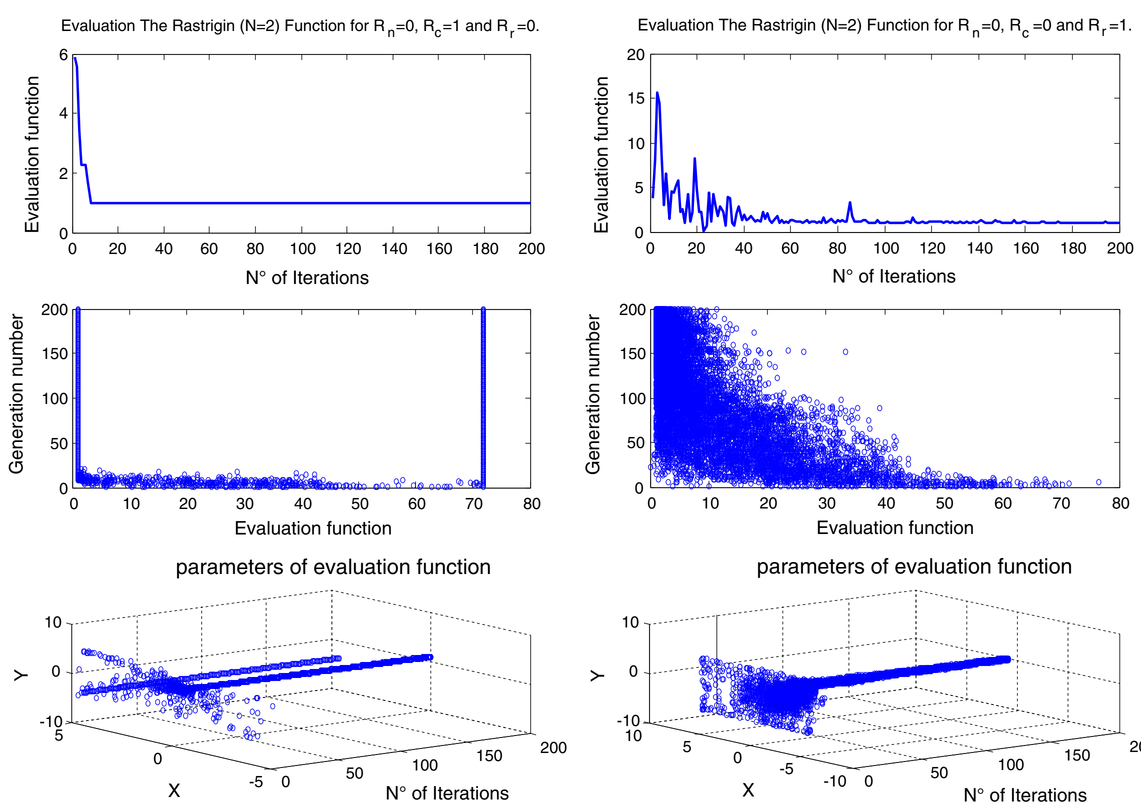

Fig. 5. Evaluation the Rastrigin function ($N = 2$) for different rates parameters; R_n, R_r and R_c.

Step 5. Compare each individual's evaluation value with Pg^{lbest} the best fitness of the particle up.

Step 6. Calculate new candidates using (11)–(13).

$$\begin{aligned} P_{Gi,d}(k+1) &= P_{Gr,d}(k) + c_3 + D_{W_i} \times randn \\ &\quad \times |P_{Gd,Max} - P_{Gd,Min}|/it, \quad \text{for } i = 1 \text{ to } R_r \times n \\ P_{Gi,d}(k+1) &= P^{best}_{Gd}(k) + c_1 + D_{W_i} \times randn \\ &\quad \times |P_{Gd,Max} - P_{Gd,Min}|/(n \times it), \\ &\quad \text{for } i = R_r \times n + 1 \text{ to } (R_r + R_n) \times n \\ P_{Gi,d}(k+1) &= P_{Gi,d}(k) + c_2 + D_{W_i} \times rand \\ &\quad \times \left|P^{best}_{Gd}(k) - P_{Gi,d}(k)\right|, \quad \text{for } i = (1 - R_c) \times n \text{ to } n \end{aligned} \tag{15}$$

where $d = 1, 2, \ldots, m$ and $P_{Gr,d}$ is individual selected randomly from the current population; n is the population size, m is the number of units, $P_{Gd,Max}$ and $P_{Gd,Min}$ are parameter upper and lower limits and k number of iterations.

Step 7. If the number of iterations reaches the maximum, then go to Step 8. Otherwise, go to Step 3.

Step 8. The individual that generates the latest P^{lbest} is the optimal generation power of each unit with the minimum total generation cost.

Fig. 6 depicts the schematic representation of the proposed algorithm to solve the ED problem.

5. Experimental results

In order to verify the feasibility and efficiency of the proposed algorithm, RTO algorithm was applied on two set of case studies. The first set includes a 23 standard benchmark functions. The second set includes four test systems (i.e., 3, 6, 13 and 15-units systems) for solving ED problem considering various constraints. In these examples, the RTO algorithm was implemented in MATLAB using Pentium IV, 3-GHz personal computer with 2 GB RAM under Windows XP.

5.1. Validation (benchmark tests)

5.1.1. Robustness testing

Before solving economic dispatch problems, RTO was benchmarked using four numerical examples which are given as follows in detail. The new algorithm RTO has been tested and compared

Table 2
Success rates of different algorithms.

RTO algorithm	*Tolerance*				
Functions	1e−5	1e−5	1e−6	1e−7	1e−8
Rosenbrock ($N = 10$)	100	100	98	98	96
Rosenbrock ($N = 5$)	100	100	100	100	100
Rosenbrock ($N = 3$)	100	100	100	100	100
Dejong ($N = 10$)	100	100	100	100	100
Dejong ($N = 3$)	100	100	100	100	100
Griewangk ($N = 10$)	100	100	100	100	100
Griewangk ($N = 5$)	100	100	100	100	100
Rastrigin ($N = 2$)	77	58	51	45	40

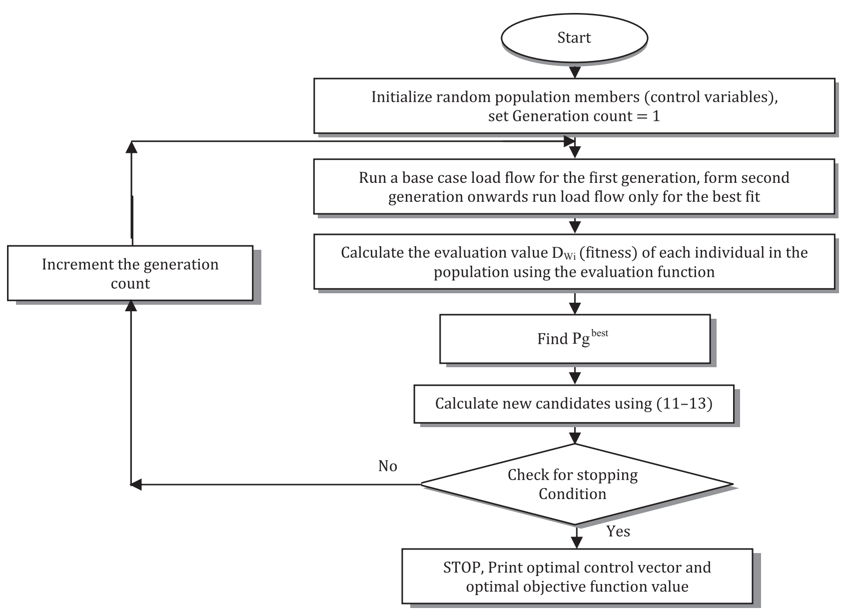

Fig. 6. Flow chart of RTO.

Table 1
Properties of test problems.

Function name	Definition	lower bound	upper bound	optimum point	Property
Rosenbrock	$\sum_{i=1}^{N-1} 100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2$	−2.048	2.048	0	Unimodal
Dejong	$\sum_{i=1}^{N} x_i^2$	−5.12	5.12	0	Unimodal
Griewangk	$\sum_{i=1}^{N}(x_i^2/4000) - \prod_{i=1}^{N}(x_i/\sqrt{i}) + 1$	−50	50	0	Multimodal
Rastrigin	$10 \times N + \sum_{i=1}^{N}(x_i^2 - 10.\cos(2\pi.x_i))$	−5.12	5.12	0	Multimodal

using the benchmark problems taken from [23]. The difficulty levels of most benchmark functions are adjustable by setting their parameters. From the standard set of benchmark problems available in the literature, four important functions two of which are unimodal (containing only one optimum) and two of which are multimodal (containing many local optima, but only one global optimum) are considered to test the efficacy of proposed methods [24]. This list comprises some widely used test functions such as sphere, Rosenbrock, Dejong, Griewangk, and Rastrigin functions given in Table 1 which shows the main properties of the selected benchmark functions used in the experiment.

Two criteria are applied to terminate the simulation of the algorithms: reaching maximum number of iterations which is set to a constant number and/or getting a minimum error.

Table 3
Success rates of RTO algorithms using different rates parameters; R_n, R_r and R_c.

	Rosenbrock ($N = 10$)		Dejong ($N = 10$)		Griewangk ($N = 10$)		Rastrigin ($N = 2$)	
Tolerance	1e−5	1e−7	1e−5	1e−7	1e−5	1e−7	1e−5	1e−7
$R_n = 1.0$, $R_r = 0.0$ and $R_c = 0.0$	0	0	0	0	0	0	2	0
$R_n = 0.7$, $R_r = 0.3$ and $R_c = 0.0$	99	24	100	100	24	0	67	62
$R_n = 0.6$, $R_r = 0.4$ and $R_c = 0.0$	99	18	100	100	20	0	77	70
$R_n = 0.3$, $R_r = 0.7$ and $R_c = 0.0$	98	11	100	80	30	0	98	98
$R_n = 0.0$, $R_r = 1.0$ and $R_c = 0.0$	16	0	0	0	0	0	0	0
$R_n = 0.9$, $R_r = 0.0$ and $R_c = 0.1$	0	0	0	0	0	0	6	4
$R_n = 0.6$, $R_r = 0.3$ and $R_c = 0.1$	100	33	100	100	94	88	66	59
$R_n = 0.3$, $R_r = 0.6$ and $R_c = 0.1$	100	37	100	100	94	88	95	92
$R_n = 0.1$, $R_r = 0.8$ and $R_c = 0.1$	100	77	100	100	96	90	98	96
$R_n = 0.0$, $R_r = 0.9$ and $R_c = 0.1$	11	0	100	98	96	93	88	35
$R_n = 0.7$, $R_r = 0.0$ and $R_c = 0.3$	19	9	2	0	100	99	25	19
$R_n = 0.4$, $R_r = 0.3$ and $R_c = 0.3$	100	98	100	100	100	100	77	72
$R_n = 0.3$, $R_r = 0.4$ and $R_c = 0.3$	100	97	100	100	100	100	74	70
$R_n = 0.0$, $R_r = 0.7$ and $R_c = 0.3$	56	27	100	100	100	100	100	70
$R_n = 0.4$, $R_r = 0.0$ and $R_c = 0.6$	93	88	0	0	100	100	32	25
$R_n = 0.1$, $R_r = 0.0$ and $R_c = 0.9$	98	92	80	74	100	100	55	52
$R_n = 0.0$, $R_r = 0.1$ and $R_c = 0.9$	100	99	100	99	100	100	85	84
$R_n = 0.0$, $R_r = 0.0$ and $R_c = 1.0$	100	97	100	98	100	99	62	59

Table 4
Comparison of success rates between different algorithms.

	BB–BC [24]		BB–CBC [24]		UBB–BC [24]		UBB–CBC [24]		RTO	
Tolerance	1e−5	1e−6	1e−5	1e−6	1e−5	1e−5	1e−6	1e−6	1e−5	1e−6
Rosenbrock ($N = 100$)	100	69	100	100	100	92	82	100	100	100
Dejong ($N = 3$)	100	31	100	70	100	100	100	61	100	78
Griewangk ($N = 2$)	23	19	31	30	36	100	100	31	39	38
Rastrigin ($N = 2$)	30	26	80	75	84	29	22	79	90	86

Table 5
Minimization result of the unimodal benchmark functions with dimension = 30. Maximum number of iterations = 1000.

Num	Test function	Results	SA	GA	PSO	RTO
F1	Sphere	Min fitness	2.1395	1.5×10^{-5}	4.7×10^{-6}	1.2×10^{-4}
		Mean fitness	5.9119	1.59×10^{-5}	7.0×10^{-4}	3.0×10^{-4}
		STD	1.6932	1.6×10^{-4}	7.8×10^{-4}	5.2×10^{-5}
F2	Schwefel 2.22	Min fitness	13.2259	0.0842	0.0407	0.2945
		Mean fitness	14.5929	0.5682	1.5622	2.5692
		STD	0.7271	0.6910	0.0842	2.3132
F3	Schwefel 1.2	Min fitness	1.2954	0.5889	50.725	6.4161
		Mean fitness	1.7439	1.4401	210.834	16.396
		STD	0.3643	1.1497	132.01	5.0881
F4	Schwefel 2.21	Min fitness	8.9323	1.4571	3.2359	1.0519
		Mean fitness	9.5756	2.0405	7.4755	4.8355
		STD	0.3939	0.3679	2.8936	2.9240
F5	Rosenbrock	Min fitness	2.1748	0.0716	23.7363	1.4×10^{-6}
		Mean fitness	10.652	16.717	60.6540	6.2514
		STD	6.5973	30.373	30.4834	15.021
F6	Schwefel	Min fitness	3.4559	3.26×10^{-5}	3.97×10^{-5}	2.7×10^{-4}
		Mean fitness	5.3777	5.01×10^{-4}	0.0053	3.4×10^{-4}
		STD	1.5127	7.59×10^{-4}	0.0070	4.6×10^{-5}
F7	Quartic	Min fitness	4.9915	0.5461	0.0326	0.0532
		Mean fitness	8.3315	0.7304	0.0840	0.4654
		STD	1.8783	0.1590	0.0462	0.3298

A 100 candidates were initialized in regions that include the global optimum for a fair evaluation. The algorithm was run for 100 times to catch their stochastic properties. In this experiment, maximum iteration number was set to 500 and the goal is not to find the global optimum values but to find out the potential of the algorithm. Algorithm success rate defined by; how often does the algorithm get the exactitude before it completes the number of the whole iterations or all 100 trials.

As shown in Table 2, the positive aspect of the RTO algorithm is its ability to find the global solution and to avoid falling into local ones. This is because number of each roots type (i.e., random type) will not change while they looking for a solution as the reached

Table 6
Minimization result of the Multimodal benchmark functions with dimension = 30. Maximum number of iterations = 1000.

Num	Test function	Results	SA	GA	PSO	RTO
F8	Schwefel 2.26	Min fitness	$4.6 \times 10^{+3}$	$2.5 \times 10^{+3}$	$3.9 \times 10^{+3}$	$3.7 \times 10^{+3}$
		Mean fitness	$5.5 \times 10^{+3}$	$3.0 \times 10^{+3}$	$4.8 \times 10^{+3}$	$4.4 \times 10^{+3}$
		STD	$7.1 \times 10^{+2}$	$4.9 \times 10^{+2}$	$6.2 \times 10^{+2}$	$4.7 \times 10^{+2}$
F9	Rastrigin	Min fitness	80.69	3.985	30.84	$1.2 \times 10 + 2$
		Mean fitness	$1.2 \times 10^{+2}$	7.963	50.63	$1.7 \times 10^{+2}$
		STD	30.75	4.0324	17.12	31.92
F10	Ackley	Min fitness	2.5394	0.0029	0.0112	2.7386
		Mean fitness	2.8432	0.4782	3.2050	7.2193
		STD	0.1935	0.6259	1.7018	4.6516
F11	Griewank	Min fitness	0.0510	2.7×10^{-7}	5.9×10^{-5}	5.2×10^{-4}
		Mean fitness	0.1560	9.9×10^{-4}	0.1020	0.0116
		STD	0.0561	0.0031	0.1458	0.0135
F12	Levy	Min fitness	6.2×10^{-11}	4.3×10^{-13}	4.7×10^{-31}	3.9×10^{-8}
		Mean fitness	1.9×10^{-9}	5.7149	1.5550	3.1579
		STD	2.5×10^{-9}	7.0450	3.9475	2.4709
F13		Min fitness	8.5193	2.4×10^{-6}	0.0110	1.3×10^{-32}
		Mean fitness	40.0168	0.0099	4.0782	1.3×10^{-32}
		STD	26.08	0.01669	5.01819	2.8×10^{-48}

Table 7
Minimization result of the Multimodal benchmark functions with maximum number of iterations = 500.

Num	Test function	Dim	Results	SA	GA	PSO	RTO
F14	Shekel's foxholes	2	Min fitness	0.9980	0.9980	0.9980	0.9980
			Mean fitness	3.6345	4.8133	0.9980	1.8906
			STD	4.9300	4.6191	0	1.0905
F15	Nowalick's	4	Min fitness	8.4×10^{-4}	7.7×10^{-4}	3.0×10^{-4}	3.0×10^{-3}
			Mean fitness	0.0020	0.0050	0.0044	9.1×10^{-3}
			STD	0.0011	0.0080	0.0084	4.0×10^{-3}
F16	Six hump Camel back	2	Min fitness	−1.0316	−1.0316	−1.0316	−1.0316
			Mean fitness	−1.0316	−1.0316	−1.0316	−1.0316
			STD	3.5×10^{-5}	1.0×10^{-11}	1.2×10^{-16}	1.0×10^{-6}
F17	Branin-Hoo	2	Min fitness	0.39788	0.39788	0.39788	0.39788
			Mean fitness	0.39826	0.39788	0.39788	0.39788
			STD	0.0011	1.1×10^{-11}	0	0
F18	Goldstein&Price	2	Min fitness	3.0000	3.0000	2.9999	2.9999
			Mean fitness	3.0000	5.7000	2.9999	2.9999
			STD	1.8×10^{-7}	8.5381	4.9×10^{-16}	2.0×10^{-16}
F19	Hartman 1	3	Min fitness	−0.16627	−3.86278	−2.7145	−3.86278
			Mean fitness	−0.01662	−3.86278	−2.8249	−3.86278
			STD	0.05258	4.9×10^{-10}	0.04836	9.3×10^{-16}
F20	Hartman 2	6	Min fitness	−0.2618	−3.2229	−3.0619	−3.2002
			Mean fitness	−0.0263	−3.2070	−1.6880	−3.2002
			STD	0.0827	0.0109	0.7329	1.3×10^{-5}
F21	sHEKEL'S 1	4	Min fitness	−2.0748	−5.0551	−4.9453	−2.63047
			Mean fitness	−0.8178	−5.0551	−4.1458	−2.63047
			STD	0.6887	3.26×10^{-10}	0.4624	3.6×10^{-16}
F22	sHEKEL'S 1	4	Min fitness	−5.0301	−5.0876	−6.4906	−10.4029
			Mean fitness	−0.8662	−5.0876	−4.4945	−4.2933
			STD	1.4704	1.7×10^{-10}	1.5114	3.2200
F23	sHEKEL'S 1	4	Min fitness	−5.0929	−5.1284	−5.0648	−2.8711
			Mean fitness	−0.9743	−5.1284	−3.2224	−2.8130
			STD	1.484085	1.4×10^{-10}	0.9442	0.0204

solution may be a local ones, roots keep their random nature which is similar to the mutation concept in GA and, thus they avoid falling in local solutions.

Table 3 presents the effect of different rate values R_n, R_r and R_c on the convergence to the solution for different functions. This table clarify the desired rates (experimentally) that should be taken to get exactly the solution according to the kind of problem. It is worthy to mention that the values of rates (R_n, R_r and R_c) that have been used in Table 2 are selected based on several trials as shown in Table 3 in order to provide the best performance (i.e., high speed of convergence an few number of iteration).

5.1.2. Comparison with different optimization techniques

First, in order to make a fair comparison between our proposed RTO algorithm with other heuristic methods [24], 500 iterations are chosen as stopping criteria in the simulations and the population size is kept fixed as 40 in the example and the benchmark tests. Table 4 presents the success rates obtained from RTO, BB–BC (Big Bang–Big Crunch), BB–CBC (Big Bang–Chaotic Big Crunch optimization), UBB–BC (Uniform Big Bang–Big Crunch), and UBB–CBC (Uniform Big Bang–Chaotic Big Crunch algorithms) at different quality levels for the benchmark functions.

Second, the proposed RTO approach has been applied on 23 standard benchmark functions that provided in [43]. The obtained results have been compared with those of simulated annealing (SA), GA and PSO. In all cases, population size was set to be 60. Dimension was set to be 30 and the maximum number of iterations was set to be 1000 for functions of Tables 5 and 6, and 500 for functions of Table 7.

Table 8
Generator data of test system 1.

Units	P^i_{min}	P^i_{max}	a	b	c	e	f
1	100	600	0.001562	7.92	561	300	0.0315
2	50	200	0.004820	7.97	78	150	0.063
3	100	400	0.001940	7.85	310	200	0.042

The RTO parameters are the various values of adjustable parameters for all the following cases are: $R_n = 0.4$, $R_r = 0.3$ and $R_c = 0.3$. In PSO, the acceleration constants c_1 and c_2 which both set as 2.0, and the inertia factor (w) which is decreasing linearly from 0.9 to 0.2. In GA, the crossover and mutation probabilities were set to be 0.7 and 0.1, respectively. In SA, the initial temperature T_0 set as 50,000, temperature reduction factor r lies in the range from 0.80 to 0.99 and the scaling factor σ for GPDF set to be 0.02.

We applied the four mentioned algorithms to the benchmark functions, and the results for the following cases are:

Table 9
Results obtained by proposed method for test system 1.

Units (MW)	Proposed RTO
1	300.2536
2	149.7485
3	399.9972
Total power output (MW)	850.000
Total cost ($/h)	**8234.07157**
Time (s)	0.3008

Bold value indicates the best solution found by proposed algorithm.

Table 10
Comparison of proposed method for test system 1.

Method	P_1 (MW)	P_2 (MW)	P_3 (MW)	P_D (MW)	Cost ($/h)
GA [26]	398.700	50.100	399.600	848.400	8222.07
EP [26]	300.264	149.736	400.000	850.000	8234.07
EP-SQP [26]	300.267	149.733	400.000	850.000	8234.07
PSO [26]	300.268	149.732	400.000	850.000	8234.07
PSO-SQP [26]	300.267	149.733	400.000	850.000	8234.07
MPSO [25]	300.27	149.74	400.00	850.000	8234.07
PS [9]	300.2663	149.7331	399.9996	849.9990	8234.05
GSA [27]	300.2102	149.7953	399.9958	850.0013	8234.1
Proposed RTO	300.2669	149.7331	400.0000	850.0000	**8234.0717**

Bold value indicates the best solution found by proposed algorithm.

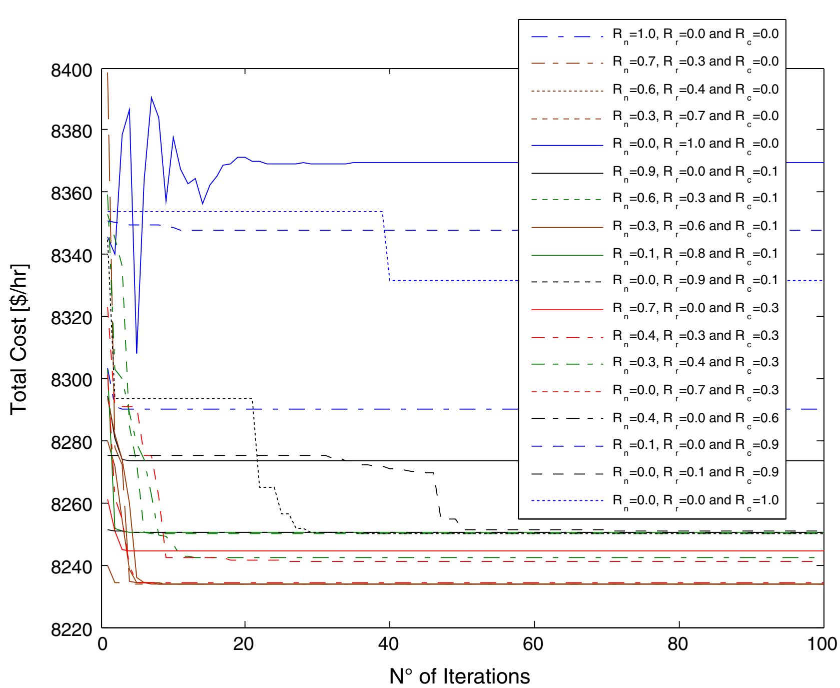

Fig. 7. Convergence characteristic of the three-generator systems for different adjustable parameters to RTO algorithm.

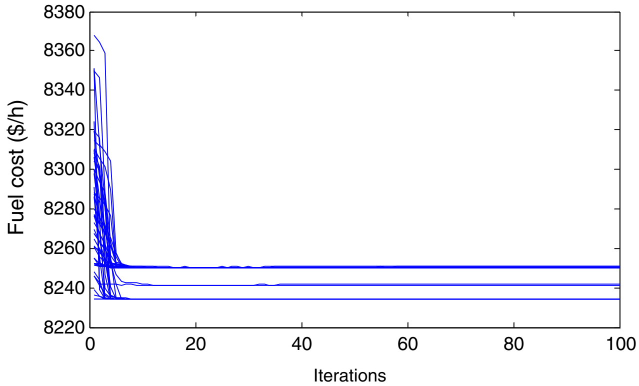

Fig. 8. Convergence of fitness value for load demand 850 MW.

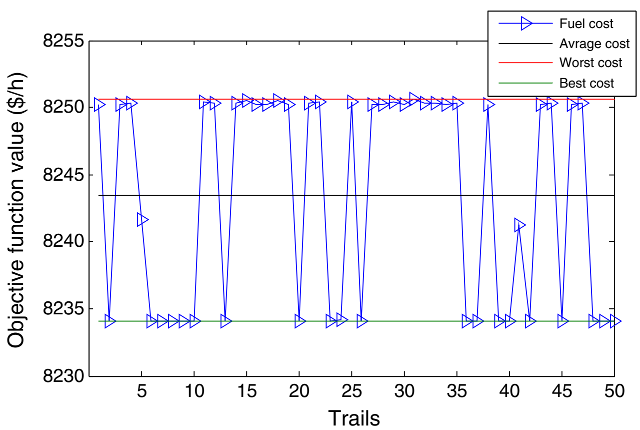

Fig. 9. Distribution of objective function value for 50 trails.

5.1.2.1. Unimodal high-dimensional functions. Functions F1–F7 are unimodal functions. In this case the convergence rate of the search algorithm is more important for unimodal functions than the final results because there are other methods which are specifically designed to optimize unimodal functions.

The results are averaged over 10 runs and the best, average and the standard deviation of the fitness function are reported for unimodal functions in Table 5. As this table illustrates, RTO provides good quality solutions.

5.1.2.2. Multimodal high-dimensional functions. Multimodal functions have many local minima and are difficult to be optimized. For multimodal functions, the final results are more important since they reflect the ability of the algorithm in escaping from poor local optima and locating a near-global optimum. Several experiments have been carried out on F8–F13 where the number of local minima increases exponentially as the dimension of the function increases. The dimension of these functions is set to be 30. The results are averaged over 10 runs and the best, average and the standard deviation of the fitness function are reported in Table 6.

5.1.2.3. Multimodal low-dimensional functions. Table 7 shows the comparison between RTO, PSO, GA and SA on multimodal low-dimensional benchmark functions that provided in [43]. The results show that RTO, PSO and GA have similar quality of solutions.

5.2. Economic dispatch problem valve-point loading effects

5.2.1. Case 1: 3-unit system with valve-point loading effects

This test case study considers three thermal units of generation with effect of valve-point as given in Table 8. In this case, the load demand was $P_D = 850$ MW.

The simulation parameters for the proposed algorithm are:

- The number of generation is 50 iterations and size of population is 100 individuals (candidates).
- Rate values are $R_n = 0.4$, $R_r = 0.3$ and $R_c = 0.3$.

R_n, R_r and R_c are adjustable parameters controlling the influence of the convergence properties of the proposed algorithm.

Fig. 7 shows the effect of various values for R_n, R_r and R_c on convergence characteristic of the proposed method for the

Table 11
Results obtained by proposed method for test system 2.

	Solution methods					
	GA [28]	GA-APO [28]	NSOA [28]	PSO [29]	MSG-HP [29]	RTO
$P_{G,1}$	150.724	133.9816	182.478	197.8648	199.6331	199.5996
$P_{G,2}$	60.8707	37.2158	48.3525	50.3374	20.0000	20.0008
$P_{G,5}$	30.8965	37.7677	19.8553	15.0000	23.7624	24.1658
$P_{G,8}$	14.2138	28.3492	17.1370	10.0000	18.3934	17.7409
$P_{G,11}$	19.4888	18.7929	13.6677	10.0000	17.1018	19.0252
$P_{G,13}$	15.9154	38.0525	12.3487	12.0000	15.6922	13.7428
$P_{G,Total}$(MW)	292.1096	294.1600	293.8395	295.2022	294.5829	294.2754
F_{total} (R/h)	996.0369	1101.491	984.9365	925.7581	925.6406	**924.9724**
P_{loss}	8.7060	10.7563	10.4395	11.8022	11.1830	10.8754
Time (s)	0.5780	0.156	0.0150	0.3529	0.6215	0.3771

Bold value indicates the best solution found by proposed algorithm.

Table 12
Comparison of results (test system 2) for 50 trial tests.

		Solution methods					
		RTO	MSG-HP [29]	PSO [29]	NSOA [28]	GA-APO [28]	GA [28]
Min	F_{total} (R/h)	**924.9724**	925.641	925.758	984.94	996.04	996.04
	Time (s)	0.3771	0.62151	0.35290	0.0150	0.156	0.141
Max	F_{total} (R/h)	943.8712	928.599	928.427	992.48	1101.49	1117.13
	Time (s)	0.3827	0.77132	0.35591	0.0310	0.578	0.5780
Mean	F_{total} (R/h)	930.17814	926.851	926.388	NA	NA	NA
	Time (s)	0.3785	0.72484	0.35749	NA	NA	NA

NA denotes that the value was not available in the literature.
Bold value indicates the best solution found by proposed algorithm.

Table 13
Comparison of simulation results for test system 3 (case I, load = 1800 MW).

Unit	HMAPSO [30]	MDE [31]	SHDE [32]	PSM [35]	HGA [36]	QPSO [33]	PSO [30]	PSO-TVAC [34]	Proposed
1	538.5611	628.318	628.3172	538.5587	628.3185	538.56	538.561	628.319	628.3072
2	224.4831	149.594	149.5986	224.6416	222.7491	224.7	299.355	149.597	224.3420
3	150.0622	222.758	222.7987	149.8468	149.5996	150.09	75.037	222.749	297.7060
4	109.8862	109.865	109.8673	109.8666	109.8665	109.87	159.734	109.867	60.0000
5	109.9902	109.864	109.8418	109.8666	109.8665	109.87	60.078	109.867	109.8529
6	109.8666	109.866	60	109.8666	109.8665	109.87	109.864	109.867	60.0000
7	109.9903	109.865	109.8641	109.8666	109.8665	109.87	109.913	109.867	60.0000
8	109.8688	60	109.8547	109.8666	60	159.753	109.87	109.867	109.7956
9	109.8668	109.866	109.8576	109.8666	109.8665	109.87	60.069	60	60.0002
10	40	40	40	77.4666	40	77.41	40.035	40	40.0000
11	77.4247	40	40	40.2166	40	40	77.561	40	40.0000
12	55	55	55	55.0347	55	55.01	55.042	55	55.0000
13	55	55	55	55.0347	55	55.01	55	55	55.0000
$P_{G,Total}$	1800	1799.996	1800	1799.9993	1799.9997	1800.002	1800	1800	1800.0044
Min cost	17969.31	17960.39	17963.89	17969.17	17963.83	17969.01	18014.16	17963.879	**17969.8024**
Mean cost	17969.31	17967.19	18046.38	18088.84	17988.04	18075.11	18104.65	18154.562	18056.9358
Max cost	17969.31	17969.09	NA	18233.52	NA	NA	18249.89	18358.31	18204.6303

NA denotes that the value was not available in the literature.
Bold value indicates the best solution found by proposed algorithm.

three-generating unit system. This figure shows that $R_n = 0.4$, $R_r = 0.3$ and $R_c = 0.3$, are suitable values for the RTO algorithm. These parameter values are used for all other examples presented. For this problem, the appropriate choice of the adjustable parameters can be codified in a way resulting from experimental and observational limits.

The results obtained for this case study are listed in Table 9. The proposed algorithm has obtained the optimal solution values by completing 100 iterations in 0.3008 s. Thus, RTO algorithm has approximately good solution for the power demand of 850 MW.

Table 14
Comparison of proposed method for test system 3 (case I, 1800 MW).

Method	Total cost ($/h)	Method	Total cost ($/h)
PSO [18]	18030.72	IGA_MU [37]	17963.98
EP-SQP [26]	17991.03	ST-HDE [32]	17963.89
HDE [32]	17975.73	HGA [7]	17963.83
CGA-MU [37]	17975.34	HQPSO(5) [8]	17963.9571
PSO-SQP [26]	17969.93	DE [38]	17963.83
PS [9]	17969.17	GSA [27]	17960.3684
UHGA [2]	17964.81	Proposed RTO	**17969.8024**
QPSO [14]	17,964		

Bold value indicates the best solution found by proposed algorithm.

The best fuel cost results obtained from the proposed RTO algorithm and other algorithms are compared in Table 10. From Table 10, it is clear that the PS approach did not meet the load demand.

A convergence characteristics of the RTO algorithm are shown in Figs. 8 and 9 which show the distribution of the generation cost of the best solution for each run in the three units test system.

5.2.2. Case 2: The IEEE 30 -bus test case: 6 -unit system with valve-point loading effects and transmission network losses

The second test system is a 6-unit system (IEEE 30 bus system) with effects of valve-point. The required load demand to be met by all generating units is 283.4 MW. The data for this system is provided in [8]. In this test system, the transmission losses are considered and the loss coefficients β matrices are shown in the Appendix. The setup for the proposed algorithm is executed with following parameters:

– The number of generation is 50 iterations and Size of population 100 individuals (candidates).

Table 11 shows the obtained results for this system. Minimum cost, mean cost and maximum cost over 50 trial runs are compared with the results of combination of Modified SubGradient (MSG)

Table 15
Comparison of simulation results for test system 3 (case II, load = 2520 MW).

Unit	HGA [36]	DE [21]	FAPSO-VDE [39]	ICA-PSO [40]	IPSO [41]	RTO
1	628.3184	628.3185	628.3185	628.32	628.319	628.2518
2	299.1992	299.1993	299.1993	299.19	299.199	299.1535
3	299.1988	299.1993	299.1993	294.51	295.878	296.1073
4	159.733	159.7331	159.7331	159.73	159.265	159.6753
5	159.7329	159.7331	159.7331	159.73	159.733	159.7332
6	159.7324	159.7331	159.7331	159.73	159.733	159.6176
7	159.733	159.7331	159.7331	159.73	159.733	159.5445
8	159.733	159.7331	159.7331	159.73	159.733	159.6311
9	159.7331	159.7331	159.7331	159.73	159.733	159.4948
10	77.3994	77.3999	77.3999	114.8	77.363	77.1423
11	77.3996	77.3999	77.3999	77.4	77.397	77.3767
12	87.6879	92.3999	87.6845	55	92.397	92.2554
13	92.3992	87.6845	92.3999	92.4	91.517	92.0241
$P_{G,Total}$	2519.9999	2519.9999	2519.9999	2520.0000	2520.0000	2520.0082
Min cost	24169.9177	24169.9177	24169.9176	24168.910	24166.8	**24167.7042**
Mean cost	NA	NA	24169.9176	24175.34	24167.37	24273.5221
Max cost	NA	NA	24169.9176	24184.92	24169.41	24428.1236

NA denotes that the value was not available in the literature.
Bold value indicates the best solution found by proposed algorithm.

and Harmony Search (HS) algorithms (MSG-HP) [29], PSO [29], the Newton's second order approach NSOA [28], combines the genetic algorithm GA with active power optimization APO (GA-APO) [28] and genetic algorithm GA [28] as shown in Table 12.

When the adjustable parameters are selected, optimal solution values for the IEEE 30 buses test system are obtained as 199.5996, 20.0008, 24.1658, 17.7409, 19.0252 and 13.7428. The proposed algorithm has found the optimal solution values for the test system by completing 50 iterations in 0.3771 s. As shown in Table 11, RTO algorithm achieves much better optimal solution values when compared to the results in the literature. In other words, the RTO algorithm is 59.9641 R/h better when compared to the NSOA with the best solution value in the literature [28], also is 0.6682 R/h better then MSG-HP algorithm.

Table 16
Comparison of proposed method for test system 3 (case II, load = 2520 MW).

Method	Total cost ($/h)	Method	Total cost ($/h)
SA [26]	24970.91	GA-MU [42]	24170.755
GA [26]	24398.23	IGAMU [42]	24169.979
GA-SA [26]	24275.71	HGA [7]	24169.92
EP-SQP [26]	24266.44	DE [39]	24169.9177
PSO-SQP [26]	24261.05	GSA [27]	24164.251357
UHGA [2]	24172.25	Proposed	**24167.7042**

Bold values indicates the best solution found by proposed algorithm.

5.2.3. Case 3: 13 -unit system with valve-point loading effects

This test case study considers the 13-generator system with valve-point loading effect. The coefficients of fuel cost functions are provided in [8]. The ED problem was solved for two different load levels (PD = 1800 MW and PD = 2500 MW). This test system has many local optima and no global solution has been reported yet. The population size and maximum iteration number are fixed to be 200 and 100, respectively. The obtained results for load demand equal to 1800 MW is presented in Table 13. The results are compared in terms of minimum cost, mean cost, and maximum cost over 50 runs with the results of hybrid multi-agent based PSO (HMAPSO) [30], modified differential evolution algorithm (MDE) [31], self-tuning hybrid differential evolution algorithm (SHDE) [32], pattern search method (PSM) [35], hybrid genetic algorithm

Table 17
Best power output for 15-units system (load = 2630 MW).

Unit power output	Methods								
	GA [44]	PSO [44]	SA [45]	GA [45]	TS [45]	PSO [45]	MTS [45]	CSPSO [46,47]	RTO
P_1 (MW)	415.31	439.12	453.6646	445.5619	453.5374	454.7167	453.9922	455.0000	455.0000
P_2 (MW)	359.72	407.97	377.6091	380.0000	371.9761	376.2002	379.7434	380.0000	380.0000
P_3 (MW)	104.43	119.63	120.3744	129.0605	129.7823	129.5547	130.0000	130.0000	129.9999
P_4 (MW)	74.99	129.99	126.2668	129.5250	129.3411	129.7083	129.9232	130.0000	129.9999
P_5 (MW)	380.28	151.07	165.3048	169.9659	169.5950	169.4407	168.0877	170.0000	170.0000
P_6 (MW)	426.79	460.00	459.2455	458.7544	457.9928	458.8153	460.0000	460.0000	460.0000
P_7 (MW)	341.32	425.56	422.8619	417.9041	426.8879	427.5733	429.2253	430.0000	430.0000
P_8 (MW)	124.79	98.57	126.4025	97.8230	95.1680	67.2834	104.3097	71.7408	70.2250
P_9 (MW)	133.14	113.49	54.4742	54.2933	76.8439	75.2673	35.0358	58.9207	60.1965
P_{10} (MW)	89.26	101.11	149.0879	144.2214	133.5044	155.5899	155.8829	160.0000	159.9999
P_{11} (MW)	60.06	33.91	77.9594	77.3002	68.3087	79.9522	79.8994	80.0000	80.0000
P_{12} (MW)	50.00	79.96	73.9489	77.0371	79.6815	79.8947	79.9037	80.0000	80.0000
P_{13} (MW)	38.77	25.00	25.0022	31.1537	28.3082	25.2744	25.0220	25.0000	25.0000
P_{14} (MW)	41.94	41.41	16.0636	15.0233	17.7661	16.7318	15.2586	15.0000	15.0000
P_{15} (MW)	22.64	35.61	15.0196	33.6125	22.8446	15.1967	15.0796	15.0000	15.0000
Total output (MW)	2668.4	2662.4	2663.29	2661.23	2661.53	2661.19	2661.36	2660.6615	2660.4216
Power loss (MW)	38.28	32.43	33.2737	31.2363	31.4100	31.1697	31.3523	30.6615	30.4216
Total cost ($/h)	33,113	32,858	32786.40	32779.81	32762.12	32724.17	32716.87	32,704	**32701.8145**

Bold value indicates the best solution found by proposed algorithm.

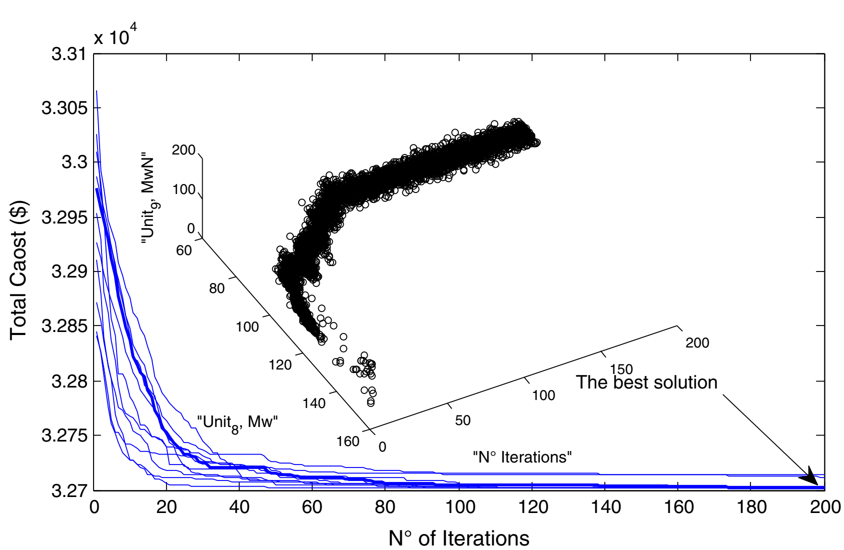

Fig. 10. Convergence characteristic of RTO algorithm for 15-unit system.

Table 18
Convergence results for test case 4.

Methods	Fuel Cost ($/h)				Average time (s)
	Best cost	Average cost	Worst cost	Standard deviation	
GA [44]	33,113	33,228	33,337	0.0087	49.31
PSO [44]	32,858	33,039	33,331	0.007	26.59
SA [45]	32786.40	32869.51	33028.95	112.32	71.25
GA [45]	32779.81	32841.21	33041.64	81.22	48.17
TSA [45]	32762.12	32822.84	32942.71	60.59	26.41
PSO [45]	32724.17	32807.45	32841.38	21.24	13.25
MTS [45]	32716.87	32767.21	32796.15	17.51	3.65
CSPSO [46,47]	32704.45	32704.45	32704.45	0.0004	16.1
RTO	**32701.81**	32704.53	32715.18	5.07	29.38

Bold value indicates the best solution found by proposed algorithm.

(HGA) [36], quantum-inspired PSO (QPSO) [33], PSO [30] and PSO with time varying acceleration coefficients (PSO-TVAC) [34]. The results of the aforementioned methods that presented in Table 14, have been directly quoted from their respective references.

Also, simulation was done for power demand of 2520 MW. The obtained results are presented in Table 15 and compared with the results of hybrid genetic algorithm (HGA) [36], differential evolution (DE) [21], FAPSO-VDE [39], improved coordinated aggregation based PSO (ICA-PSO) algorithm [40] and Iteration PSO (IPSO) [41]. The minimum, average and maximum costs presented in Table 16 are obtained over 50 trial runs. It can be observed from Table 13 that the proposed technique provided almost significantly better results in comparison with the previously developed techniques.

5.2.4. Case 4: 15-units with prohibited operating zones, ramp rate limits, and transmission network losses

Experiments are performed on the 15-unit power system, which considers the prohibited operating zones, ramp rate limits, and transmission network losses. Units 2, 5, 6, and 12 have up to three prohibited operating zones. The system supplies a total load of 2630 MW. The input data and coefficients for transmission network losses are provided in [44], the population size and maximum iteration number are fixed to be 50 and 200, respectively, and the rate values are R_n = 0.4, R_r = 0.3 and R_c = 0.3.

The best results produced by RTO are presented in Table 17, they were compared with the results reported in literatures, such as GA [44], PSO [44], SA [45], TS [45], PSO [45], MTS [45] and CSPSO [46,47], from Table 17, it clearly shows that the proposed method yields lower generation cost than other methods; therefore, this method can provide a significant cost saving compared to other methods. The convergence characteristic of the proposed method is shown in Fig. 10. Table 18 provides the statistical results obtained with 30 trials, such as the minimum cost, average cost, maximum cost, standard deviation, and average execution time. In Table 18, the best results from the proposed RTO are compared with the results reported in literatures [44–47], the results show that the RTO provide better solutions than other methods while satisfying the system constraints exactly, with the cost of $32 701.8145, which is the minimum cost found so far.

6. Conclusions

This paper proposed a rooted tree algorithm which mimics a plant roots in looking for water under the ground. Economic dispatch problem with valve-point effects being attempted using RTO algorithm for various test systems to evaluates the performance of the proposed RTO algorithm. The efficiency of the proposed RTO algorithm has been verified and tested through its application on some well-known functions and by comparing its performance with recent published techniques. RTO algorithm provides a new searching way for solution as one of its characteristics is the large field of research due to the behavior of the roots. The simulation results reveal the superiority of the proposed RTO algorithm in solving the DE problem with valve point effects. Therefore this approach could also be extended to other optimization and control problems.

Appendix A

Transmission loss coefficients for IEEE 30-bus 6-machines system (Test system 2)

$$\beta = \begin{bmatrix} 0.0224 & 0.0103 & 0.0016 & -0.0053 & 0.0009 & -0.0013 \\ 0.0103 & 0.0158 & 0.001 & -0.0074 & 0.0007 & 0.0024 \\ 0.0016 & 0.001 & 0.0474 & -0.0687 & -0.006 & -0.035 \\ -0.0053 & -0.0074 & -0.0687 & 0.3464 & 0.0105 & 0.0534 \\ 0.0009 & 0.0007 & -0.006 & 0.0105 & 0.0119 & 0.0007 \\ -0.0013 & 0.0024 & -0.035 & 0.5340 & 0.0007 & 0.2353 \end{bmatrix}$$

$$\beta_0 = [\,-0.0005 \quad 0.0016 \quad -0.0029 \quad 0.006 \quad 0.0014 \quad 0.0015\,]$$

$$\beta_{00} = 0.0011$$

References

[1] Srinivasa RA, Vaisakh K. Shuffled differential evolution for economic dispatch with valve point loading effects. Electr Power Energy Syst 2013;46:342–52.
[2] Mahor A, Prasad V, Rangnekar S. Economic dispatch using particle swarm optimization: a review. Renew Sust Energy 2009;13(8):2134–41.
[3] Wood AJ, Wollenberg BF. Power generation operation and control. John Wiley & Sons; 1984.
[4] Walters DC, Sheble GB. Genetic algorithm solution of economic dispatch with valve point loading. IEEE Trans Power Syst 1993;8(3):1325–32.
[5] Neto JXV, Bernert DLA, Coelho LS. Improved quantum-inspired evolutionary algorithm with diversity information applied to ED problem with prohibited operating zones. Energy Convers Manage 2011;52(1):8–14.
[6] Sivasubramani S, Swarup KS. Multi-agent based differential evolution algorithm for economic dispatch with generator constraints. J Electr Syst 2009:1–13.
[7] Dakuo H, Fuli W, Zhizhong M. A hybrid genetic algorithm approach based on differential evolution for economic dispatch with valve-point effect. Electr Power Energy Syst 2008;30:31–8.
[8] Coelho LDS, Mariani VC. Particle swarm approach based on quantum mechanics and harmonic oscillator potential well for economic load dispatch with valve-point effects. Energy Convers Manage 2008;49:3080–5.
[9] Alsumait JS, Al-Othman AK, Sykulski JK. Application of PS method to power system valve-point ELD. Electr Power Energy Syst 2007;29:720–30.
[10] Su CT, Lin CT. New approach with a Hopfield modelling framework to economic dispatch. IEEE Trans Power Syst 2000;15(2):541–5.
[11] Zhisheng Z. Quantum-behaved particle swarm optimization algorithm for economic load dispatch of power system. Exp Syst Appl 2010;37:1800–3.
[12] Bhattacharya A, Chattopadhyay PK. Solving complex economic load dispatch problems using biogeography-based optimization. Exp Syst Appl 2010;37:3605–15.

[13] Hosseini SH, Dobakhshari AS, Jalayer R. A novel mathematical-heuristic method for non-convex dynamic economic dispatch. Int Rev Electr Eng 2009;4(1):108–9.
[14] Subramanian S, Anandhakumar R. Dynamic economic dispatch solution using composite cost function. Int Rev Electr Eng 2010;5(4):1664–9 [Part B].
[15] Hooshmand R, Mohammadi AH. Emission and economic load & reserve dispatch with frequency constraints in competitive power market. Int Rev Electr Eng 2008;3(4):682–90.
[16] Holland JH. Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence. 2nd ed. Cambridge: MIT Press; 1992.
[17] Fogel LJ, Owens AJ, Walsh MJ. Artificial intelligence through simulated evolution. New York: John Wiley and Sons; 1966.
[18] Colorni A, Dorigo M, Maniezzo V. Distributed optimization by ant colonies. Actes de la première conférence européenne sur la vie artificielle. Paris, France: Elsevier Publishing; 1991. p. 134–42.
[19] Kennedy J, Eberhart R. Particle swarm optimization. In: Proceedings of IEEE international conference on neural networks IV; 1995. p. 1942–8, doi:http://dx.doi.org/10.1109/ICNN.1995.488968.
[20] Goldberg DE. Genetic algorithms in search, optimization, and machine learning. Boston: Addison-Wesley; 1989.
[21] Kirkpatrick S, Gelatt Jr CD, Vecchi MP. Optimization by simulated annealing. Science 1983;220:671–80.
[22] Holland JH. Adaptation in natural and artificial systems. Ann Arbor: The University of Michigan Press; 1975.
[23] Salomon R. Evolutionary algorithms and gradient search: similarities and differences. IEEE Trans Evol Comput 1997;2(2):45–55.
[24] Bilal Alatas. Uniform Big Bang–chaotic big crunch optimization. Commun Nonlin Sci Numer Simul 2011;16:3696–703.
[25] Yang XS, Hosseinib SSS, Gandomic AH. Firefly algorithm for solving nonconvex economic dispatch problems with valve loading effect. Appl Soft Comput 2012;12(3):1180–6.
[26] Victorie TAA, Jeyakumar AE. Hybrid PSO-SQP for economic dispatch with valve-point effect. Electr Power Syst Res 2004;71:51–9.
[27] Duman S, Güvenç U, Yörükeren N. Gravitational search algorithm for economic dispatch with valve-point effects. Int Rev Ele Eng 2010;5(6):2890–5.
[28] Malik TN, Asar A, Wyne MF, Akhtar S. A new hybrid approach for the solution of nonconvex economic dispatch problem with valve-point effects. Electr Power Syst Res 2010;80:1128–36.
[29] Celal, Yasar, Serdar Özyön. A new hybrid approach for nonconvex economic dispatch problem with valve-point effect. Energy 2011;36:5838–45.
[30] Kumar R, Sharma D, Sadu A. A hybrid multi-agent based particle swarm optimization algorithm for economic power dispatch. Int J Electr Power Energy Syst 2011;33:115–23.
[31] Amjady N, Sharifzadeh H. Solution of non-convex economic dispatch problem considering valve loading effect by a new modified differential evolution algorithm. Int J Elect Power Energy Syst 2010;32:893–903.
[32] Wang SK, Chiou JP, Liu CW. Non-smooth/non-convex economic dispatch by a novel hybrid differential evolution algorithm. IET Gener Transm Distrib 2007;5:793–803.
[33] Meng K, Wang HG, Dong Z, Wong KP. Quantum-inspired particle swarm optimization for valve-point economic load dispatch. IEEE Trans Power Syst 2010;25:215–22.
[34] Chaturvedi KT, Pandit M, Srivastava L. Particle swarm optimization with time varying acceleration coefficients for non-convex economic power dispatch. Int J Elect Power Energy Syst 2009;31:249–57.
[35] AL-Sumait JS, Sykulski JK, AL-Othman AK. Solution of different types of economic load dispatch problems using a PS method. Electr Power Compos Syst 2008;36:250–65.
[36] He D, Wang F, Mao Z. A hybrid GA approach based on differential evolution for economic dispatch with valve-point effect. Int J Elect Power Energy Syst 2008;30:31–8.
[37] Chiang CL. Improved genetic algorithm for economic dispatch of units with valve-point effects and multiple fuels. IEEE Trans Power Syst 2005;20(4):1690–9.
[38] Noman N, Iba H. Differential evolution for economic load dispatch problems. Electr Power Syst Res 2008;78:1322–31.
[39] Niknam T, Doagou Mojarrad H, Zeinoddini Meymand H. A novel H-PSO for economic dispatch with valve-point loading effects. Energy Convers Manage 2011;52:1800–9.
[40] Vlachogiannis JG, Lee KY. Economic load dispatch a comparative study on heuristic optimization techniques with an improved coordinated aggregation-based PSO. IEEE Trans Power Syst 2009;24:991–1001.
[41] Mohammadi-Ivatloo B, Rabiee A, Soroudi A, Ehsan M. Iteration PSO with time varying acceleration coefficients for solving non-convex economic dispatch problems. Electr Power Energy Syst 2012;42:508–16.
[42] Chiang CL. Genetic-based algorithm for power economic load dispatch. IET Gener Transm Distrib 2007;1(2):261–9.
[43] Rashedi E, Nezamabadi-pour H, Saryazdi S. GSA: a gravitational search algorithm. Inf Sci 2009;179:2232–48.
[44] Gaing ZL. Particle swarm optimization to solving the economic dispatch considering the generator constraints. IEEE Trans Power Syst 2003;18(3):1187–95.
[45] Pothiya S, Ngamroo I, Kongprawechnon W. Application of multiple tabu search algorithm to solve dynamic economic dispatch considering generator constraints. Energy Convers Manage 2008;49:506–16.
[46] Park, Jeong Y-W, Shin J-R, Lee KY. An improved particle swarm optimization for nonconvex economic dispatch problems. IEEE Trans Power Syst 2010;25(1):156-166.
[47] Amjady N, Sharifzadeh H. Solution of non-convex economic dispatch problem considering valve loading effect by a new modified differential evolution algorithm. Electr Power Energy Syst 2010;32:893–903.