Pattanaik et al. Protection and Control of Modern Power Systems (2017) 2:17
DOI 10.1186/s41601-017-0049-x

Protection and Control of
Modern Power Systems

REVIEW Open Access

Review on application and comparison of metaheuristic techniques to multi-area economic dispatch problem

Jagat Kishore Pattanaik[1*], Mousumi Basu[1] and Deba Prasad Dash[2]

Abstract

This paper presents both application and comparison of the metaheuristic techniques to multi-area economic dispatch (MAED) problem with tie line constraints considering transmission losses, multiple fuels, valve-point loading and prohibited operating zones. The metaheuristic techniques such as differential evolution, evolutionary programming, genetic algorithm and simulated annealing are applied to solve MAED problem. These metaheuristic techniques for MAED problem are evaluated on three different test systems, both small and large, involving varying degree of complexity and the results are compared against each other.

Keywords: Multi-area economic dispatch, Tie line constraints, Differential evolution, Evolutionary programming, Genetic algorithm, Simulated annealing

1 Introduction

Economic dispatch (ED) is one of the important optimization problems in power system operation. ED allocates the load demand among the committed generators most economically while satisfying the physical and operational constraints in a single area. Generally, the generators are divided into several generation areas interconnected by tie-lines. Multi-area economic dispatch (MAED) is an extension of economic dispatch. MAED determines the generation level and interchange power between areas such that total fuel cost in all areas is minimized while satisfying power balance constraints, generating limits constraints and tie-line capacity constraints.

The economic dispatch problem is frequently solved without considering transmission constraints. However, some researchers have taken transmission capacity constraints into account. Shoults et al. [1] solved economic dispatch problem considering import and export constraints between areas. This study provides a complete formulation of multi-area generation scheduling, and a framework for multi-area studies. Romano et al. [2] presented the Dantzig–Wolfe decomposition principle to the constrained economic dispatch of multi-area systems. An application of linear programming to transmission constrained production cost analysis was proposed in Ref. [3]. Helmick et al. [4] solved multi-area economic dispatch with area control error. Wang and Shahidehpour [5] proposed a decomposition approach for solving multi-area generation scheduling with tie-line constraints using expert systems. Network flow models for solving the multi-area economic dispatch problem with transmission constraints have been proposed by Streiffert [6]. An algorithm for multi-area economic dispatch and calculation of short range margin cost based prices has been presented by Wernerus and Soder [7], where the multi-area economic dispatch problem was solved via Newton–Raphson's method. Yalcinoz and Short [8] solved multi-area economic dispatch problems by using Hopfield neural network approach. Jayabarathi et al. [9] solved multi-area economic dispatch problems with tie line constraints using evolutionary programming. The direct search method for solving economic dispatch problem considering transmission capacity constraints was presented in Ref. [10]. Chen [11] develops a hybrid approach of combining sequential dispatch with a direct search method to deal with the multi-product and multi-area

* Correspondence: jagat.ju@gmail.com
[1]Department of Power Engineering, Jadavpur University, Salt Lake City, Kolkata 700098, India
Full list of author information is available at the end of the article

electricity market dispatch problem. But these methods did not consider transmission loss.

With the emergence of metaheuristic techniques, attention has been gradually shifted to applications of such technology-based approaches to handle the complexity involved in real world problems. Metaheuristic techniques have been given much attention by many researchers due their ability to seek for the near global optimal solution.

This paper investigates the applicability of the following four metaheuristic techniques in the MAED problem: differential evolution (DE), evolutionary programming (EP), genetic algorithm (GA), and simulated annealing (SA).

Here, three types of MAED problems have been considered. These are A) multi area economic dispatch with quadratic cost function prohibited operating zones and transmission losses (MAEDQCPOZTL) B) multi area economic dispatch with valve point loading (MAEDVPL) C) multi area economic dispatch with valve point loading multiple fuel sources and transmission losses (MAEDVPLMFTL).

The metaheuristic techniques are evaluated against three different test systems for comparison with each other.

2 Problem formulation

The objective of MAED is to minimize the total cost of supplying loads to all areas while satisfying power balance constraints, generating limits constraints and tie-line capacity constraints.

Three different types of MAED problems have been considered.

2.1 MAEDQCPOZTL

The objective function F_t, total cost of committed generators of all areas, of MAED problem may be written as

$$F_t = \sum_{i=1}^{N}\sum_{j=1}^{M_i} F_{ij}(P_{ij}) = \sum_{i=1}^{N}\sum_{j=1}^{M_i} a_{ij} + b_{ij}P_{ij} + c_{ij}P_{ij}^2 \tag{1}$$

where $F_{ij}(P_{ij})$ is the cost function of jth generator in area i and is usually expressed as a quadratic polynomial; a_{ij}, b_{ij} and c_{ij} are the cost coefficients of jth generator in area i; N is the number of areas, M_i is the number of committed generators in area i; P_{ij} is the real power output of jth generator in area i. The MAED problem minimizes F_t subject to the following constraints

2.1.1 Real power balance constraint

$$\sum_{j=1}^{M_i} P_{ij} = P_{Di} + P_{Li} + \sum_{k,k\neq i} T_{ik} \qquad i\in N \tag{2}$$

The transmission loss P_{Li} of area i may be expressed by using B-coefficients as

$$P_{Li} = \sum_{l=1}^{M_i}\sum_{j=1}^{M_i} P_{ij}B_{ilj}P_{il} + \sum_{j=1}^{M_i} B_{0ij}P_{ij} + B_{00i} \tag{3}$$

where P_{Di} is the real power demand of area i; T_{ik} is the tie line real power transfer from area i to area k. T_{ik} is positive when power flows from area i to area k and T_{ik} is negative when power flows from area k to area i.

2.1.2 Tie line capacity constraints

The tie line real power transfer T_{ik} from area i to area k should not exceed the tie line transfer capacity for security consideration.

$$-T_{ik}^{max} \le T_{ik} \le T_{ik}^{max} \tag{4}$$

where T_{ik}^{max} is the power flow limit from area i to area k and - T_{ik}^{max} is the power flow limit from area k to area i.

2.1.3 Real power generation capacity constraints

The real power generated by each generator should be within its lower limit P_{ij}^{min} and upper limit P_{ij}^{max}, so that

$$P_{ij}^{min} \le P_{ij} \le P_{ij}^{max} \qquad i\in N \text{ and } j\in M_i \tag{5}$$

2.1.4 Prohibited operating zone

The prohibited operating zones are the range of power output of a generator where the operation causes undue vibration of the turbine shaft bearing caused by opening or closing of the steam valve. Normally operation is avoided in such regions. The feasible operating zones of unit can be described as follows:

$$\begin{aligned} &P_{ij}^{min} \le P_{ij} \le P_{ij,1}^{l} \\ &P_{ij,m-1}^{u} \le P_{ij} \le P_{ij,m}^{l} \ ; \quad m = 2, 3, \dots, n_{ij} \\ &P_{ij,n_{ij}}^{u} \le P_{ij} \le P_{ij}^{max} \end{aligned} \tag{6}$$

where m represents the number of prohibited operating zones of j the generator in area i. $P_{ij,m-1}^{u}$ is the upper limit of $(m-1)$ th prohibited operating zone of j the generator in area i. $P_{ij,m}^{l}$ is the lower limit of mth prohibited operating zone of j the generator in area i. Total number of prohibited operating zone of j the generator in area i is n_{ij}.

2.2 MAEDVPL

To model the effect of valve-points, a recurring rectified sinusoid contribution is added to the quadratic function [12]. The fuel cost function considering valve-point loadings of the generator is given as

$$F_t = \sum_{i=1}^{N}\sum_{j=1}^{M_i} F_{ij}(P_{ij}) = \sum_{i=1}^{N}\sum_{j=1}^{M_i} a_{ij} + b_{ij}P_{ij} + c_{ij}P_{ij}^2 + \left| d_{ij} \times \sin\left\{ e_{ij} \times \left(P_{ij}^{\min} - P_{ij} \right) \right\} \right| \tag{7}$$

where d_{ij} and e_{ij} are cost coefficients of j th generator in area i due to valve-point effect. The objective of MAEDVPL is to minimize F_t subject to the constraints given in (2), (4) and (5). Here transmission loss (P_L) is not considered.

2.3 MAEDVPLMFTL

Since generators are practically supplied with multi-fuel sources [13], each generator should be represented with several piecewise quadratic functions superimposed sine terms reflecting the effect of fuel type changes and the generator must identify the most economical fuel to burn. The fuel cost function of the j th generator in area i with N_F fuel types considering valve-point loading is expressed as

$$F_{ij}(P_{ij}) = a_{ijm} + b_{ijm}P_{ij} + c_{ijm}{P_{ij}}^2 + \left| d_{ijm} \times \sin\left\{ e_{ijm} \times \left(P_{ijm}^{\min} - P_{ij} \right) \right\} \right| \tag{8}$$

if $P_{ijm}^{\min} \le P_{ij} \le P_{ijm}^{\max}$ for fuel type m and $m = 1, 2, \ldots, N_F$

The objective function F_t is given by

$$F_t = \sum_{i=1}^{N}\sum_{j=1}^{M_i} F_{ij}(P_{ij}) \tag{9}$$

The objective function F_t is to be minimized subject to the constraints given in (2), (4) and (5).

3 Determination of generation level of slack generator

M_i committed generators in area i deliver their power output subject to the power balance constraint (2), tie line capacity constraints (4) and the respective generation capacity constraints (5). Assuming the power loading of first ($M_i - 1$) generators are known, the power level of the M_i th generator (i.e. the slack generator) is given by

$$P_{iM_i} = P_{Di} + P_{Li} + \sum_{k,k\neq i} T_{ik} - \sum_{j=1}^{M_i-1} P_{ij} \tag{10}$$

The transmission loss P_{Li} is a function of all generator outputs including the slack generator and it is given by

$$P_{Li} = \sum_{l=1}^{M_i-1}\sum_{j=1}^{M_i-1} P_{ij}B_{ilj}P_{il} + 2P_{iM_i}\left(\sum_{j=1}^{M_i-1} B_{iM_ij}P_{ij}\right) + B_{iM_iM_i}P_{iM_i}^2 + \sum_{j=1}^{M_i-1} B_{0ij}P_{ij} + B_{0iM_i}P_{iM_i} + B_{00i} \tag{11}$$

Expanding and rearranging, Eq. (10) becomes

$$B_{iM_iM_i}P_{iM_i}^2 + \left(2\sum_{j=1}^{M_i-1} B_{iM_ij}P_{ij} + B_{0iM_i} - 1\right)P_{iM_i} + \left(P_{Di} + \sum_{k,k\neq i} T_{ik} + \sum_{j=1}^{M_i-1}\sum_{l=1}^{M_i-1} P_{ij}B_{ilj}P_{il} + \sum_{j=1}^{M_i-1} B_{0ij}P_{ij} - \sum_{j=1}^{M_i-1} P_{ij} + B_{00i}\right) = 0 \tag{12}$$

The loading of the slack generator (i.e. M_i th) can then be found by solving Eq. (12) using standard algebraic method

4 Metaheuristic techniques

Several metaheuristic techniques have evolved in recent past that facilitate to solve optimization problems which were previously difficult or impossible to solve. These techniques include differential evolution, evolutionary programming, genetic algorithm, simulated annealing, etc. Reports of applications of each of these techniques have been widely published. The most important advantage of metaheuristic techniques lies in the fact that they are not limited by restrictive assumptions about the search space like continuity, existence of derivative of objective function, etc.

These methods share some similarities. The DE is introduced first, and followed by EP, GA and SA.

4.1 Differential evolution

Differential Evolution (DE) [14] is a type of evolutionary algorithm originally proposed by Price and Storn [15] for optimization problems over a continuous domain. DE is exceptionally simple, significantly faster and robust. The basic idea of DE is to adapt the search during the evolutionary process. At the start of the evolution, the perturbations are large since parent populations are far away from each other. As the evolutionary process matures, the population converges to a small region and the perturbations adaptively become small. As a result, the evolutionary algorithm performs a global exploratory search during the early stages of the evolutionary process and local exploitation during the mature stage of the search. In DE the fittest of an offspring competes one-to-one with that of corresponding parent which is different from other evolutionary algorithms. This one-to-one competition gives rise to faster convergence rate.

Price and Storn gave the working principle of DE with simple strategy in [14]. Later on, they suggested ten different strategies of DE [15]. Strategy-7 (DE/rad/1/bin) is the most successful and widely used strategy. The key parameters of control in DE are population size (N_P), scaling factor (F) and crossover constant (C_R). The optimization process in DE is carried out with three basic operations: mutation, crossover and selection. The DE algorithm is described as follows:

4.1.1 Initialization

The initial population of N_P vectors is randomly selected based on uniform probability distribution for all variables to cover the entire search uniformly. Each individual X_i is a vector that contains as many parameters as the problem decision variables D. Random values are assigned to each decision parameter in every vector according to:

$$X_{ij}^0 \sim U\left(X_j^{\min}, X_j^{\max}\right) \tag{13}$$

where $i = 1,\ldots,N_P$ and $j = 1,\ldots,D$; $X_j^{\min}$ and $X_j^{\max}$ are the lower and upper bounds of the j th decision variable; $U\left(X_j^{\min}, X_j^{\max}\right)$ denotes a uniform random variable ranging over $\left[X_j^{\min}, X_j^{\max}\right]$. X_{ij}^0 is the initial j th variable of i th population. All the vectors should satisfy the constraints. Evaluate the value of the cost function $f\left(X_i^0\right)$ of each vector.

4.1.2 Mutation

DE generates new parameter vectors by adding the weighted difference vector between two population members to a third member. For each target vector X_i^g at g th generation the noisy vector $X_i^{/g}$ is obtained by

$$X_i^{/g} = X_a^g + S_F\left(X_b^g - X_c^g\right), \qquad i \in N_P \tag{14}$$

where X_a^{ga}, X_b^g and X_c^g are selected randomly from N_P vectors at g th generation and $a \neq b \neq c \neq i$. The scaling factor (S_F), in the range $0 < S_F \leq 1.2$, controls the amount of perturbation added to the parent vector. The noisy vectors should satisfy the constraint.

4.1.3 Crossover

Perform crossover for each target vector X_i^g with its noisy vector $X_i^{/g}$ and create a trial vector $X_i^{//g}$ such that

$$X_i^{//g} = \begin{cases} X^{i/g} & , \text{ if } \rho \leq C_R \\ X_i^g & , \text{ Otherwise} \end{cases} \quad , \quad i \in N_P \tag{15}$$

where ρ is an uniformly distributed random number within [0, 1]. The crossover constant (C_R), in the range $0 \leq C_R \leq 1$, controls the diversity of the population and aids the algorithm to escape from local optima.

4.1.4 Selection

Perform selection for each target vector, X_i^g by comparing its cost with that of the trial vector, $X_i^{//g}$. The vector that has lesser cost of the two would survive for the next generation.

$$X_i^{g+1} = \begin{cases} X_i^{//g} & , \text{ if } f\left(X_i^{//g}\right) \leq f\left(X_i^g\right) \\ X_i^g & , \text{ otherwise} \end{cases} \quad , \ i \in N_P \tag{16}$$

The process is repeated until the maximum number of generations or no improvement is seen in the best individual after many generations.

4.2 Evolutionary programming

Evolutionary Programming (EP) [16] is a technique in the field of evolutionary computation. It seeks the optimal solution by evolving a population of candidate solutions over a number of generations or iterations. During each iteration, a second new population is formed from an existing population through the use of a mutation operator. This operator produces a new solution by perturbing each component of an existing solution by a random amount. The degree of optimality of each of the candidate solutions or individuals is measured by their fitness, which can be defined as a function of the objective function of the problem. Through the use of a competition scheme, the individuals in each population compete with each other. The winning individuals form a resultant population, which is regarded as the next generation. For optimization to occur, the competition scheme must be such that the more optimal solutions have a greater chance of survival than the poorer solutions. Through this the population evolves towards the global optimal point. The algorithm is described as follows:

1) Initialization: The initial population of control variables is selected randomly from the set of uniformly distributed control variables ranging over their upper and lower limits. The fitness score f_i is obtained according to the objective function and the environment.
2) Statistics: The maximum fitness $f_{\max}$, minimum fitness $f_{\min}$, the sum of fitness $\sum f$, and average fitness f_{avg} of this generation are calculated.
3) Mutation: Each selected parent, for example X_i, is mutated and added to its population with the following rule:

$$X_{i+m,j} = X_{ij} + N\left(0, \beta\left(\overline{x}_j - \underline{x}_j\right)\frac{f_i}{f_{\max}}\right), \quad j \in n, i \in N_P \tag{17}$$

where n is the number of decision variables in an individual, N_P is the population size, X_{ij} denotes the j th element of the i th individual; $N(\mu, \sigma^2)$ represents a Gaussian random variable with mean μ and variance σ^2; $f_{\max}$ is the maximum fitness of the old generation which is obtained in statistics; $\overline{x}_j$ and $\underline{x}_j$ are respectively maximum and minimum limits of the j th element; and β is the mutation scale, $0 < \beta \leq 1$, that could be adaptively decreased during generations. If any mutated value exceeds its limit, it will be given the limit value. The mutation process allows an individual with larger fitness to produce more offspring for the next generation.

4) Competition: Several individuals (k) which have the best fitness are kept as the parents for the next generation. Other individuals in the combined population of size ($2N_P - k$) have to compete with each other to get their chances for the next generation. A weight value w_i of the i th individual is calculated by the following competition:

$$w_i = \sum_{t=1}^{N_t} w_{i,t} \tag{18}$$

where N_t is the competition number generated randomly; $w_{i,t}$ is either 0 for loss or 1 for win as the i th individual competes with a randomly selected (rth) individual in the combined population. The value of $w_{i,t}$ is given in the following equation:

$$w_{i,t} = \begin{cases} 1 & \text{if } \; f_i < f_r \\ 0 & \text{Otherwise} \end{cases} \tag{19}$$

where f_r is the fitness of randomly selected r th individuals, and f_i is the fitness of the i th individual. When all $2N_P$ individuals, get their competition weights, they will be ranked in a descending order according to their corresponding value w_i. The first m individuals are selected along with their corresponding fitness f_i to be the bases for the next generation. The maximum, minimum and the average fitness and the sum of the fitness of the current generation are then calculated in the statistics.

5) Convergence test: If the convergence condition is not met, the mutation and competition will run again. The maximum generation number can be used for convergence condition. Other criteria, such as the ratio of the average and the maximum fitness of the population is computed and generations are repeated until

$$\left\{f_{avg}/f_{\max}\right\} \geq \delta \tag{20}$$

where δ should be very close to 1, which represents the degree of satisfaction. If the convergence has reached a given accuracy, an optimal solution has been found for an optimization problem.

4.3 Genetic algorithm

Genetic algorithm [17] is based on the mechanics of natural selection. An initial population of candidate solutions is created randomly. Each of these candidate solutions is termed as individual. Each individual is assigned a fitness, which measures its quality. During each generation of the evolutionary process, individuals with higher fitness are favored and more probabilities to be selected as parents. After parents are selected for reproduction, they produce children via the processes of crossover and mutation. The individuals formed during reproduction explore different areas of the solution space. These new individuals replace lesser-fit individuals of the existing population.

Due to difficulties of binary representation when dealing with continuous search space with large dimensions, the proposed approach has been implemented using real-coded genetic algorithm (RCGA) [18]. The simulated Binary Crossover (SBX) and polynomial mutation are explained as follows.

4.3.1 Simulated binary crossover (SBX) operator

The procedure of computing child populations c_1 and c_2 from two parent populations y_1 and y_2 under SBX operator as follows:

1. Create a random number u between 0 and 1.
2. Find a parameter γ using a polynomial probability distribution as follows:

$$\gamma = \begin{cases} (u\alpha)^{1/(\eta_c+1)}, & \text{if} \quad u \leq \frac{1}{\alpha} \\ (1/(2-u\alpha))^{1/(\eta_c+)}, & \text{otherwise} \end{cases} \tag{21}$$

where $\alpha = 2 - \beta^{-(\eta_c+1)}$ and $\beta = 1 + \frac{2}{y_2 - y_1}\min[(y_1 - y_l), (y_u - y_2)]$

Here, the parameter y is assumed to vary in $[y_l, y_u]$. Here, the parameter η_c is the distribution index for SBX and can take any non-negative value. A small value of η_c allows the creation of child populations far away from parents and a large value restricts only near-parent populations to be created as child populations.

3. The intermediate populations are calculated as follows:

Table 1 Simulation results for test system 1

	DE	SA	EP	RCGA
$P_{1,1}$ (MW)	500.0000	500.0000	500.0000	500.0000
$P_{1,2}$ (MW)	200.0000	200.0000	200.0000	200.0000
$P_{1,3}$ (MW)	150.0000	150.0000	149.9919	149.6328
$P_{2,1}$ (MW)	204.3341	204.2157	206.4493	205.9398
$P_{2,2}$ (MW)	154.7048	155.0575	154.8892	155.8322
$P_{2,3}$ (MW)	67.5770	67.3516	65.2717	65.2209
T_{12} (MW)	82.7731	82.7731	82.7652	82.4135
P_{L1} (MW)	9.4269	9.4269	9.4267	9.4193
P_{L2} (MW)	4.1890	4.1979	4.1754	4.2064
Cost ($/h)	12255.39	12255.39	12255.43	12256.23
CPU time (second)	17.6875	14.7656	21.3281	24.2031

$$\begin{aligned} c_{p1} &= 0.5[(y_1 + y_2) - \gamma(|y_2 - y_1|)] \\ c_{p2} &= 0.5[(y_1 + y_2) + \gamma(|y_2 - y_1|)] \end{aligned} \quad (22)$$

Each variable is chosen with a probability p_c and the above SBX operator is applied variable-by-variable.

4.3.2 Polynomial mutation operator

A polynomial probability distribution is used to create a child population in the vicinity of a parent population under the mutation operator. The following procedure is used:

1. Create a random number u between 0 and 1.
2. Calculate the parameter δ as follows:

$$\delta = \begin{cases} \left[2u + (1-2u)(1-\phi)^{(\eta_m+1)}\right]^{\frac{1}{(\eta_m+1)}} - 1, & \text{if } u \le 0.5 \\ 1 - \left[2(1-u) + 2(u-0.5)(1-\phi)^{(\eta_m+1)}\right]^{\frac{1}{(\eta_m+1)}}, & \text{otherwise} \end{cases} \quad (23)$$

where $\phi = \frac{\min[(c_p - y_l), (y_u - c_p)]}{(y_u - y_l)}$

The parameter η_m is the distribution index for mutation and takes any non-negative value.

3. Calculate the mutated child as follows:

$$c_1 = c_{p1} + \delta(y_u - y_l)$$

$$c_2 = c_{p2} + \delta(y_u - y_l)$$

The perturbance in the population can be adjusted by varying η_m and p_m with generations as given below:

$$\eta_m = \eta_{m\min} + gen \quad (24)$$

$$p_m = \frac{1}{n} + \frac{gen}{gen_{\max}}\left(1 - \frac{1}{n}\right) \quad (25)$$

where $\eta_{m\min}$ is the user defined minimum value for η_m, p_m is the probability of mutation, and n is the number of decision variables

4.4 Simulated annealing

Simulated annealing [19] is a powerful optimization technique which exploits the resemblance between a minimization process and the cooling of molten metal. The physical annealing process is simulated in the simulated

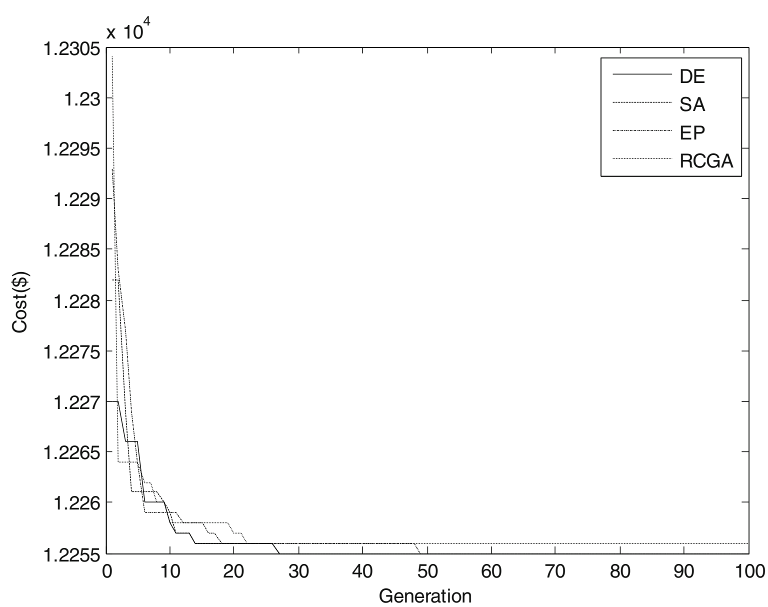

Fig. 1 Cost convergence characteristic of test system 1

Table 2 Simulation results for test system 2

Power (MW)	DE		SA		EP		RCGA	
		Fuel		Fuel		Fuel		Fuel
$P_{1,1}$ (MW)	225.9431	2	228.1730	2	223.8491	2	239.0958	2
$P_{1,2}$ (MW)	211.1594	1	213.3402	1	209.5759	1	216.1166	1
$P_{1,3}$ (MW)	489.9216	2	482.8722	2	496.0680	2	484.1506	2
$P_{1,4}$ (MW)	240.6232	3	242.6425	3	237.9954	3	240.6228	3
$P_{2,1}$ (MW)	254.0397	1	253.5059	1	259.4299	1	259.6639	1
$P_{2,2}$ (MW)	235.4927	3	236.5760	3	228.9422	3	219.9107	3
$P_{2,3}$ (MW)	263.8837	1	266.6356	1	264.1133	1	254.5140	1
$P_{3,1}$ (MW)	237.0006	3	234.3130	3	238.2280	3	231.3565	3
$P_{3,2}$ (MW)	328.7373	1	325.9516	1	331.2982	1	341.9624	1
$P_{3,3}$ (MW)	248.8607	1	251.4034	1	246.6025	1	248.2782	1
T_{21} (MW)	99.8288		100		100		93.1700	
T_{31} (MW)	99.7334		99.8797		100		93.8739	
T_{32} (MW)	31.2615		28.1853		32.5231		43.7824	
P_{L1} (MW)	17.2095		16.9000		17.4884		17.0297	
P_{L2} (MW)	9.8488		9.9028		10.0085		9.7010	
P_{L3} (MW)	8.6037		8.6030		8.6056		8.9408	
Cost ($/h)	653.9995		654.0916		655.1716		657.3325	
CPU time (second)	95.0351		10.0156		108.0625		133.8438	

annealing (SA) technique for the determination of global or near-global optimum solutions for optimization problems. In this algorithm, a parameter T_0, called temperature, is defined. Starting from a high temperature, a molten metal is cooled slowly until it is solidified at a low temperature. The iteration number in the SA technique is analogous to the temperature level. During each iteration, a candidate solution is generated. If this solution is a better solution, it will be accepted and used to generate yet another candidate solution. If it is a deteriorated solution, the solution will be accepted when its probability of acceptance $Pr(\Delta)$ as given by Eq. (26) is greater than a randomly generated number between 0 and 1:

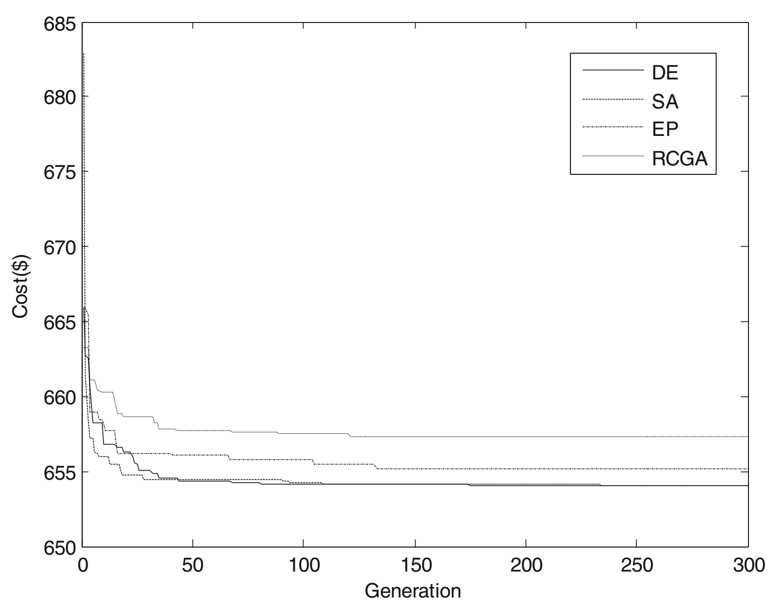

Fig. 2 Cost convergence characteristic of test system 2

Table 3 Simulation results for test system 3

Power (MW)	DE	SA	EP	RCGA	Power (MW)	DE	SA	EP	RCGA
$P_{1,1}$	111.5448	110.9120	107.6644	95.7552	$P_{3,4}$	523.4073	523.3366	525.7752	518.1120
$P_{1,2}$	111.7092	111.8740	112.0673	88.5828	$P_{3,5}$	523.7703	525.5247	531.2092	538.1994
$P_{1,3}$	98.2429	110.2589	91.8132	97.6063	$P_{3,6}$	523.5424	523.2794	513.5659	527.4775
$P_{1,4}$	179.8834	179.7351	175.3171	126.4966	$P_{3,7}$	10.1621	10.0002	11.3612	24.4133
$P_{1,5}$	95.9500	88.8739	92.4242	71.0127	$P_{3,8}$	10.1326	10.0006	10.0000	28.9856
$P_{1,6}$	139.3533	68.0000	112.5634	116.3866	$P_{3,9}$	10.6366	10.0006	10.0000	28.8571
$P_{1,7}$	259.3395	184.9322	257.5370	244.5857	$P_{3,10}$	88.1189	93.2065	78.3523	87.9016
$P_{1,8}$	285.3569	285.0432	297.3619	210.6920	$P_{4,1}$	161.2220	190.0000	162.4480	159.7482
$P_{1,9}$	284.9627	284.6015	285.2035	236.1685	$P_{4,2}$	189.5668	189.9990	166.3508	153.6255
$P_{1,10}$	130.2217	130.0008	134.5862	130.1286	$P_{4,3}$	189.9240	159.7546	190.0000	160.4706
$P_{2,1}$	243.6005	168.6194	162.4313	367.4862	$P_{4,4}$	165.6621	165.6736	178.4541	169.9359
$P_{2,2}$	95.3890	318.3986	217.8387	297.9501	$P_{4,5}$	165.4321	164.8248	168.0752	168.5220
$P_{2,3}$	214.5171	304.5197	125.0000	394.9246	$P_{4,6}$	164.9868	196.1794	174.4529	172.2638
$P_{2,4}$	394.0808	394.2792	384.0187	370.3473	$P_{4,7}$	109.8137	89.1143	77.3875	91.2423
$P_{2,5}$	394.2481	469.0618	397.6902	455.7123	$P_{4,8}$	109.7935	89.1147	90.1059	86.4778
$P_{2,6}$	394.4360	304.5195	407.4993	393.9673	$P_{4,9}$	90.1543	104.7206	109.5654	88.3627
$P_{2,7}$	489.9552	489.2801	500.0000	424.1994	$P_{4,10}$	459.1140	458.7992	549.0335	279.2691
$P_{2,8}$	488.8885	489.2803	480.8874	484.5498	T_{12}	172.0652	192.6532	200	−71.7855
$P_{2,9}$	511.4713	511.2790	524.8487	528.4148	T_{31}	−36.3060	160.6028	17.5885	161.9336
$P_{2,10}$	511.4125	511.2805	499.7857	511.3403	T_{32}	191.1128	−46.9736	200	95.2833
$P_{3,1}$	523.2896	524.8208	523.4522	525.4497	T_{41}	86.8070	52.8188	90.8733	−76.1340
$P_{3,2}$	523.2950	523.2802	526.5051	510.7391	T_{42}	98.8231	93.8021	100	−52.3900
$P_{3,3}$	523.4129	433.6204	537.3675	533.6399	T_{43}	45.0391	86.5590	100	83.4418
Total cost ($/h)						121794.8	123337.1	123591.9	128046.5
CPU time (second)						134.8125	29.2813	144.5000	160.5313

$$\mathrm{P}r(\Delta) = 1/(1 + \exp(\Delta/\mathrm{T}_{\nu})) \tag{26}$$

where Δ is the amount of deterioration between the new and the current solutions and T_{ν} is the temperature at which the new solution is generated. Accepting deteriorated solutions in the above manner enables the SA solution to 'jump' out of the local optimum solution points and to seek the global optimum solution. In forming the new solution the current solution is perturbed [20] according to the Gaussian probability distribution function (GPDF). The mean of the GPDF is taken to be the current solution, and its standard deviation is given by the product of the temperature and a scaling factor σ. The value of σ is less than one, and together with the value of temperature, it governs the size of the neighborhood space of the current solution and hence the amount of perturbation. The amount of perturbation is dependent upon the temperature when σ is kept at a constant value. In each iteration, the procedure for generating and testing the candidate solution is repeated for a specified number of trials so that thermal equilibrium is reached for each temperature. The last accepted candidate solution is then taken as the starting solution for the generation of candidate solutions in the next iteration. Simulated annealing with a slow cooling schedule usually has larger capacity to find the optimal solution than that of a fast cooling schedule. The reduction of the temperature in successive iterations is governed by the following geometric function [19]

$$\mathrm{T}_{\nu} = r^{(\nu-1)}\mathrm{T}_0 \tag{27}$$

where ν is the iteration number and r is temperature reduction factor. T_0 is the initial temperature, the value of which can be set arbitrarily or estimated using the method described in reference [20]. The iterative process is terminated when there is no significant improvement in the solution after a prespecified number of iterations. It can also be terminated when the maximum number of iterations is reached.

4.5 Simulation results

A comparative study is performed for the four metaheuristic techniques by solving the MAED problem for three different test systems. All metaheuristic techniques

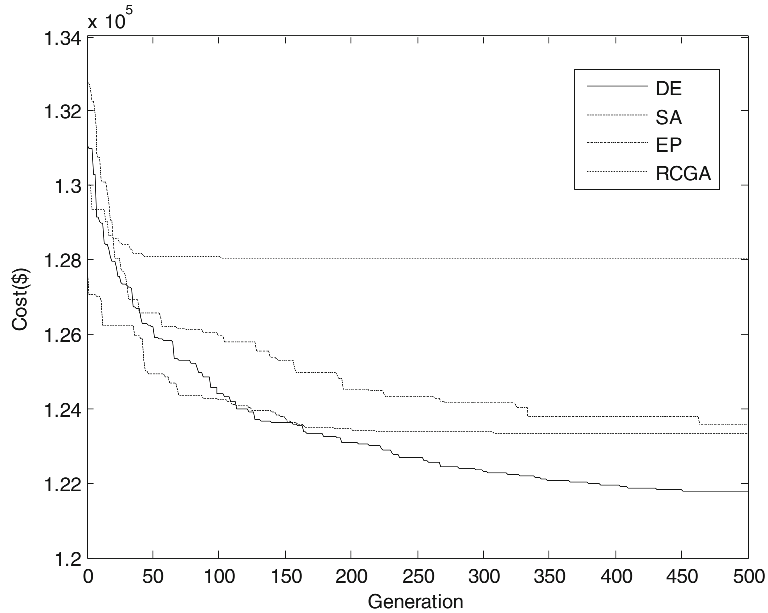

Fig. 3 Cost convergence characteristic of test system 3

for the MAED problems are implemented by using MATLAB 7.0 on a PC (Pentium-IV, 80 GB, 3.0 GHz).

The initial temperature (T_0) of SA algorithm has been determined by using the procedures described in [20]. As per guideline [19], the value of r lies in the range from 0.80 to 0.99. For seeking the optimal solution, the value of r is required to be set close to 0.99 so that a slow cooling process is simulated. The appropriate setting of r is set by experimenting with its value in the range from 0.95 to 0.99, and this value is found to be 0.98. Number of trials at each temperature has been taken 30. In this paper, iterative process is terminated when the maximum number of iterations is reached.

4.5.1 Test system 1

This system consists of two areas. Each area consists of three generators with prohibited operating zones. Transmission loss is considered here. The generator data has modified from [21]. The generator data and B-coefficients are given in the Appendix 1. The percentage of the total load demand in area 1 is 60% and 40% in area 2. The total load demand is 1263 MW and power flow limit of the system is 100 MW.

The problem is solved by using DE, EP, RCGA, and SA. In case of DE, the population size, scaling factor, and crossover rate have been selected as 100, 0.75, and 1.0 respectively for the test system under consideration. The population size and scaling factor have been selected as 100, and 0.1 respectively in case of EP. In case of RCGA, the population size, crossover and mutation probabilities have been selected as 100, 0.9 and 0.2 respectively.

Maximum number of generations has been selected 100 for all the four metaheuristic techniques discussed in this paper.

Results obtained from the four metaheuristic techniques i.e. DE, EP, RCGA, and SA have been summarized in Table 1. Figure 1 gives the comparison of convergence of minimum total cost obtained by DE, EP, RCGA, and SA.

4.5.2 Test system 2

This system comprises ten generators with valve-point loading and multi-fuel sources having three fuel options. Transmission loss is considered here. The generator data has been taken from [13]. The total load demand is 2700 MW. The ten generators are divided into three areas. Area 1 consists of the first four units; area 2 includes the next three units and area 3 includes the last three units. The load demand in area 1 is assumed as 50% of the total demand. The load demand in area 2 is assumed as 25% and in area 3 is taken as 25% of the total demand. The power flow limit from area 1 to area 2 or from area 2 to area 1 is 100 MW. The power flow limit from area 1 to area 3 or from area 3 to area 1 is 100 MW. Also the power flow limit from area 2 to area 3 or from area 3 to area 2 is 100 MW. The B-coefficients are given in the Appendix 2.

The problem is solved by using four metaheuristic techniques i.e. DE, EP, RCGA, and SA.

In case of DE, the population size, scaling factor, and crossover rate have been selected as 200, 0.75, and 1.0 respectively for the test system under consideration. The population size and scaling factor have been selected as 100, and 0.1 respectively in case of EP. In case of RCGA, the population size, crossover and mutation probabilities have been selected as 100, 0.9 and 0.2 respectively. Maximum number of generations has been selected 300 for DE, EP, RCGA, and SA.

Results obtained from DE, EP, RCGA and RCGA have been presented in Table 2. The cost convergence characteristic of this test system obtained from DE, EP, RCGA and SA is shown in Fig. 2.

4.5.3 Test system 3

This system comprises forty generators with valve-point loading. The generator data has been taken from [22]. The total load demand is 10500 MW. The forty generators are divided into four areas. Area 1 includes first ten units and 15% of the total load demand. Area 2 has second ten generators and 40% of the total load demand. Area 3 consists of third ten generators and 30% of the total load demand. Area four includes last ten generators and 15% of the total load demand. The power flow limit from area 1 to area 2 or from area 2 to area 1 is 200 MW. The power flow limit from area 1 to area 3 or from area 3 to area 1 is 200 MW. The power flow limit from area 2 to area 3 or from area 3 to area 2 is 200 MW. The power flow limit from area 4 to area 1 or from area 1 to area 4 is 100 MW. The power flow limit from area 4 to area 2 or from area 2 to area 4 is 100 MW. The power flow limit from area 4 to area 3 or from area 3 to area 4 is 100 MW.

Transmission loss is neglected here.

Four metaheuristic techniques i.e. DE, EP, RCGA, and SA have been used to solve the problem.

The population size, scaling factor, and crossover rate have been been selected as 400, 0.75 and 1.0 respectively in case of DE. In EP, the population size and scaling factor have been selected 200 and 0.1 respectively. In case of RCGA, the population size, crossover and mutation probabilities have been selected as 200, 0.9 and 0.2 respectively. Maximum number of generations has been selected 500 for DE, EP, RCGA and SA.

Results obtained from DE, EP, RCGA and SA have been depicted in Table 3. The cost convergence characteristic of this test system obtained from DE, EP, RCGA and SA is shown in Fig. 3.

From Tables 1, 2 and 3, it can be inferred that, the lowest minimum total cost amongst the four is achieved by DE, followed by SA. Minimum total cost obtained by EP is more than DE and SA. RCGA is the worst performer. The CPU time requirement is least in case of SA and highest in the case of RCGA amongst the four metaheuristic techniques discussed in the paper.

5 Conclusion

In this paper, a comparison analysis has been done for the four metaheuristic techniques viz., differential evolution, evolutionary programming, real coded genetic algorithm and simulated annealing technique for multi-area economic dispatch problem considering transmission losses, multiple fuels, valve-point loading and prohibited operating zones with respect to minimum cost and CPU time. Differential evolution achieves the lowest minimum cost and SA requires least CPU time amongst the four metaheuristic techniques.

6 Appendix 1

Table 4 Data for 2 area system

Generator *ij*	a_{ij}	b_{ij}	c_{ij}	P_{ij}^{min}	P_{ij}^{max}	Prohibited zones
	\$/h	\$/MWh	\$/(MW)^{2}h	MW	MW	MW
$G_{1,1}$	550	8.10	0.00028	100	500	[210 240] [350 380]
$G_{1,2}$	350	7.50	0.00056	50	200	[90 110] [140 160]
$G_{1,3}$	310	8.10	0.00056	50	150	[80 90] [110 120]
$G_{2,1}$	240	7.74	0.00324	80	300	[150 170] [210 240]
$G_{2,2}$	200	8.00	0.00254	50	200	[90 110] [140 150]
$G_{2,3}$	126	8.60	0.00284	50	120	[75 85] [100 105]

The transmission loss formula coefficients of two-area system are:

$$B_1 = \begin{bmatrix} 17 & 12 & 7 \\ 12 & 14 & 9 \\ 7 & 9 & 31 \end{bmatrix} \quad X10^{-6}$$

$$B_{01} = [-0.3908 \quad -0.1297 \quad 0.7047] \quad X10^{-3}$$

$$B_{001} = 0.045$$

$$B_2 = \begin{bmatrix} 24 & -6 & -8 \\ -6 & 129 & -2 \\ -8 & -2 & 150 \end{bmatrix} \quad X10^{-6}$$

$$B_{02} = [0.0591 \quad 0.2161 \quad -0.6635] \quad X10^{-3}$$

$$B_{002} = 0.056$$

7 Appendix 2

The transmission loss formula coefficients of three-area system are:

$$B_1 = \begin{bmatrix} 8.70 & 0.43 & -4.61 & 0.36 \\ 0.43 & 8.30 & -0.97 & 0.22 \\ -4.61 & -0.97 & 9.00 & -2.00 \\ 0.36 & 0.22 & -2.00 & 5.30 \end{bmatrix} \quad X10^{-5}$$

$$B_{01} = [-0.3908 \quad -0.1297 \quad 0.7047 \quad 0.0591] \quad X10^{-3}$$

$$B_{001} = 0.056$$

$$B_2 = \begin{bmatrix} 8.60 & -0.80 & 0.37 \\ -0.80 & 9.08 & -4.90 \\ 0.37 & -4.90 & 8.24 \end{bmatrix} \quad X10^{-5}$$

$$B_{02} = [0.2161 \quad -0.6635 \quad 0.5034] \quad X10^{-3}$$

$$B_{002} = 0.045$$

$$B_3 = \begin{bmatrix} 1.20 & -0.96 & 0.56 \\ -0.96 & 4.93 & -0.30 \\ 0.56 & -0.30 & 5.99 \end{bmatrix} \quad X10^{-5}$$

$$B_{03} = [-0.3216 \quad 0.4635 \quad 0.3503] \quad X10^{-3}$$

$$B_{003} = 0.055$$

Funding
There is NO Funding Information available for this manuscript.

Authors' contributions
JKP makes substantial contributions to conception, design and acquisition of data analysis and interpretation of data. JKP drafted the article and revising it thoroughly for preparation of the manuscript for the esteemed journal. Also he did the simulation part by using different test data for three different test systems. As a corresponding author he takes the primary responsibility for communication of the journal during the manuscript submission, peer review, publication process, and typically ensures that all the journal's administrative requirements, such as providing details of authorship. JKP will be available throughout the submission and peer review process to respond to editorial queries in a timely manner. Also he will be available after publication to respond to critiques of the work and cooperate with any requests from the journal for data or additional information should be answered about the paper arise after publication. JKP also agrees to be accountable for all aspects of the work in ensuring that questions related to the accuracy or integrity of any part of the work are appropriately investigated and resolved. MB participated in the peer review process of the manuscript and involved in the test data preparation. She reviewed the manuscript thoroughly. DPD participated in the peer review process of the manuscript and to compare the performance of the proposed method with that of other evolutionary methods. All authors read and approved the final manuscript.

Authors' information
Jagat Kishore Pattanaik received the Master's degree in Electrical Engineering in 2011 from Jadavpur University, Kolkata, India. He is currently working towards the Ph.D degree in the Department of Power Engineering, Jadavpur University. His research interest is Soft computing application in Power system Engineering.
Dr. Mousumi Basu received the Ph.D. degree from Jadavpur University, Kolkata, India. She is currently working as an Associate Professor in the Department of Power Engineering, Jadavpur University. His research interest is Power system Optimization and Soft computing technique.
Dr. Deba Prasad Dash received the Ph.D. degree from Jadavpur University, Kolkata, India. He is currently working as a Professor in the Department of Electrical Engineering, Orissa Engineering College, Bhubaneswar, Odisha, India. His research interests are Power system Stability, Power system protection & Soft computing technique.

Competing interests
We have declared that we have NO significant competing financial, professional or personal interests that might have influenced the performance or presentation of the work described in the manuscript.

Author details
[1]Department of Power Engineering, Jadavpur University, Salt Lake City, Kolkata 700098, India. [2]Electrical Engineering Department, Orissa Engineering College, Bhubaneswar, India.

Received: 3 July 2016 Accepted: 18 April 2017
Published online: 03 May 2017

References
1. Shoults, R. R., Chang, S. K., Helmick, S., & Grady, W. M. (1980). A practical approach to unit commitment, economic dispatch and savings allocation for multiple-area pool operation with import/export constraints. *IEEE Trans Power Apparatus Syst., 99*(2), 625–635.
2. Romano, R., Quintana, V. H., Lopez, R., & Valadez, V. (1981). Constrained economic dispatch of multi-area systems using the Dantzig–Wolfe decomposition principle. *IEEE Trans. Power Apparatus Syst., 100*(4), 2127–2137.
3. Desell, A. L., McClelland, E. C., Tammar, K., & Van Horne, P. R. (1984). Transmission constrained production cost analysis in power system planning. *IEEE Trans Power Apparatus Syst., 103*(8), 2192–2198.
4. Helmick, S. D., & Shoults, R. R. (1985). A practical approach to an interim multi-area economic dispatch using limited computer resources. *IEEE Trans Power Apparatus Syst., 104*(6), 1400–1404.
5. Wang, C., & Shahidehpour, S. M. (1992). A decomposition approach to non-linear multi area generation scheduling with tie-line constraints using expert systems. *IEEE Transactions on Power Systems, 7*(4), 1409–1418.
6. Streiffert, D. (1995). Multi-area economic dispatch with tie line constraints. *IEEE Transactions on Power Systems, 10*(4), 1946–1951.
7. Wernerus, J, Soder, L (1995) Area price based multi-area economic dispatch with tie line losses and constraints. In: IEEE/KTH Stockholm power tech conference (pp. 710–715). Sweden
8. Yalcinoz, T., & Short, M. J. (1998). Neural networks approach for solving economic dispatch problem with transmission capacity constraints. *IEEE Transactions on Power Systems, 13*(2), 307–313.
9. Jayabarathi, T., Sadasivam, G., & Ramachandran, V. (2000). Evolutionary programming based multi-area economic dispatch with tie line constraints. *Electric Machine and Power System, 28*, 1165–1176.
10. Chen, C. L., & Chen, N. (2001). Direct Search Method for solving Economic Dispatch Problem Considering Transmission Capacity Constraints. *IEEE Transactions on Power Systems, 16*(4), 764–769.
11. Chen, C. L. (2005). Optimal generation and reserve dispatch in a multi-area competitive market using a hybrid direct search method. *Energy Conversion and Management, 46*, 2856–2872.
12. Walter, D. C., & Sheble, G. B. (1993). Genetic algorithm solution of economic dispatch with valve point loading. *IEEE Transactions on Power Systems, 8*, 1325–1332.
13. Chiang, C.-L. (2005). Improved genetic algorithm for power economic dispatch of units with valve-point effects and multiple fuels. *IEEE Transactions on Power Systems, 20*(4), 1690–1699.
14. Storn, R., & Price, K. V. (1997). Differential evolution- a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization, 11*(4), 341–359.
15. Price, K. V., Storn, R., & Lampinen, J. (2005). *Differential Evolution: A Practical Approach to Global Optimization*. Berlin: Springer.
16. Yang, H. T., Yang, P. C., & Huang, C. L. (1996). Evolutionary Programming based economic dispatch for units with non-smooth fuel cost functions. *IEEE transactions on Power Systems, 11*(1), 112–118.
17. Goldberg, D. (1989). *Genetic Algorithms in Search, Optimization & Machine Learning*. Reading: Addison-Wesley Publishing Company, Inc.
18. Deb, K., & Agrawal, R. B. (1995). Simulated binary crossover for continuous search space. *Complex Systems, 9*(2), 115–148.
19. Kirkpatrick, S., Gelatt, C., & Vecchi, M. (1983). Optimization by simulated annealing. *Science, 22*, 671–680.
20. Wong, K. P., & Fung, C. C. (1993). Simulated annealing based economic dispatch algorithm. *IEE Proceedings Generation Transmission and Distribution, 140*(6), 509–515.
21. Gaing, Z.-L. (2003). Particle Swarm Optimization to Solving the Economic Dispatch Considering the Generator Constraints. *IEEE Transactions on Power Systems, 18*(3), 1187–1195.
22. Sinha, N., Chakrabarti, R., & Chattopadhyay, P. K. (2003). Evolutionary programming techniques for economic load dispatch. *IEEE Transactions on Evolutionary Computation, 7*(1), 83–94.