Available online at www.sciencedirect.com
ScienceDirect
Electrical Power and Energy Systems 30 (2008) 297–307
ELECTRICAL POWER & ENERGY SYSTEMS
www.elsevier.com/locate/ijepes

Solving economic load dispatch problems in power systems using chaotic and Gaussian particle swarm optimization approaches

Leandro dos Santos Coelho [a,*], Chu-Sheng Lee [b]

[a] Pontifical Catholic University of Paraná, PUCPR, Industrial and Systems Engineering Graduate Program, CCET/PPGEPS, Imaculada Conceição, 1155, Zip code 80215-901, Curitiba, Paraná, Brazil
[b] Department of Electrical Engineering, National Formosa University, 64, Wen-Hua Road, Huwei, Yunlin 632, Taiwan

Received 25 September 2006; received in revised form 6 August 2007; accepted 10 August 2007

Abstract

The objective of the Economic Dispatch Problems (EDPs) of electric power generation is to schedule the committed generating units outputs so as to meet the required load demand at minimum operating cost while satisfying all units and system equality and inequality constraints. Recently, global optimization approaches inspired by swarm intelligence and evolutionary computation approaches have proven to be a potential alternative for the optimization of difficult EDPs. Particle swarm optimization (PSO) is a population-based stochastic algorithm driven by the simulation of a social psychological metaphor instead of the survival of the fittest individual. Inspired by the swarm intelligence and probabilities theories, this work presents the use of combining of PSO, Gaussian probability distribution functions and/or chaotic sequences. In this context, this paper proposes improved PSO approaches for solving EDPs that takes into account nonlinear generator features such as ramp-rate limits and prohibited operating zones in the power system operation. The PSO and its variants are validated for two test systems consisting of 15 and 20 thermal generation units. The proposed combined method outperforms other modern metaheuristic optimization techniques reported in the recent literature in solving for the two constrained EDPs case studies.

Keywords: Economic dispatch problem; Electric power generation; Particle swarm optimization; Thermal generator constraints; Chaotic sequences

1. Introduction

The Economic Dispatch Problems (EDPs) is to determine the optimal combination of power outputs of all generating units to minimize the total fuel cost while satisfying the load demand and operational constraints [1].

In a liberalized electricity market, the optimization of economic dispatch is of economic value to the network operator. The economic dispatch is a relevant procedure in the operation of a power system. Over the past years, many optimization methods have been proposed in the literature. A spectrum of the advances in economic dispatch is well discussed in [2–28]. When compared with the conventional (classical) techniques [4–13], modern heuristic optimization techniques based on operational research and artificial intelligence concepts, such as evolutionary algorithms [14–19], simulated annealing [20,21], artificial neural networks [22–24], and taboo search [26,27] have been given attention by many researchers due to their ability to find an almost global optimal solution for EDPs with operating constraints.

EDPs have recently been solved by Particle Swarm Optimization (PSO) approaches [28–32]. The PSO originally developed by Eberhart and Kennedy in 1995 [33,34] is a population-based stochastic algorithm. Similarly to genetic algorithms [35], an evolutionary algorithm approach, the PSO is an evolutionary optimization tool of swarm intelligence field based on a swarm (population), where each member is seen as a particle, and each particle is a potential solution to the problem under analysis. Each particle in PSO has a randomized velocity associated to it, which moves through the space of the problem. However, unlike

* Corresponding author. Tel./fax: +55 413271 1345.
E-mail address: leandro.coelho@pucpr.br (L.S. Coelho).

0142-0615/$ - see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ijepes.2007.08.001

genetic algorithms, PSO does not have operators, such as crossover and mutation. PSO does not implement the survival of the fittest individuals; rather, it implements the simulation of social behavior [36]. PSO, however, allows each particle to maintain a memory of the best solution that it has found and the best solution found in the particle's neighborhood is swarm.

In PSO, a uniform probability distribution to generate random numbers into the velocity update equation is used. The use of other probability distributions may improve the ability to fine-tuning or even to escape from local optima. In the meantime, it has been proposed the use of the Gaussian [37–39], Cauchy [40], and exponential [41] probability distribution functions, and chaotic sequences [42] to generate random numbers to updating the velocity equation. All these approaches attempted to improve the performance of the standard PSO, but the amount of parameters of the algorithm to tune remained the same.

This paper proposes the Gaussian probability distribution and also chaotic sequences in PSO approaches to solve EDPs with 15 and 20 thermal units with generator constraints. Simulation results obtained through the PSO approaches are analyzed and compared with those reported in recent literature. The proposed PSO approaches of improvements in the setup of classical PSO algorithm using Gaussian and chaotic signals are powerful strategies to diversify the particle's swarm in PSO and improve the PSO's performance in preventing premature convergence to local minima.

The remaining sections of this paper are organized as follows: Section 2 describes the formulation of an EDP. Section 3 then describes the Gaussian and chaotic sequences for PSO approaches adopted here, while Section 4 details the procedure of constraint handling in PSO. Section 5 discusses the computational procedure and analyzes the PSO results when applied to case studies of EDPs with 15 and 20 thermal units. Lastly, Section 6 outlines our conclusions.

2. Formulation of an EDP with generator constraints

The EDP is to find the optimal combination of power generation that minimizes the total fuel cost while at thermal power units satisfying the total demand subjected to the operating constraints of a power system with a defined interval (typically 1 h). The essential operation constraints are the power balance constraint, where the total generated power must be equals to the load demands plus the transmission losses on the electrical network, and the power limit constraints, where individual generator units must be operated within their specified range.

In this context, for power balance, an equality constraint should be attempted. The generated power should be the same as the total load demand plus the total line losses. In this case, the active power balance is given by

$$\sum_{i=1}^{n} P_i - P_{\mathrm{L}} - P_{\mathrm{D}} = 0 \tag{1}$$

where P_i is the power of generator i (in MW); n is the number of generators in the system; P_{D} is the system's total demand (in MW); P_{L} represents the total line losses (in MW).

Inequality constraints for each generator must be also satisfied. Generation power of each generator should be laid between maximum and minimum limits. The inequality constraint for each generator is represented by Eq. (2) given by

$$P_i^{\min} \leqslant P_i \leqslant P_i^{\max} \tag{2}$$

where $P_i^{\min}$ and $P_i^{\max}$ are the output of the minimum and maximum operation of the generating unit i (in MW), respectively. The mathematical formulation of the total fuel cost function is formulated as follows:

$$\min f = \sum_{i=1}^{n} F_i(P_i) \tag{3}$$

where F_i is the total fuel cost for the ith generator (in \$/h). Generally, the fuel cost of thermal generating unit is represented in polynomial function,

$$F_i(P_i) = a_i P_i^2 + b_i P_i + c_i \tag{4}$$

where a_i, b_i and c_i are cost coefficients of generator i. However, the Eq. (4) can be modified using a sine function to model the ripples due to valve point effect of generator. Details about the valve point effect in generators can be found in [4,5,18,28].

In this study, the ramp-rate limits, prohibited operating zone-constraints, and transmission losses are considered [14,19,30,31,43]. The constraints of EDP at specific operating interval can be represented by Eqs. (5)–(8) given by

(i) Ramp-rate limit constraints:

$$\max(P_i^{\min}, P_i^0 - \mathrm{DR}_i) \leqslant P_i \leqslant \min(P_i^{\max}, P_i^0 + \mathrm{UR}_i) \tag{5}$$

where $P_i(t)$ is the present output power, P_i^0 is the previous output power, UR_i is the up-ramp limit of the ith generator (in units of MW/time-period), and DR_i is the down-ramp limit of the ith generator (in units of MW/time-period).

(ii) Prohibited operating zones constraints:

$$P_i \in \begin{cases} P_i^{\min} \leqslant P_i \leqslant P_{i,1}^{l} \\ P_{i,k-1}^{\mathrm{u}} \leqslant P_i \leqslant P_{i,k}^{l}, & k = 1, \ldots, zo_i \\ P_{i,z_i}^{\mathrm{u}} \leqslant P_i \leqslant P_i^{\max} \end{cases} \tag{6}$$

where $P_{i,k}^{\mathrm{l}}$ and $P_{i,k}^{\mathrm{u}}$ are the lower and upper bounds of the kth prohibited zone of unit i, respectively; k is the index of prohibited zones (zo_i).

(iii) Line flow constraint:

$$|P_{\mathrm{f},j}| \leqslant P_{\mathrm{f},j}^{\max}, \quad j = 1, \ldots, L \tag{7}$$

where $P_{\mathrm{f},j}$ is real power flow of line j and L is the number of transmission lines; and the transmission network losses, P_{L}, must be into account to achieve true economic load dispatch. Network loss is a function of generation unit. Clearly, calculation of transmission losses on the electrical network requires detailed network information, which adds complexity to EDPs [4,44,32]. In the methodology of constant loss formula coefficients (*loss coefficient method*) or B-coefficients, the network losses are expressed as a quadratic function of the generators power outputs that can be approximated in the form

$$P_{\mathrm{L}} = \sum_{i=1}^{n}\sum_{j=1}^{n} P_i B_{ij} P_j + \sum_{i=1}^{n} B_{i0} P_i + B_{00} \tag{8}$$

where B_{ij} is the ijth element of the loss coefficient square matrix, B_{i0} is the ith element of the loss coefficient vector, and B_{00} is the loss coefficient constant.

3. Optimization methodology based on PSO for the EDP

Social insect societies are distributed systems, which despite the simplicity of their individuals, present a highly structured social organization. As a result of this organization, insect societies can accomplish complex tasks that, in some cases, far exceed the individual capabilities of a single insect, as ants for example. The field of swarm intelligence is an emerging research area that presents features of self-organization and cooperation principles among group members bio-inspired on social insect societies. Swarm intelligence is inspired by nature, based on the fact that the live animals of a group contribute with their individual experiences to the group, rendering it stronger to face other groups. The most familiar representatives of swarm intelligence in optimization problems are the food-searching behavior of ant colonies [45], particle swarm optimization [46], artificial immune systems [47], and bacterial foraging [48].

The proposal of PSO algorithm was put forward by several scientists who developed bio-inspired computational simulations of the movement of organisms such as flocks of birds and schools of fish. Such simulations were heavily based on manipulating the distances between particles, i.e., the synchrony of the behavior of the swarm was seen as an effort to keep an optimal distance between them. In the next subsection, the fundamentals and implementation details about the PSO are described.

3.1. Fundamentals of PSO

In theory, at least, particles of a swarm may benefit from the prior discoveries and experiences of all the members of a swarm when foraging [49]. The fundamental point of developing PSO is a hypothesis in which the exchange of information among creatures of the same species offers some sort of evolutionary advantage [50]. Generally, the PSO is characterized as a simple heuristic of well-balanced mechanism with flexibility to enhance and adapt to both global and local exploration abilities. It is a stochastic search technique with reduced memory requirement, computationally effective and easier to implement compared to other metaheuristics of evolutionary computation and swarm intelligence fields.

Similarly to other population-based algorithms, PSO exploits a population of search points to probe the search space. Each individual in swarm, referred to as a 'particle', represents a potential solution. Each particle utilizes two important kinds of information in decision process. The first one is their own experience; that is, they have tried the choices and know which state has been better so far, and they know how good it was. The second one is other particle's experiences; that is, they have knowledge of how the other agents around them have performed.

Each particle in PSO keeps track of its coordinates in the problem space, which are associated with the best solution (best fitness) it has achieved so far. This value is called *pbest*. Another "best" value that is tracked by the global version of the particle swarm optimizer is the overall best value and its location obtained so far by any particle in the population. This location is called *gbest*.

Each particle moves its position in search domain and updates its velocity according to its own flying experience and neighbor's flying experience toward its *pbest* and *gbest* locations (global version of PSO). Acceleration is weighted by random terms, with separate random numbers being generated for acceleration toward *pbest* and *gbest* locations, respectively.

The basic elements of standard PSO are briefly stated and defined as follows:

- Particle $x_i(t)$, $i = 1, \ldots, n$: It is a potential solution represented by an n-dimensional vector, where n is the number of decision variables.
- Swarm: It is an apparently disorganized population of moving particles that tend to cluster together while each particle seems to be moving in a random direction.
- Individual best position $p_i(t)$, $i = 1, \ldots, n$: As a particle moves through the search space, it compares the fitness value at the current position to the best fitness value it has ever attained at any time up to the current time.
- Global best position, $p_g(t)$: It is the best position among all individual best positions achieved so far.
- Particle velocity $v_i(t)$, $i = 1, \ldots, n$: It is the velocity of the moving particles, which is represented by an n-dimensional vector. According to the individual best and global best positions, the particles velocity is updated. After obtaining the velocity updating, each particle position is changed to the next generation.

The procedure for implementing the global version (or star neighborhood topology) of PSO can be summarized by the following steps (see also the PSO flow chart in Fig. 1):

Step 1. *Initialization*: Initialize a swarm (population) of particles with random positions and velocities in the n dimensional problem space using a uniform probability distribution function.

Step 2. *Evaluation*: Evaluate the fitness value of each particle in swarm (population).

Step 3. *First comparison*: Compare each particle's fitness with the particle's *pbest*. If the current value is better than *pbest*, then set the *pbest* value equal to the current value and the *pbest* location equal to the current location in n-dimensional space.

Step 4. *Second comparison*: Compare the fitness with the population's overall previous best. If the current value is better than *gbest*, then reset *gbest* to the current particle's array index and value.

Step 5. *Updating*: Change the velocity and position of the particle according to Eqs. (9) and (10), respectively [33,34,40,41]:

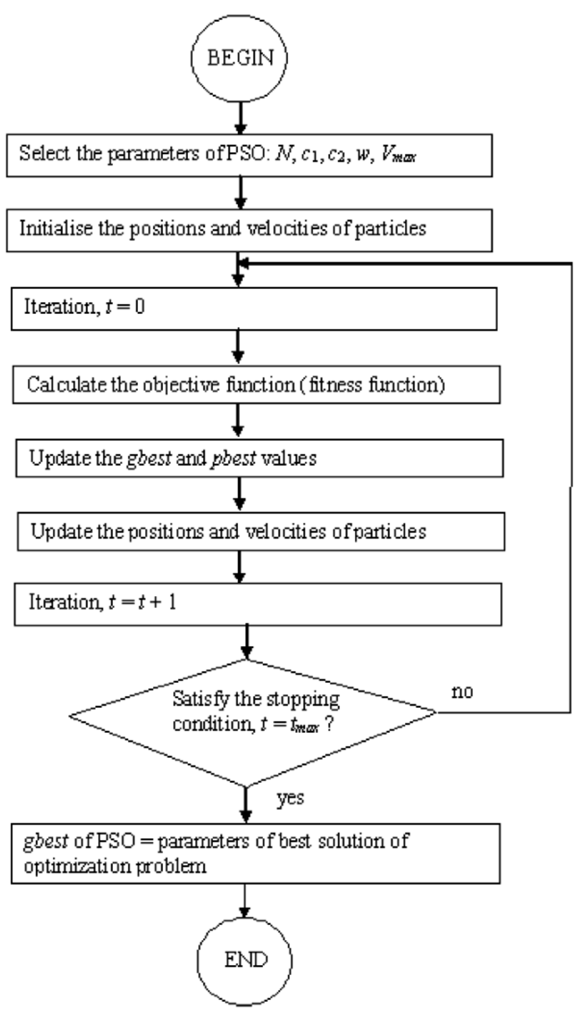

Fig. 1. Flow chart in a basic PSO approach.

$$v_i(t+1) = w(t) \cdot v_i(t) + c_1 \cdot ud \cdot [p_i(t) - x_i(t)] + c_2 \cdot Ud \cdot [p_g(t) - x_i(t)] \tag{9}$$

$$x_i(t+1) = x_i(t) + \Delta t \cdot v_i(t+1) \tag{10}$$

where $t = 1, 2, \ldots, t_{\max}$ indicates the iterations, $w(t)$ is the inertia weight; $v_i = [v_{i1}, v_{i2}, \ldots, v_{in}]^{\mathrm{T}}$ stands for the velocity of the ith particle, $x_i = [x_{i1}, x_{i2}, \ldots, x_{in}]^{\mathrm{T}}$ stands for the position of the ith particle, and $p_i = [p_{i1}, p_{i2}, \ldots, p_{in}]^{\mathrm{T}}$ represents the best previous position of the ith particle; positive constants c_1 and c_2 are the cognitive and social components that are the acceleration constants responsible for varying the particle velocity towards *pbest* and *gbest*, respectively. Index g represents the index of the best particle among all the particles in the swarm. Variables *ud* and *Ud* are two random functions based on uniform probability distribution functions in the range $[0, 1]$. Eq. (10) represents the update of particles positions, according to its previous position and its velocity, considering $\Delta t = 1$.

Step 6. *Stopping criterion*: Loop to step 2 until a stopping criterion is met, usually a sufficiently good fitness or a maximum number of iterations.

The use of variable w is responsible for dynamically adjusting the velocity of the particles, so it is responsible for balancing between local and global searches, hence requiring fewer iterations for the algorithm to converge. A low value of inertia weight implies a local search, while a high value leads to a global search.

Applying a large inertia weight at the start off the algorithm and making it decay to a small value through the PSO execution makes the algorithm search globally at the beginning of the search, and search locally at the end of the execution. The following weighting function $w(t)$ is used in (9):

$$w(t) = w_{\max} - \frac{w_{\max} - w_{\min}}{t_{\max}} t \tag{11}$$

Eq. (11) shows how the inertia weight is updated, considering $w_{\max}$ and $w_{\min}$ are the initial and final weights, respectively.

Particle velocity in each dimension are clamped to a maximum velocity $V_{\max}$. If the sum of accelerations causes the velocity in that dimension to exceed $V_{\max}$, which is a parameter specified by the user, then the velocity in that dimension is limited to $V_{\max}$. $V_{\max}$ is a parameter serving to determine the resolution with which the regions around the current solutions are searched. If $V_{\max}$ is too high, the PSO facilitates a global search, and particles might fly past good solutions. Conversely, if $V_{\max}$ is too small, the PSO facilitates a local search and particles may not explore sufficiently beyond locally good regions. Previous experience with PSO (trial and error, mostly) led us to set the $V_{\max}$ to 20% of the dynamic range of the variable in each dimension.

The first part in Eq. (9) is the momentum part of the particle. The inertia weight w represents the degree of the momentum of the particles. The second part is the 'cognition' part, which represents the independent thinking of the particle itself. The third part is the 'social' part, which represents the society behavior of the population.

In this work, new approaches to PSO are proposed in the next subsection. The aim is to modify the Eq. (9) of the conventional PSO (case 1) with *ud* and *Ud* based on uniform distribution to use it with Gaussian distribution and/or chaotic sequences in the range $[0,1]$.

3.2. New PSO approaches based on Gaussian distribution and/or chaotic sequences

Coelho and Krohling [40] proposed the use of a truncated Gaussian and Cauchy probability distribution to generate random numbers for the velocity updating equation of PSO. In this paper, new approaches to PSO are proposed which are based on Gaussian probability distribution linked with chaotic sequences. Firstly, random numbers are generated using the Gaussian probability distribution and/or chaotic sequences in the interval $[-1,1]$, and then mapped to the interval $[0,1]$. The use of chaotic sequences in PSO could be useful to escape from local optima, while the Gaussian distribution could provide a faster convergence in local search.

An essential feature of chaotic systems is that small changes in the parameters or the starting values for the data lead to vastly different future behaviors, such as stable fixed points, periodic oscillations, bifurcations, and ergodicity. These behaviors can be analyzed based on the meaning of Lyapunov exponents and the attractor theory [51,52].

Recently, the optimization techniques using chaotic sequences have received a great deal of attention in literature [53–59]. Chaotic optimization approaches are generally based on ergodicity, stochastic properties and irregularity of chaotic signals. This paper provides any approaches introducing chaotic mapping in PSO to improve the global convergence. The use of chaotic sequences in PSO can be helpful to escape more easily from local minima than can be done through the traditional PSO. In this work, the logistic map [51,52,60] for chaotic PSO approach was adopted. Thee logistic map is given by

$$y(k) = \mu \cdot y(k-1) \cdot [1 - y(k-1)] \tag{12}$$

where k is the sample, and μ is a control parameter, $0 \leqslant \mu \leqslant 4$. The behavior of the system of Eq. (14) is greatly changed with the variation of μ. The value of μ determines whether y stabilizes at a constant size, oscillates with limited bound, or behaves chaotically in an unpredictable pattern. Eq. (14) is deterministic, displaying chaotic dynamics when $\mu = 4$ and $y(1) \notin \{0, 0.25, 0.50, 0.75, 1\}$. In this case, $y(t)$ is distributed in the range $[0,1]$ provided the initial $y(1) \in [0,1]$ and $y(1) = 0.48$, as was adopted here. This work proposes new PSO approaches which combination with chaotic sequences based on logistic map, and Gaussian distribution. In this work, *Sd* and *sd* sequences using logistic map employed in PSO are equal for the results of Eq. (14) for $y(k)$ mapped to the interval $[0,1]$. The modification of the Eq. (9) (conventional PSO as *Type 1*) proceeds as following:

Type 2: It is used a function with Gaussian distribution, *gd*, to generate random numbers in the interval $[0,1]$ for the "cognitive part":

$$v_i(t+1) = w(t) \cdot v_i(t) + c_1 \cdot gd \cdot [p_i(t) - x_i(t)] + c_2 \cdot Ud \cdot [p_g(t) - x_i(t)] \tag{13}$$

Type 3: It is used a function with Gaussian distribution *Gd*, to generate random numbers in the interval $[0,1]$ for the "social part":

$$v_i(t+1) = w(t) \cdot v_i(t) + c_1 \cdot ud \cdot [p_i(t) - x_i(t)] + c_2 \cdot Gd \cdot [p_g(t) - x_i(t)] \tag{14}$$

Type 4: It is used a function with Gaussian distribution, *gd* and *Gd*, to generate random numbers in the interval $[0,1]$ for the "cognitive part" and "social part":

$$v_i(t+1) = w(t) \cdot v_i(t) + c_1 \cdot gd \cdot [p_i(t) - x_i(t)] + c_2 \cdot Gd \cdot [p_g(t) - x_i(t)] \tag{15}$$

Type 5: It is used a function to generate chaotic sequences, *Sd*, in the interval $[0,1]$ for the "social part":

$$v_i(t+1) = w(t) \cdot v_i(t) + c_1 \cdot ud \cdot [p_i(t) - x_i(t)] + c_2 \cdot Sd \cdot [p_g(t) - x_i(t)] \tag{16}$$

Type 6: It is used a function to generate chaotic sequences, *sd*, in the interval $[0,1]$ for the "cognitive part":

$$v_i(t+1) = w(t) \cdot v_i(t) + c_1 \cdot sd \cdot [p_i(t) - x_i(t)] + c_2 \cdot Ud \cdot [p_g(t) - x_i(t)] \tag{17}$$

Type 7: It is used a function to generate random numbers in the interval $[0,1]$ with Gaussian distribution, *gd*, for the "cognitive part" and the chaotic sequences, *Sd*, in the interval $[0,1]$ for the "social part":

$$v_i(t+1) = w(t) \cdot v_i(t) + c_1 \cdot gd \cdot [p_i(t) - x_i(t)] + c_2 \cdot Sd \cdot [p_g(t) - x_i(t)] \tag{18}$$

Type 8: It is used a function to generate random numbers in the interval $[0,1]$ with Gaussian distribution, *Gd*, for the "social part" and the chaotic sequences, *sd*, in the interval $[0,1]$ for the "cognitive part":

$$v_i(t+1) = w(t) \cdot v_i(t) + c_1 \cdot sd \cdot [p_i(t) - x_i(t)] + c_2 \cdot Gd \cdot [p_g(t) - x_i(t)] \tag{19}$$

4. Constraints handling with PSO approaches

A key factor in the application of PSO approaches to the optimization of an EDP is how the PSO algorithm handles the constraints relating to the problem.

Most optimization problems have constraints. The search space in constrained optimization problems consists of two kinds of points: feasible and unfeasible. Feasible points satisfy all the constraints, while unfeasible points violate at least one of them. Therefore, the solution or set of solutions obtained as the final result of an optimization method must necessarily be feasible, i.e., they must satisfy all constraints. The methods based on the use of penalty functions are usually employed to treat constrained optimization problems [61,62]. A constrained problem can be transformed into an unconstrained one by penalizing the constraints and building a single objective function, which in turn is minimized using an unconstrained optimization algorithm.

Over the last few decades, several methods have been proposed to handle constraints in optimization problems [63]. These methods can be grouped into four categories: methods that preserve the feasibility of solutions, penalty-based methods, methods that clearly distinguish between feasible and unfeasible solutions and hybrid methods.

When optimization algorithms are used for constrained optimization problems, it is common to handle constraints using concepts of penalty functions (which penalize unfeasible solutions), i.e., one attempt to solve an unconstrained problem in the search space S using a modified fitness function f (we are minimizing the fitness function in this paper) such as

$$\min f = \begin{cases} f(P_i), & \text{if } P_i \in F \\ f(P_i) + \text{penalty}(P_i), & \text{otherwise} \end{cases} \tag{20}$$

where penalty(P_i) is zero and no constraint is violated; otherwise it is positive. The penalty function is usually based on a distance measured to the nearest solution in the feasible region F or to the effort to repair the solution.

In this work, the methodology used to constraint handling in PSO approaches is divided into two steps. The first step involves finding solutions for the decision variables that lie within user-defined upper ($\lim_{\text{upper}}$) and lower ($\lim_{\text{lower}}$) bounds, that is, $x \in [\lim_{\text{lower}}, \lim_{\text{upper}}]$. Whenever a lower bound or an upper bound restriction fails to be satisfied, a repair rule is applied according to Eqs. (21) and (22), respectively:

$$P_i^j(t+1) = P_i^j(t) + \beta \cdot \text{rand}_i[0,1]\left\{\lim_{\text{upper}}(P_i^j(t)) - \lim_{\text{lower}}(P_i^j(t))\right\} \tag{21}$$

$$P_i^j(t+1) = P_i^j(t) - \beta \cdot \text{rand}_i[0,1]\left\{\lim_{\text{upper}}(P_i^j(t)) - \lim_{\text{lower}}(P_i^j(t))\right\} \tag{22}$$

where $\beta \in [0,1]$ is a user-defined parameter (β is set to 0.01 in this work); t is the current generation number, and rand$[0,1]$ is a uniformly distributed random value between 0 and 1.

In the second step, if the equality constraint of Eq. (1) and inequality constraints of Eqs. (5)–(7) are not solved, Eq. (3) is rewritten as

$$\min f = \sum_{i=1}^{n} F_i(P_i) + q_1\left(\sum_{i=1}^{n} P_i - P_L - P_D\right)^2 + q_2 \sum_{j=1}^{n} V_{k,j} \tag{23}$$

where q_1 and q_2 are positive constants (penalty factors) associate with the power balance and prohibited zones

Table 1
Data for the fifteen thermal units of generating unit capacity and coefficients

Unit	$P_i^{\min}$ (MW)	$P_i^{\max}$ (MW)	a (\$/MW^2)	b (\$/MW)	c (\$)
1	150	455	0.000299	10.1	671
2	150	455	0.000183	10.2	574
3	20	130	0.001126	8.8	374
4	20	130	0.001126	8.8	374
5	150	470	0.000205	10.4	461
6	135	460	0.000301	10.1	630
7	135	465	0.000364	9.8	548
8	60	300	0.000338	11.2	227
9	25	162	0.000807	11.2	173
10	25	160	0.001203	10.7	175
11	20	80	0.003586	10.2	186
12	20	80	0.005513	9.9	230
13	25	85	0.000371	13.1	225
14	15	55	0.001929	12.1	309
15	15	55	0.004447	12.4	323

Table 2
Data for the fifteen thermal units of ramp-rate limits and prohibited zones of the generating units

Unit	P_i^0	UR_i (MW/h)	DR_i (MW/h)	Prohibited zones		
				Zone 1	Zone 2	Zone 3
1	400	80	120			
2	300	80	120	[185 255]	[305 335]	[420 450]
3	105	130	130			
4	100	130	130			
5	90	80	120	[180 200]	[305 335]	[390 420]
6	400	80	120	[230 255]	[365 395]	[430 455]
7	350	80	120			
8	95	65	100			
9	105	60	100			
10	110	60	100			
11	60	80	80			
12	40	80	80	[30 40]	[55 65]	
13	30	80	80			
14	20	55	55			
15	20	55	55			

constraints, respectively. These penalty factors were tuned empirically and their values are $q_1 = 50$ and $q_2 = 1$ in the studied cases in this work. The V_j is expressed as follows:

$$V_{k,j} = \begin{cases} 1, & \text{if } P_{k,j} \text{ violates the prohibited zones} \\ 0, & \text{otherwise} \end{cases} \tag{24}$$

5. Simulation results

In this paper, to assess the efficiency of the proposed PSO approaches, two case studies (15 and 20 generators) of EDPs were applied in which the objective functions were ramp-rate limits, prohibited operating zones in the power system operation, and transmission losses are employed to demonstrate were taken into account.

$$B_{ij} = 10^{-3} \cdot \begin{bmatrix} 1.4 & 1.2 & 0.7 & -0.1 & -0.3 & -0.1 & -0.1 & -0.1 & -0.3 & -0.5 & -0.3 & -0.2 & 0.4 & 0.3 & -0.1 \\ 1.2 & 1.5 & 1.3 & 0.0 & -0.5 & -0.2 & 0.0 & 0.1 & -0.2 & -0.4 & -0.4 & 0.0 & 0.4 & 1.0 & -0.2 \\ 0.7 & 1.3 & 7.6 & -0.1 & -1.3 & -0.9 & -0.1 & 0.0 & -0.8 & -1.2 & -1.7 & 0.0 & -2.6 & 11.1 & -2.8 \\ -0.1 & 0.0 & -0.1 & 3.4 & -0.7 & -0.4 & 1.1 & 5.0 & 2.9 & 3.2 & -1.1 & 0.0 & 0.1 & 0.1 & -2.6 \\ -0.3 & -0.5 & -1.3 & -0.7 & 9.0 & 1.4 & -0.3 & -1.2 & -1.0 & -1.3 & 0.7 & -0.2 & -0.2 & -2.4 & -0.3 \\ -0.1 & -0.2 & -0.9 & -0.4 & 1.4 & 1.6 & 0.0 & -0.6 & -0.5 & -0.8 & 1.1 & -0.1 & -0.2 & -1.7 & 0.3 \\ -0.1 & 0.0 & -0.1 & 1.1 & -0.3 & 0.0 & 1.5 & 1.7 & 1.5 & 0.9 & -0.5 & 0.7 & 0.0 & -0.2 & -0.8 \\ -0.1 & 0.1 & 0.0 & 5.0 & -1.2 & -0.6 & 1.7 & 16.8 & 8.2 & 7.9 & -2.3 & -3.6 & 0.1 & 0.5 & -7.8 \\ -0.3 & -0.2 & -0.8 & 2.9 & -1.0 & -0.5 & 1.5 & 8.2 & 12.9 & 11.6 & -2.1 & -2.5 & 0.7 & -1.2 & -7.2 \\ -0.5 & -0.4 & -1.2 & 3.2 & -1.3 & -0.8 & 0.9 & 7.9 & 11.6 & 20.0 & -2.7 & -3.4 & 0.9 & -1.1 & -8.8 \\ -0.3 & -0.4 & -1.7 & -1.1 & 0.7 & 1.1 & -0.5 & -2.3 & -2.1 & -2.7 & 14.0 & 0.1 & 0.4 & -3.8 & 16.8 \\ -0.2 & 0.0 & 0.0 & 0.0 & -0.2 & -0.1 & 0.7 & -3.6 & -2.5 & -3.4 & 0.1 & 5.4 & -0.1 & -0.4 & 2.8 \\ 0.4 & 0.4 & -2.6 & 0.1 & -0.2 & -0.2 & 0.0 & 0.1 & 0.7 & 0.9 & 0.4 & -0.1 & 10.3 & -10.1 & 2.8 \\ 0.3 & 1.0 & 11.1 & 0.1 & -2.4 & -1.7 & -0.2 & 0.5 & -1.2 & -1.1 & -3.8 & -0.4 & -10.1 & 57.8 & -9.4 \\ -0.1 & -0.2 & -2.8 & -2.6 & -0.3 & 0.3 & -0.8 & -7.8 & -7.2 & -8.8 & 16.8 & 2.8 & 2.8 & -9.4 & 128.3 \end{bmatrix} \tag{25}$$

$$B_{i0} = 10^{-3} \cdot [-0.1 \quad -0.2 \quad 2.8 \quad -0.1 \quad 0.1 \quad -0.3 \quad -0.2 \quad -0.2 \quad 0.6 \quad 3.9 \quad -1.7 \quad 0.0 \quad -3.2 \quad 6.7 \quad -6.4] \tag{26}$$

$$B_{00} = 0.0055 \tag{27}$$

Each PSO approach was implemented in Matlab (MathWorks). All the programs were run on a 3.2 GHz Pentium IV processor with 2 GB of RAM (Random Access Memory).

In each case study, 50 independent runs were made for each of the optimization methods involving 50 different initial trial solutions for each optimization method. The setup of PSO approaches (star or *gbest* topology) used was the following: $c_1 = c_2 = 2.05$, $w_{max} = 1.1$, and $w_{min} = 0.8$. In these case studies, the stopping criterion t_{max} was 100 generations for the proposed PSO algorithms. The population size N was 50 and 30 to case studies I and II, respectively. All the B-coefficients are given in per unit (p.u.) on a 100 MVA base capacity.

5.1. Case study I: Fifteen-unit system

This case study consists of fifteen thermal units. All thermal units are within the ramp-rate limits and prohibited operating zones. The data shown in Tables 1 and 2 are also available in [19,30,31,43,64]. In this case, the load demand expected to be determined is $P_D = 2630$ MW. The B matrix of the transmission loss coefficient is given by

Table 3 shows the mean time, the minimum, mean and standard deviations, and the maximum cost achieved by the PSO approaches. As indicated in Table 3, the PSO(4) was the approach that obtained the minimum cost for the EDP of fifteen thermal units. However, the PSO(6)

Table 3
Convergence results (50 runs) of a case study I considering a fifteen-unit system with $P_D = 2630$ MW

Method	Minimum cost ($/h)	Mean cost ($/h)	Standard deviation of cost ($/h)	Maximum cost ($/h)	Mean CPU time (s)
PSO(1)	32775.68	35340.70	1942.84	38104.54	0.59
PSO(2)	32700.59	35090.89	1881.94	38023.80	0.64
PSO(3)	32656.97	35059.71	1853.71	*37899.13*	0.64
PSO(4)	*32508.12*	35122.79	1918.62	38044.42	0.69
PSO(5)	32833.18	34712.56	1848.07	38020.55	0.60
PSO(6)	32721.30	*32941.16*	*1513.57*	38040.89	0.60
PSO(7)	32831.48	34839.63	1901.38	37966.58	0.64
PSO(8)	32582.77	34396.46	1752.32	38031.86	0.58
Evolution strategy [19]	32568.54	32,620	–	32,710	0.48*
Particle swarm optimization [30]	33,858	33,039	–	33,331	26.59*
Genetic algorithm [30]	33,113	33,228	–	33,337	49.31*

* Values obtained from references.

obtained the best mean cost among the tested PSO techniques.

The best result obtained for solution vector P_i, $i = 1, \ldots, 15$ by PSO(4) with minimum cost of 32,508.12 $/h is given in Tables 3 and 4. Table 4 compares also the results obtained in this paper with those of other studies reported in the literature. Note that in the case study of fifteen thermal units, the results of the PSO(4) were comparatively lower than recent studies presented in the literature in [19,30,31].

5.2. Case study II: Twenty-unit system

This case study consists of twenty thermal units. This system supplies a total load demand of $P_D = 2500$ MW. The data shown in Table 5 are also available in [65]. In this second case, the B_{i0} and B_{00} present zero values. The B matrix of the transmission line loss coefficient is given by

$$B_{ij} = 10^{-3} \cdot \begin{bmatrix} 8.70 & 0.43 & -4.61 & 0.36 & 0.32 & -0.66 & 0.96 & -1.60 & 0.80 & -0.10 & 3.60 & 0.64 & 0.79 & 2.10 & 1.70 & 0.80 & -3.20 & 0.70 & 0.48 & -0.70 \\ 0.43 & 8.30 & -0.97 & 0.22 & 0.75 & -0.28 & 5.04 & 1.70 & 0.54 & 7.20 & -0.28 & 0.98 & -0.46 & 1.30 & 0.80 & -0.20 & 0.52 & -1.70 & 0.80 & 0.20 \\ -4.61 & -0.97 & 9.00 & -2.00 & 0.63 & 3.00 & 1.70 & -4.30 & 3.10 & -2.00 & 0.70 & -0.77 & 0.93 & 4.60 & -0.30 & 4.20 & 0.38 & 0.70 & -2.00 & 3.60 \\ 0.36 & 0.22 & -2.00 & 5.30 & 0.47 & 2.62 & -1.96 & 2.10 & 0.67 & 1.80 & -0.45 & 0.92 & 2.40 & 7.60 & -0.20 & 0.70 & -1.00 & 0.86 & 1.60 & 0.87 \\ 0.32 & 0.75 & 0.63 & 0.47 & 8.60 & -0.80 & 0.37 & 0.72 & -0.90 & 0.69 & 1.80 & 4.30 & -2.80 & -0.70 & 2.30 & 3.60 & 0.80 & 0.20 & -3.00 & 0.50 \\ -0.66 & -0.28 & 3.00 & 2.62 & -0.80 & 11.8 & -4.90 & 0.30 & 3.00 & -3.00 & 0.40 & 0.78 & 6.40 & 2.60 & -0.20 & 2.10 & -0.40 & 2.30 & 1.60 & -2.10 \\ 0.96 & 5.04 & 1.70 & -1.96 & 0.37 & -4.90 & 8.24 & -0.90 & 5.90 & -0.60 & 8.50 & -0.83 & 7.20 & 4.80 & -0.90 & -0.10 & 1.30 & 0.76 & 1.90 & 1.30 \\ -1.60 & 1.70 & -4.30 & 2.10 & 0.72 & 0.30 & -0.90 & 1.20 & -0.96 & 0.56 & 1.60 & 0.80 & -0.40 & 0.23 & 0.75 & -0.56 & 0.80 & -0.30 & 5.30 & 0.80 \\ 0.80 & 0.54 & 3.10 & 0.67 & -0.90 & 3.00 & 5.90 & -0.96 & 0.93 & -0.30 & 6.50 & 2.30 & 2.60 & 0.58 & -0.10 & 0.23 & -0.30 & 1.50 & 0.74 & 0.70 \\ -0.10 & 7.20 & -2.00 & 1.80 & 0.69 & -3.00 & -0.60 & 0.56 & -0.30 & 0.99 & -6.60 & 3.90 & 2.30 & -0.30 & 2.80 & -0.80 & 0.38 & 1.90 & 0.47 & -0.26 \\ 3.60 & -0.28 & 0.70 & -0.45 & 1.80 & 0.40 & 8.50 & 1.60 & 6.50 & -6.60 & 10.7 & 5.30 & -0.60 & 0.70 & 1.90 & -2.60 & 0.93 & -0.60 & 3.80 & -1.50 \\ 0.64 & 0.98 & -0.77 & 0.92 & 4.30 & 0.78 & -0.83 & 0.80 & 2.30 & 3.90 & 5.30 & 8.00 & 0.90 & 2.10 & -0.70 & 5.70 & 5.40 & 1.50 & 0.70 & 0.10 \\ 0.79 & -0.46 & 0.93 & 2.40 & -2.80 & 6.40 & 7.20 & -0.40 & 2.60 & 2.30 & -0.60 & 0.90 & 11.0 & 0.87 & -1.00 & 3.60 & 0.46 & -0.90 & 0.60 & 1.50 \\ 2.10 & 1.30 & 4.60 & 7.60 & -0.70 & 2.60 & 4.80 & 0.23 & 0.58 & -0.30 & 0.70 & 2.10 & 0.87 & 3.80 & 0.50 & -0.70 & 1.90 & 2.30 & -0.97 & 0.90 \\ 1.70 & 0.80 & -0.30 & -0.20 & 2.30 & -0.20 & -0.90 & 0.75 & -0.10 & 2.80 & 1.90 & -0.70 & -1.00 & 0.50 & 11.0 & 1.90 & -0.80 & 2.60 & 2.30 & -0.10 \\ 0.80 & -0.20 & 4.20 & 0.70 & 3.60 & 2.10 & -0.10 & -0.56 & 0.23 & -0.80 & -2.60 & 5.70 & 3.60 & -0.70 & 1.90 & 10.8 & 2.50 & -1.80 & 0.90 & -2.60 \\ -3.20 & 0.52 & 0.38 & -1.00 & 0.80 & -0.40 & 1.30 & 0.80 & -0.30 & 0.38 & 0.93 & 5.40 & 0.46 & 1.90 & -0.80 & 2.50 & 8.70 & 4.20 & -0.30 & 0.68 \\ 0.70 & -1.70 & 0.70 & 0.86 & 0.20 & 2.30 & 0.76 & -0.30 & 1.50 & 1.90 & -0.60 & 1.50 & -0.90 & 2.30 & 2.60 & -1.80 & 4.20 & 2.20 & 0.16 & -0.30 \\ 0.48 & 0.80 & -2.00 & 1.60 & -3.00 & 1.60 & 1.90 & 5.30 & 0.74 & 0.47 & 3.80 & 0.70 & 0.60 & -0.97 & 2.30 & 0.90 & -0.30 & 0.16 & 7.60 & 0.69 \\ -0.70 & 0.20 & 3.60 & 0.87 & 0.50 & -2.10 & 1.30 & 0.80 & 0.70 & -0.26 & -1.50 & 0.10 & 1.50 & 0.90 & -0.10 & -2.60 & 0.68 & -0.30 & 0.69 & 7.00 \end{bmatrix} \tag{28}$$

In this second case, the results of numerical simulation of tested PSO approaches are summarized in Table 6. From Table 6 it can see that the PSO approaches perform better

Table 4
Comparison of four methods: best result for the case study I

Unit power output (MW)	PSO(4) proposed	Evolution strategy [19]	Particle swarm optimization [30]	Genetic algorithm [30]
P_1	440.4990	455.00	439.12	415.31
P_2	179.5947	380.00	407.97	359.72
P_3	21.0524	130.00	119.63	104.42
P_4	87.1376	150.00	129.99	74.98
P_5	360.7675	168.92	151.07	380.28
P_6	395.8330	459.34	459.99	426.79
P_7	432.0085	430.00	425.56	341.32
P_8	168.9198	97.42	98.56	124.79
P_9	162.0000	30.61	113.49	133.14
P_{10}	138.4343	142.56	101.11	89.26
P_{11}	52.6294	80.00	33.91	60.06
P_{12}	66.8875	85.00	79.96	50.00
P_{13}	62.7471	15.00	25.00	38.77
P_{14}	47.5574	15.00	41.41	41.94
P_{15}	27.6065	15.00	35.61	22.64
Total power, (MW)	2643.6745	2653.85	2662.41	2668.44
P_L, (MW)	13.6745	23.85	32.42	38.28
Total cost, ($/h)	32508.12	32568.54	32858.00	33113.00

Table 5
Data for the twenty thermal units of generating unit capacity and coefficients

Unit	P_i^{min} (MW)	P_i^{max} (MW)	a ($/MW^2)	b ($/MW)	c ($)
1	150	600	0.00068	18.19	1000
2	50	200	0.00071	19.26	970
3	50	200	0.00650	19.80	600
4	50	200	0.00500	19.10	700
5	50	160	0.00738	18.10	420
6	20	100	0.00612	19.26	360
7	25	125	0.00790	17.14	490
8	50	150	0.00813	18.92	660
9	50	200	0.00522	18.27	765
10	30	150	0.00573	18.92	770
11	100	300	0.00480	16.69	800
12	150	500	0.00310	16.76	970
13	40	160	0.00850	17.36	900
14	20	130	0.00511	18.70	700
15	25	185	0.00398	18.70	450
16	20	80	0.07120	14.26	370
17	30	85	0.00890	19.14	480
18	30	120	0.00713	18.92	680
19	40	120	0.00622	18.47	700
20	30	100	0.00773	19.79	850

than the lambda-iteration and Hopfield neural network methods [65] in terms of quality of solution. As indicated in Table 6, the PSO(3) was the approach that obtained the minimum cost for the EDP of twenty thermal units with transmission losses. However, the PSO(6) obtained the best mean cost among the tested techniques. The best result obtained for solution vector P_i, $i = 1, \ldots, 20$ by PSO(3) with minimum cost of 59,804.05 $/h is given in Table 7.

It is clear from Table 7, the total power obtained by PSO(3) is closed to the constraint of $P_D = 2500$ MW. In this context, these PSO(3) approach performs better than the lambda-iteration and Hopfield neural network methods [65] in terms of the power loss.

Table 6
Convergence results (50 runs) of a case study II considering a twenty-unit system with $P_D = 2500$ MW

Method	Minimum cost ($/h)	Mean cost ($/h)	Standard deviation of cost ($/h)	Maximum cost ($/h)	Mean CPU time (s)
PSO(1)	60803.51	61223.95	507.56	62982.19	0.37
PSO(2)	59852.13	61142.17	437.21	62925.14	0.44
PSO(3)	*59804.05*	61171.84	532.44	63184.63	0.44
PSO(4)	60526.15	61123.36	330.89	62757.19	0.50
PSO(5)	60833.87	61220.93	516.21	62977.63	0.40
PSO(6)	60775.01	*61098.15*	*178.48*	*61573.73*	0.40
PSO(7)	60709.73	61106.08	252.83	62354.42	0.45
PSO(8)	60782.43	61101.92	299.08	62669.33	0.36
Lambda-iteration method [65]	62456.6391	–	–	–	0.033*
Hopfield neural network [65]	62456.6341	–	–	–	0.006*

* Values obtained from references.

Table 7
Comparison of three methods: best result for the case study II

Unit power output, (MW)	PSO(3) proposed	Lambda-iteration method [65]	Hopfield neural network [65]
P_1	563.3155	512.7805	512.7804
P_2	106.5639	169.1033	169.1035
P_3	98.7093	126.8898	126.8897
P_4	117.3171	102.8657	102.8656
P_5	67.0781	113.6836	113.6836
P_6	51.4702	73.5710	73.5709
P_7	47.7261	115.2878	115.2876
P_8	82.4271	116.3994	116.3994
P_9	52.0884	100.4062	100.4063
P_{10}	106.5097	106.0267	106.0267
P_{11}	197.9428	150.2394	150.2395
P_{12}	488.3315	292.7648	292.7647
P_{13}	99.9464	119.1154	119.1155
P_{14}	79.8941	30.8340	30.8342
P_{15}	101.525	115.8057	115.8056
P_{16}	25.8380	36.2545	36.2545
P_{17}	70.0153	66.8590	66.8590
P_{18}	53.9530	87.9720	87.9720
P_{19}	65.4271	100.8033	100.8033
P_{20}	36.2552	54.3050	54.3050
Total power, (MW)	2512.3343	2591.9670	2591.9669
P_L, (MW)	12.3343	91.9670	91.9669
Total cost, ($/h)	59804.05	62456.6391	62456.6341

6. Conclusion

This paper has demonstrated the feasibility of employing modified PSO approaches for efficient solving of EDPs with generator constraints. PSO is an effective optimization method that belongs to the category of evolutionary methods. Its development is based on the observations of social behavior of animals such as bird flocking, fish schooling, and swarm theory. Like evolutionary algorithms, PSO technique conducts search using a population of particles, corresponding to individuals. Each particle represents a candidate solution to the problem at hand.

In relation to the procedure involved in solving the EDP, the simulation results achieved by PSO(4) and PSO(3) to the case studies I and II, respectively, were better than the presented results in literature. The results of these simulations with modified PSO approaches are very encouraging and represent an important contribution to PSO algorithm setups.

In this paper, to enrich the searching behavior and to avoid being trapped into local optimum, a chaotic sequence based on logistic map is incorporated as a randomizer instead of traditional uniform random function into the PSO(5)–(8) approaches. The track of chaotic variable can travel ergodically over the whole search space. In general, the above chaotic variable has special characters, i.e. ergodicity, pseudo-randomness and irregularity. The chaotic PSO(5)–(8) approaches exhibit slightly better performance in terms of mean solutions (in 50 runs) when compared to the PSO(1) in two case studies, due to its ability to achieve sustainable development keeping the diversity of particles.

Methods combining PSO with Gaussian and chaotic signals can be very effective in solving EDPs. In future, we will focus mainly on the conception of PSO approaches incorporating local search with Cauchy and exponential probability density functions for the solution of EDPs taking generator constraints into account.

Acknowledgments

This work was supported by the National Council of Scientific and Technologic Development of Brazil—CNPq—under Grant 309646/2006-5/PQ. The valuable comments and suggestions of the reviewers are also acknowledged. They helped the authors to improve the presentation of the paper.

References

[1] Liu D, Cai Y. Taguchi method for solving the economic dispatch problem with nonsmooth cost functions. IEEE Trans Power Syst 2005;20(4):2006–14.
[2] Chowdhury BH, Rahman S. A review of recent advances in economic dispatch. IEEE Trans Power Syst 1990;5(4):1248–57.
[3] Happ HH. Optimal power dispatch – a comprehensive survey. IEEE Trans Power Appar Syst 1977;96(3):841–54.
[4] Wood AJ, Wollenberg BF. Power generation, operation and control. New York: John Wiley & Sons; 1994.
[5] Sadaat H. Power system analysis. WCB/McGraw-Hill; 1999.
[6] Ross DW, Kim S. Dynamic economic dispatch of generation. IEEE Trans Power Appar Syst 1980;99(6):2060–8.
[7] Liang ZX, Glover JD. A zoom feature for a programming solution to economic dispatch including transmission losses. IEEE Trans Power Syst 1992;7(3):544–50.
[8] Fan JY, Zhang L. Real-time economic dispatch with line flow and emission constraints using quadratic programming. IEEE Trans Power Syst 1998;13(2):320–5.
[9] Chen CL, Wang CL. Branch-and-bound scheduling for thermal generating units. IEEE Trans Energy Conver 1993;8(2):184–9.
[10] Rabin AJ, Coonick AH, Coory BJ. A homogeneous linear programming algorithm for the security constrained economic dispatch problem. IEEE Trans Power Syst 2000;15(3):930–6.
[11] Lin CE, Viviani GL. Hierarchical economic dispatch for piecewise quadratic cost functions. IEEE Trans Power Appar Syst 1984;103(6):1170–5.
[12] Chen SD, Chen JF. A direct Newton–Raphson economic emission dispatch. Electr Power Energy Syst 2003;25:411–7.
[13] Yang HT, Chen SL. Incorporating a multi-criteria decision procedure into the combined dynamic programming/production simulation algorithm for generation expansion planning. IEEE Trans Power Syst 1989;4(1):165–75.
[14] Walters DC, Sheblé GB. Genetic algorithm solution of economic dispatch with valve point loading. IEEE Trans Power Syst 1993;8(3):1325–32.
[15] Kim JO, Shin DJ, Park JN, Singh C. Atavistic genetic algorithm for economic dispatch with valve point effect. Elect Power Syst Res 2002;62:201–7.
[16] Li F, Morgan R, Williams D. Hybrid genetic approaches to ramping rate constrained dynamic economic dispatch. Elect Power Syst Res 1997;43(2):97–103.
[17] Jayabarathi T, Sadasivam G, Ramachandran V. Evolutionary programming based economic dispatch of generators with prohibited operating zones. Elect Power Syst Res 1999;52(3):261–6.
[18] Sinha N, Chakrabarti R, Chattopadhyay PK. Evolutionary programming techniques for economic load dispatch. IEEE Trans Evol Comput 2003;7(1):83–94.
[19] Pereira-Neto A, Unsihuay C, Saavedra OR. Efficient evolutionary strategy optimisation procedure to solve the nonconvex economic dispatch problem with generator constraints. IEEE Proc Gener Transm Distrib 2005;152(5):653–60.
[20] Wong KP, Wong YW. Genetic and genetic/simulated-annealing approaches to economic dispatch. IEEE Proc Gener Transm Distrib 1994;141(5):507–13.
[21] Roa-Sepulveda CA, Pavez-Lazo BJ. A solution to the optimal power flow using simulated annealing. Electr Power Energy Syst 2003;25(1):47–57.
[22] Kumar J, Seblé GB. Clamped state solution of artificial neural network for real-time economic dispatch. IEEE Trans Power Syst 1995;10(2):925–31.
[23] Yalcinoz T, Short MJ. Neural network approach for solving economic dispatch problem with transmission capacity constraints. IEEE Trans Power Syst 1998;13(2):307–13.
[24] Da Silva IN, Nepomuceno L, Bastos TM. An efficient Hopfield network to solve economic dispatch problems with transmission system representation. Electr Power Energy Syst 2004;26(9):733–8.
[25] Aravindhababu P, Nayar KR. Economic dispatch based on optimal lambda using radial basis function network. Electr Power Energy Syst 2002;24(7):551–6.
[26] Lin WM, Chen FS, Tsay MT. An improved tabu search for economic dispatch with multiple minima. IEEE Trans Power Syst 2002;17(1):108–12.
[27] Khamsawang S, Boonseng C, Pothiya S. Solving the economic dispatch problem with tabu search algorithm. In: IEEE International

Conference on Industrial Technology 2002, Bangkok, Thailand. p. 274–8.
[28] Park JB, Lee KS, Shin JR, Lee KY. A particle swarm optimization for economic dispatch with nonsmooth cost function. IEEE Trans Power Syst 2005;20(1):34–42.
[29] Victoire TAA, Jeyakumar AE. Hybrid PSO-SQP for economic dispatch with valve-point effect. Elect Power Syst Res 2004;71(1):51–9.
[30] Gaing ZL. Particle swarm optimization to solving the economic dispatch considering the generator constraints. IEEE Trans Power Syst 2003;18(3):1187–95.
[31] Gaing ZL. Closure to Discussion of 'Particle swarm optimization to solving the economic dispatch considering the generator constraints'. IEEE Trans Power Syst 2004;19(4):2122–3.
[32] Moustafa YG, Mekhamer SF, Moustafa YG, El-Sherif N, Mansour MM. A modified particle swarm optimizer applied to the solution of the economic dispatch problem. In: International conference on electrical, electronic, and computer engineering 2004, ICEEC, Cairo, Egypt. p. 724–31.
[33] Eberhart RC, Kennedy, JF. A new optimizer using particle swarm theory. In: Proceedings of the 6th international symposium on micro machine and human science, Nagoya, Japan; 1995. p. 39–43.
[34] Kennedy JF, Eberhart RC. Particle swarm optimization. In: Proceedings of the IEEE international conference on neural networks, vol. IV, Perth, Australia 1995, p. 1942–8.
[35] Goldberg DE. Genetic algorithms in search, optimization, and machine learning. Reading (MA): Addison Wesley; 1989.
[36] Coelho LS, Mariani VC. Use of cultural particle swarm optimization for Loney's solenoids design. In: Proceedings of 12th Biennial IEEE conference on electromagnetic field computation, Miami, FL, USA; 2006.
[37] Krohling RA. Gaussian swarm: a novel particle swarm optimization algorithm. In: Proceedings of the IEEE conference on cybernetics and intelligent systems (CIS), Singapore 2004, p. 372–6.
[38] Secrest BR, Lamont GB. Visualizing particle swarm optimization – Gaussian particle swarm optimization. In: Proceedings of the IEEE swarm intelligence symposium, Indianapolis, IN, USA; 2003. p. 198–204.
[39] Higashi N, Iba H. Particle swarm optimization with Gaussian mutation. In: Proceedings of the IEEE swarm intelligence symposium, Indianapolis, IN, USA; 2003. p. 72–9.
[40] Coelho LS, Krohling RA. Predictive controller tuning using modified particle swarm optimisation based on Cauchy and Gaussian distributions. In: Soft Computing: Methodologies and Applications, Springer Engineering series in Advances in Soft Computing; 2005. p. 287–98.
[41] Krohling, RA, Coelho LS. PSO-E: Particle swarm with exponential distribution IEEE World Congress on Computational Intelligence. In: Proceedings of IEEE Congress on Evolutionary Computation, Vancouver, Canada; 2006. p. 5577–82.
[42] Coelho LS, Herrera BM. Fuzzy modeling using chaotic particle swarm approaches applied to a yo-yo motion system. IEEE Trans Indust Electron 2007.
[43] Lee FN. Breipohl AM Reserve constrained economic dispatch with prohibited operating zones. IEEE Trans Power Syst 1993;8(1):246–54.
[44] Gross CA. Power system analysis. 2nd ed. New York, (NY, USA): Wiley; 1986.
[45] Dorigo M, Stützle T. Ant colony optimization. Cambridge (MA): MIT; 2004 [A Bradford Book].
[46] Kennedy JF, Eberhart RC, Shi RC. Swarm intelligence. San Francisco (CA, USA): Morgan Kaufmann Pub.; 2001.
[47] Castro LN, Timmis JI. Artificial immune systems: a new computational intelligence approach. London (UK): Springer-Verlag; 2002.
[48] Passino KM. Biomimicry of bacterial foraging for distributed optimization and control. IEEE Contr Syst 2002;22(3):52–67.
[49] Wilson EO. Sociobiology: the new synthesis. Cambridge (MA): Belknap Press; 1975.
[50] Ratnaweera A, Halgamuge SK, Watson HC. Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients. IEEE Trans Evol Comput 2004;8(3):240–55.
[51] Strogatz SH. Nonlinear dynamics and chaos. Massachussetts: Perseus Publishing; 2000.
[52] Parker TS, Chua LO. Practical numerical algorithms for chaotic system. Berlin (Germany): Springer-Verlag; 1989.
[53] Li B, Jiang W. Optimizing complex functions by chaos search. Cybern Syst 1998;29(4):409–19.
[54] Jiang C, Ma Y, Wang C. PID controller parameters optimization of hydro-turbine governing systems using deterministic-chaotic-mutation evolutionary programming (DCMEP). Energy Conv Manage, 47(9–10):1222–30.
[55] Li L, Yang Y, Peng H, Wang X. Parameters identification of chaotic systems via chaotic ant swarm. Chaos Solitons Fractals 2006;28(15):82–90.
[56] Wang L, Smith K. On chaotic simulated annealing. IEEE Trans Neural Networks 1998;9(4):716–8.
[57] Liao GC, Tsao TP. Application embedded chaos search immune genetic algorithm for short-term unit commitment. Elect Power Syst Res 2004;71(2):135–44.
[58] Zilong G, Sun'an W, Jian Z. A novel immune evolutionary algorithm incorporating chaos optimization. Pattern Recogn Lett 2006;27(1):2–8.
[59] Mingjun J, Huanwen T. Application of chaos in simulated annealing. Chaos Solitons Fractals 2004;21(4):933–41.
[60] Shengsong L, Min W, Zhijian H. Hybrid algorithm of chaos optimisation and SLP for optimal power flow problems with multimodal characteristic. IEEE Proc Gener Transm Distrib 2003;150(5):543–7.
[61] Arora JS. Introduction to optimum design. New York (NY, USA): MGraw-Hill; 1989.
[62] Rao SS. Engineering optimization. 3rd ed. Hoboken (NJ, USA): John Wiley & Sons; 1996.
[63] Michalewicz Z, Schoenauer M. Evolutionary algorithms for constrained parameter optimization problems. Evol Comput 1996;4(1):1–32.
[64] Gaing ZL. Closure to Discussion of Particle swarm optimization to solving the economic dispatch considering the generator constraints. IEEE Trans Power Syst 2004;19(4):2122–3.
[65] Su CT, Lin CT. New approach with a Hopfield modeling framework to economic dispatch. IEEE Trans Power Syst 2000;15(2):541–5.