Pseudo-Gradient Based Particle Swarm Optimization Method for Nonconvex Economic Dispatch

Vo Ngoc Dieu, Peter Schegner and Weerakorn Ongsakul

Abstract This chapter proposes a pseudo-gradient based particle swarm optimization (PGPSO) method for solving nonconvex economic dispatch (ED) including valve point effects, multiple fuels, and prohibited operating zones. The proposed PGPSO is based on the self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients (HPSO-TVAC) with position of particles guided by a pseudo-gradient. The pseudo-gradient here is to determine an appropriate direction for the particles during their movement so that they can quickly move to an optimal solution. The proposed method has been tested on several systems and the obtained results are compared to those from many other methods available in the literature. The test results have indicated that the proposed method can obtain less expensive total costs than many others in a faster computing manner, especially for the large-scale systems. Therefore, the proposed PGPSO is favorable for online implementation in the practical ED problems.

Keywords Economic dispatch · Multiple fuel options · Particle swarm optimization · Prohibited zones · Pseudo gradient · Valve point effects

V. N. Dieu (✉)
Department of Power Systems, Ho Chi Minh City University of Technology,
Ho Chi Minh, Vietnam
e-mail: vndieu@gmail.com

P. Schegner
Institute of Electrical Power Systems and High Voltage Engineering,
Technische Universität Dresden, 01069 Dresden, Germany
e-mail: schegner@ieeh.et.tu-dresden.de

W. Ongsakul
Energy Field of Study, Asian Institute of Technology,
Pathumthani 12120, Thailand
e-mail: ongsakul@ait.asia

I. Zelinka et al. (eds.), *Power, Control and Optimization*, Lecture Notes in Electrical Engineering 239, DOI: 10.1007/978-3-319-00206-4_1,

Nomenclature

a_i, b_i, c_i	fuel cost coefficients of unit i
e_i, f_i	fuel cost coefficients of unit i reflecting valve-point effects
a_{ij}, b_{ij}, c_{ij}	fuel cost coefficients for fuel type j of unit i
e_{ij}, f_{ij}	fuel cost coefficients for fuel type j of unit i reflecting valve-point effects
B_{ij}, B_{0i}, B_{00}	B-matrix coefficients for transmission power loss
c_{1i}, c_{1f}	initial and final values of cognitive acceleration factor, respectively
c_{2i}, c_{2f}	initial and final values of social acceleration factor, respectively
c_1, c_2	cognitive and social acceleration coefficients, respectively
DR_i	ramp down rate limit of unit i
$g_p\left(x_{id}^{(k)}\right)$	pseudo-gradient at point k for particle d of element i
N	total number of generating units
n_i	number of prohibited operating zones of unit i
N_p	number of particles
P_D	total system load demand
P_i	power output of unit i
$P_{i,\text{high}}$	highest possible power output of generator i
$P_{i,\text{low}}$	lowest possible power output of generator i
$P_{i,\max}$	maximum power output of generator i
$P_{i,\min}$	minimum power output of generator i
$P_{ij,\min}$	minimum power output for fuel j of generator i
P_L	total transmission loss
P_{ik}^{l}	lower bound for prohibited zone k of generator i
P_{ik}^{u}	upper bound for prohibited zone k of generator i
UR_i	ramp up rate limit of unit i
v_{id}	velocity of particle d for element i
x_{id}	position of particle d for element i
Ω	set of units with prohibited operating zones
$\delta(x_{id})$	direction indicator for position of element i in particle d

1 Introduction

The economic dispatch (ED) is used to determine the economical real power output of the online units so as their total production cost is minimized while satisfying the unit and system operating constraints [1, 2]. For mathematical convenience, the objective function of the ED problem was approximated by a single quadratic function [3], which is differentiable. Nevertheless, the input-output characteristics of thermal generating units are more complicated due to the effects of valve point loadings [4], multiple fuels [5], or prohibited zones [6].

Therefore, the practical ED problem should be formulated as a non-convex objective function subject to complex constraints, which cannot be directly solved by the mathematical programming techniques. Hence, more advanced techniques have to be developed to deal with the optimization problem with multiple minima.

Several conventional methods have been applied for solving ED problems such as gradient search, Newton's method, dynamic programming (DP) [3], hierarchical approach based on the numerical method (HNUM) [5], decomposition method (DM) [6], lambda iteration method [7], Maclaurin series-based Lagrangian (MSL) method [8], and novel direct search (NDS) [9]. Among these methods, the MSL and NDS methods can directly deal with a non-convex ED problem with non-differentiable objective. Although these methods can quickly find a solution for the problem, the obtained results are still the local optimum solution, especially for the large-scale systems. In general, the conventional methods are not effective for implementation on the ED problems with non-differentiable objective function. Recently, many methods based on artificial intelligence have been developed for solving ED problems such as Hopfield neural network (HNN) [10, 11], genetic algorithm (GA) [12], evolutionary programming (EP) [13], evolutionary algorithm (EA) [14, 15], differential evolution (DE) [16], artificial bee colony (ABC) algorithm [17], artificial immune system (AIS) [18], biogeography-based optimization (BBO) [19], and particle swarm optimization (PSO) [20–27]. Among of them, the HNN method which is based on the minimization of its energy function can be only applied to the convex optimization problems with differentiable objective function. Although this method can be implemented on large-scale problems, it suffers many drawbacks such as large number of iterations, linear constraint requirement, and local optimum solution. The others are the meta-heuristic search methods which are based on a population for searching an optimal solution for the problems. These search methods can overcome the drawbacks of the HNN and conventional methods due to their ability to find near optimal solution for non-convex optimization problems. However, for the large-scale and non-smooth problems with multiple minima these methods may suffer low solution quality and long computational time. Of all the meta-heuristic search methods, PSO is the most popular method implemented for solving different ED problems due to its powerful search ability, especially for nonconvex problems [21]. Several improvements for PSO method have been proposed for solving nonconvex ED problems such as quantum-inspired PSO (QPSO) [22], self-organizing hierarchical PSO (SOH_PSO) [23], modified PSO (MPSO) [24, 25], PSO with modified stochastic acceleration factors (PSO-MSAF) [26], new PSO with local random search (NPSO-LRS) [27], and simulated annealing like PSO (SA-PSO) [28]. These improved PSO-based methods can obtain higher solution quality than the conventional PSO method for complicated nonconvex optimization problems. In addition to the single methods, hybrid methods have been also developed for dealing with the nonconvex ED problems such as hybrid technique integrating the uniform design with the genetic algorithm (UHGA) [29], combining of chaotic differential evolution and quadratic programming (DEC-SQP) [30, 31], self-tuning hybrid differential evolution (self-tuning HDE) [32], and fuzzy adaptive particle swarm optimization algorithm with

Nelder–Mead simplex search (FAPSO-NM) [33]. These hybrid methods utilize the advantages of each element method to enhance their search ability for the complex problems. Consequently, they become powerful search methods for obtaining higher solution quality than the element methods. However, these hybrid methods may be more complicated and slower than the element methods due to combination of several operations.

In this chapter, a pseudo-gradient based particle swarm optimization (PGPSO) method is proposed for solving nonconvex ED problems considering valve point effects, multiple fuels, and prohibited operating zones. The proposed PGPSO is based on the self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients (HPSO-TVAC) [34] with position of particles guided by a pseudo-gradient [35]. The pseudo-gradient here is to determine an appropriate direction for the particles during their movement so that they can quickly move to an optimal solution. The proposed method has been tested on several systems and the obtained results are compared to those from many other methods available in the literature.

The remaining organization the chapter is as follows. The formulation of nonconvex ED problems with valve point loading effects, multiple fuels, and prohibited operating zones are presented in Sect. 2. The PGPSO method is addressed in Sect. 3. The implementation of the PGPSO method to the nonconvex ED problems is described in Sect. 4. The numerical results are followed in Sect. 5. Finally, the conclusion is given.

2 Problem Formulation

The objective of an ED problem is to minimize the total cost of thermal generating units of a system over some appropriate period (1 h typically) while satisfying various constraints including system and unit operating constraints. Mathematically, the nonconvex ED problems are formulated as follows.

2.1 ED Problem with Valve Point Effects

The ED problem with valve point effects (VPE) is a non-smooth and non-convex problem with multiple minima due to taking into consideration of ripples in the heat-rate curve of boilers. The model of valve point loading effects has been proposed in Walter and Sheble [36] by introducing a sinusoidal function added to the quadratic fuel cost function. The objective of the problem is written as:

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

where the fuel cost function of unit i with VPE is represented by Sinha et al. [13]:

$$F_i(P_i) = a_i + b_i P_i + c_i P_i^2 + \left| e_i \times \sin(f_i \times (P_{i,\min} - P_i)) \right| \tag{2}$$

subject to

1. *Real power balance*: The total real power output of generating units satisfies total real load demand plus power loss:

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

where the power loss P_L can be approximately calculated by Kron's formula [3]:

$$P_L = \sum_{i=1}^{N} \sum_{j=1}^{N} P_i B_{ij} P_j + \sum_{i=1}^{N} B_{0i} P_i + B_{00} \tag{4}$$

2. *Generator capacity limits*: The real power output of generating units should be within between their upper and lower bounds by:

$$P_{i,\min} \le P_i \le P_{i,\max} \tag{5}$$

2.2 ED Problem with Multiple Fuels

In the ED problem with multiple fuels (MF), the piecewise quadratic function is used to represent the multiple fuels which are available for each generating unit [5]. The fuel cost function of unit i is defined as [10]:

$$F_i(P_i) = \begin{cases} a_{i1} + b_{i1} P_i + c_{i1} P_i^2, \text{ fuel } 1, \ P_{i,\min} \le P_i \le P_{i1} \\ a_{i2} + b_{i2} P_i + c_{i2} P_i^2, \text{ fuel } 2, \ P_{i1} < P_i \le P_{i2} \\ \cdots \\ a_{ij} + b_{ij} P_i + c_{ij} P_i^2, \text{ fuel } j, \ P_{ij-1} < P_i \le P_{i,\max} \end{cases} \tag{6}$$

For generator i with j fuel options in (6), its cost curve is divided into j discrete segments between lower limit $P_{i,\min}$ and upper limit $P_{i,\max}$, in which each fuel type is represented by a quadratic function with lower power output limit P_{ij-1} and upper power output limit P_{ij}.

The objective of the ED problem with multiple fuels is to minimize total generator cost (1) with fuel cost function in (6) subject to the real power balance constraint (3) and generator capacity limits (5).

2.3 ED Problem with Both Valve Point Effects and Multiple Fuels

Thermal generating units can be supplied with multiple fuel sources and their boilers have also valve points for controlling their power outputs [12]. The fuel cost function of generating unit i is represented as:

$$F_i(P_i) = \begin{cases} F_{i1}(P_i), \text{ fuel } 1, \ P_{i,\min} \le P_i \le P_{i1} \\ F_{i2}(P_i), \text{ fuel } 2, \ P_{i1} < P_i \le P_{i2} \\ \cdots \\ F_{ij}(P_i), \text{ fuel } j, \ P_{ij-1} < P_i \le P_{i,\max} \end{cases} \tag{7}$$

where the fuel cost function for fuel type j of unit i is determined by:

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

The objective of the ED problem with both valve point effects and multiple fuels is to minimize total generator cost (1) with fuel cost function in (7) subject to the real power balance constraint (3) and generator capacity limits (5).

2.4 ED Problem with Prohibited Operating Zones

In some practical cases, thermal generating units may have prohibited operating zones (POZ) due to physical constraints on components of units. Consequently, the whole operating region of a generating unit with prohibited operating zones will be broken into several isolated feasible sub-regions [20].

The fuel cost function for each unit in the ED problem with POZ is a quadratic function as in (2) neglecting the sinusoidal term and the equality and inequality constraints for this problem include the real power balance constraint (3), generator capacity limits (5) for units having no POZ, and

1. *Prohibited operating zones*: For generating units with POZ, their entire feasible operating zones are decomposed in feasible sub-regions and their feasible operating points should be in one of the sub-regions as follows:

$$P_i \in \begin{cases} P_{i,\min} \le P_i \le P_{i1}^l \\ P_{ik-1}^u \le P_i \le P_{ik}^l,\ k = 2, \ldots, n_i \\ P_{in_i}^u \le P_i \le P_{i,\max} \end{cases} ; \forall i \in \Omega \tag{9}$$

2. *Ramp rate constraints:* The increased or decreased power output of a unit from its initial operating point to the next one should not exceed its ramp up and down rate limits. The ramp rate constraints are determined by:

$$P_i - P_{i0} \le UR_i, \text{ if generation increases} \tag{10}$$

$$P_{i0} - P_i \le DR_i, \text{ if generation decreases} \tag{11}$$

3 Pseudo-Gradient Based Particle Swarm Optimization

3.1 Self-organizing Hierarchical Particle Swarm Optimizer

PSO has become one of the most popular methods applied in various optimization problems due to its simplicity and capability to find near optimal solution, especially for complicated and non-convex problems [37]. Several improvements have been made to enhance the search ability of PSO to deal with more complicated and larger-scale problems [21]. In this chapter, the new improvement is based on the self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients (HPSO-TVAC) [34]. For the implementation of the HPSO-TVAC on a n-dimension optimization problem, the position and velocity vectors of particle d are represented by $x_d = [x_{1d}, x_{2d}, \ldots, x_{nd}]$ and $v_d = [v_{1d}, v_{2d}, \ldots, v_{nd}]$, respectively. Suppose that the best previous position of particle d is represented by $pbest_d = [p_{1d}, p_{2d}, \ldots, p_{nd}]$ and the best particle among all particles is represented by $gbest$, the velocity and position of each particle in the next iteration $(k + 1)$ for are calculated as follows:

$$v_{id}^{(k+1)} = c_1^{(k)} \times rand_1 \times \left(pbest_d^{(k)} - x_{id}^{(k)}\right) + c_2^{(k)} \times rand_2 \times \left(gbest^{(k)} - x_{id}^{(k)}\right) \tag{12}$$

if $v_{id} = 0$ and $rand_3 < 0.5$ then

$$v_{id} = rand_4 \times v_{id,\max} \text{ else } v_{id} = -rand_5 \times v_{id,\max} \tag{13}$$

$$x_{id}^{(k+1)} = x_{id}^{(k)} + v_{id}^{(k+1)} \tag{14}$$

where $rand_i$, $i = 1, 2, \ldots, 5$ are randomly generated numbers in [0, 1] and the cognitive and social acceleration coefficients at iteration k are determined by:

$$c_1^{(k)} = (c_{1f} - c_{1i})\frac{k}{Iter_{\max}} + c_{1i} \tag{15}$$

$$c_2^{(k)} = (c_{2f} - c_{2i})\frac{k}{Iter_{\max}} + c_{2i} \tag{16}$$

where k is the iteration counter and $Iter_{\max}$ is the maximum number of iterations.

The upper and lower bounds for each particle position x_{id} are limited by the maximum and minimum limits of the variable represented by the particle, respectively. The velocity of each particle is limited in $[-v_{id,\max}, v_{id,\max}]$, $i = 1, \ldots, N$, $d = 1, \ldots, N_p$, where the maximum and minimum velocities for element i of particle d in the search space is determined by:

$$v_{id,\max} = R \times (x_{id,\max} - x_{id,\min}) \tag{17}$$

$$v_{id,\min} = -v_{id,\max} \tag{18}$$

where $v_{id,\max}$ and $v_{id,\min}$ are the maximum and minimum limits for element i of particle d and R is the limit coefficient for maximum velocity of particles.

3.2 Pseudo-Gradient Concept

Pseudo-gradient is to determine the search direction for each individual in population based methods [38]. The advantage of the pseudo-gradient is that it can provide a good direction in the search space of a problem without requiring the objective function to be differentiable. Therefore, the pseudo-gradient method is suitable for implementation on the meta-heuristic search methods for solving non-convex problems with multiple minima.

For a non-differentiable n-dimension optimization problem with objective function $f(x)$ where $x = [x_1, x_2, \ldots, x_n]$, a pseudo-gradient $g_p(x)$ for the objective function is defined as follows [35]: Supposed that $x_k = [x_{k1}, x_{k2}, \ldots, x_{kn}]$ is a point in the search space of the problem and it moves to another point x_l. There are two abilities for this movement.

1. If $f(x_l) < f(x_k)$, the direction from x_k to x_l is defined as the *positive direction*. The pseudo-gradient at point x_l is determined by:

$$g_p(x_l) = [\delta(x_{l1}), \delta(x_{l2}), \ldots, \delta(x_{ln})]^T \tag{19}$$

where $\delta(x_{li})$ is the direction indicator of element x_i moving from point k to point l defined by:

$$\delta(x_{li}) = \begin{cases} 1 & \text{if } x_{li} > x_{ki} \\ 0 & \text{if } x_{li} = x_{ki} \\ -1 & \text{if } x_{li} < x_{ki} \end{cases} \tag{20}$$

2. If $f(x_l) \geq f(x_k)$, the direction from x_k to x_l is defined as the *negative direction*. The pseudo-gradient at point x_l is determined by:

$$g_p(x_l) = 0 \tag{21}$$

The pseudo-gradient can also indicate a good direction similar to the conventional gradient in the search space based on the two last points. From the definition, if $g_p(x_l) \neq 0$, it implies that a better solution for the objective function could be found in the next step based on the direction indicated by the pseudo-gradient $g_p(x_l)$ at point l. Otherwise, the search direction at this point should be changed.

3.3 Proposed Pseudo-Gradient Based Particle Swarm Optimization

In this chapter, the proposed PGPSO is a combination of the HPSO-TVAC and pseudo-gradient. For implementation of the pseudo-gradient in the HPSO-TVAC, the two points considered here corresponding to x_k and x_l are $x^{(k)}$ and $x^{(k+1)}$, respectively. Therefore, the updated position for particles in (14) is rewritten as:

$$x_{id}^{(k+1)} = \begin{cases} x_{id}^{(k)} + \alpha \times \delta\left(x_{id}^{(k+1)}\right) \times \left|v_{id}^{(k+1)}\right| & \text{if } g_p\left(x_{id}^{(k+1)}\right) \neq 0 \\ x_{id}^{(k)} + v_{id}^{(k+1)} & \text{otherwise} \end{cases} \tag{22}$$

where $\alpha > 0$ is the acceleration factor for updating particle's position.

In (22), if the pseudo-gradient is non-zero, the position of the involved particle is quickly displaced to the global solution by its enhanced velocity; otherwise the position is normally updated as in (14). The value of α can be adjusted so that a particle can move faster or slower depending on the characteristic of each problem. In fact, too large value of α may lead to optimal solution ignored since the particles are at their limit positions while too small value of α may lead to particles trapped in local minima in the search space. In our experience, the proper values of α can be tuned in the range from 1 to 10.

The proposed PGPSO here is an improvement from the HPSO-TVAC method with its search capability enhanced by the pseudo-gradient. The advantage of the PGPSO is that its search capability is better than the HPSO-TVAC with the

support from the pseudo-gradient. During the search process, the particle's velocity will be accelerated if the pseudo-gradient indicates that the direction will lead to minimization of the objective function. Consequently, the particle can quickly move to the optimal position. In contrast, if the pseudo-gradient indicates that the direction will not lead to minimization of the objective function, the particle's position will be updated as the conventional PSO method. Moreover, the acceleration factor can be appropriately selected so that the particles can avoid ignoring optimal position or being trapped in local minima. Therefore, the PGPSO is a simple PSO method but more efficient than the HPSO-TVAC method, especially for nonconvex optimization problems with multiple minima.

4 Implementation of PGPSO to Nonconvex ED Problems

The proposed PGPSO here is implemented for the general problem which includes all the generator characteristics and constraints from the formulated problems. For properly handling the equality constraint of real power balance from the problem, the slack variable method is used. In addition, a heuristic search is also applied for a repairing strategy in case of prohibited zones violated. Other constraints for slack unit limits are handled in the fitness function for the problem. Therefore, all the constraints in the general problem are properly handled in the proposed PGPSO method.

4.1 Calculation of Power Output for Slack Unit

To guarantee that the equality constraint (3) is always satisfied, a slack generating unit is arbitrarily chosen and therefore its power output will be dependent on the power outputs of remaining $N - 1$ generating units in the system. The method for calculation of power output for the slack unit is given in Kuo [28].

4.2 Handling of Ramp Rate Constraints and POZ Violation

To handle the ramp rate limits, the highest and lowest possible power outputs of units are determined based on their power output limits, initial power output and ramp rate constraints as:

$$P_{i,\text{high}} = \min\{P_{i,\max}, P_{i0} + UR_i\} \tag{23}$$

$$P_{i,\text{low}} = \max\{P_{i,\min}, P_{i0} - DR_i\} \tag{24}$$

If the highest and lowest possible power outputs of a generating unit violate its POZ, the new limits should be redefined. Suppose that the highest or lowest possible power output of unit i violates its prohibited zone k, the new limit is redefined as follows:

$$P_{i,\text{high}}^{\text{new}} = \min\{P_{i,\text{high}}, P_{ik}^{l}\} \tag{25}$$

or

$$P_{i,\text{low}}^{\text{new}} = \max\{P_{i,\text{low}}, P_{ik}^{u}\} \tag{26}$$

When a unit operates in one of its prohibited zones, a repairing strategy is used to force the unit either to move toward the lower bound or upper bound of that zone. For making a decision based on the operating point of a unit located in one of its prohibited zones, the middle point of each prohibited zone P_{ik}^{m} is firstly determined by:

$$P_{ik}^{m} = \frac{P_{ik}^{l} + P_{ik}^{u}}{2} \tag{27}$$

This middle point divides a prohibited zone in two sub-zones, the left and right prohibited sub-zones with respect to the point. Therefore, the operating point P_i of unit i violating its prohibited zone k will be adjusted by:

$$P_i^{\text{new}} = \begin{cases} P_{ik}^{l} & \text{if } P_i \le P_{ik}^{m} \\ P_{ik}^{u} & \text{if } P_i > P_{ik}^{m} \end{cases} \tag{28}$$

However, when ramp rate constraints are included, the strategy for handling the POZ violation is more complicated. There are possibilities for adjusting the operating point of unit i when violating its prohibited zone k based on its initial power output P_{i0} and the violated position as follows:

- If $P_i > P_{ik}^{m}$ then

$$P_i^{\text{new}} = \begin{cases} \max\left\{P_{ik}^{u}, P_{i,\text{low}}\right\}, & \text{if } P_{i0} \ge P_{ik}^{u} \\ P_{ik}^{u}, & \text{if } P_{i0} \le P_{ik}^{l} \text{ and } P_{ik}^{u} \le P_{i,\text{high}} \\ \min\left\{P_{ik}^{l}, P_{i,\text{high}}\right\}, & \text{if } P_{i0} \le P_{ik}^{l} \text{ and } P_{ik}^{u} > P_{i,\text{high}} \end{cases} \tag{29}$$

- If $P_i < P_{ik}^{m}$ then

$$P_i^{\text{new}} = \begin{cases} \min\left\{P_{ik}^{l}, P_{i,\text{high}}\right\}, & \text{if } P_{i0} \le P_{ik}^{l} \\ P_{ik}^{l}, & \text{if } P_{i0} \ge P_{ik}^{u} \text{ and } P_{ik}^{l} \ge P_{i,\text{low}} \\ \max\left\{P_{ik}^{u}, P_{i,\text{low}}\right\}, & \text{if } P_{i0} \ge P_{ik}^{u} \text{ and } P_{ik}^{l} < P_{i,\text{low}} \end{cases} \tag{30}$$

4.3 Implementation of PGPSO

4.3.1 Initialization

A population of N_p particles is represented by $x = [x_1, x_2, \ldots, x_{Np}]$, where each particle's position $x_d = [P_{1d}, \ldots, P_{s-1d}, P_{s+1d}, \ldots, P_{Nd}]^T$ $(d = 1, \ldots, N_p)$ representing for power output of $N - 1$ generating units is initialized by:

$$x_{id}^{(0)} = (1 + rand) \times P_i^{(0)};\ i \neq s \tag{31}$$

where *rand* is a normally distributed stochastic number, $P_i^{(0)}$ is the initial operating point obtained by quadratic programming (QP) with quadratic objective function neglecting power loss, and s is the slack unit which is initially selected.

For the obtained initial solution, its upper and lower limits based on the generator limits should be satisfied:

$$x_{id} = \begin{cases} P_{i,\text{high}} & \text{if } x_{id} > P_{i,\text{high}} \\ P_{i,\text{low}} & \text{if } x_{id} < P_{i,\text{low}} \\ x_{id}^{(0)} & \text{otherwise} \end{cases};\ i \neq s \tag{32}$$

In addition, this initial solution should be also checked for POZ violation. If the violation is found, the repairing strategy in Sect. 4.2 is used to move the operating point to a feasible region. Based on the initial position of particles, the fitness function FT_d to be minimized corresponding to each particle for the considered problem is calculated:

$$FT_d = \sum_{i=1}^{N} F_i(x_{id}) + K_s \times \left(P_{sd} - P_s^{\text{lim}}\right)^2 \tag{33}$$

where K_s is the penalty factor for the slack unit, P_{sd} is the power output of the slack unit calculated from Sect. 4.1 corresponding to particle d in the population, and the power limits for the slack unit P_s^{lim} are determined based on its calculated output as follows:

$$P_s^{\text{lim}} = \begin{cases} P_{s,\text{high}} & \text{if } P_{sd} > P_{s,\text{high}} \\ P_{s,\text{low}} & \text{if } P_{sd} < P_{s,\text{low}} \\ P_{sd} & \text{otherwise} \end{cases} \tag{34}$$

where $P_{s,\text{high}}$ and $P_{s,\text{low}}$ are the highest and lowest possible power outputs of the slack unit.

The initial position is set to best value of each particle's position $pbest_d$ $(d = 1, \ldots, N_d)$ and the particle's position corresponding to the best fitness function in (33) is set to the best particle *gbest* among all particles in the population.

4.3.2 Calculation of Particle's Velocity and Position

The obtained initial position of particles is used for calculation of their velocity. The new velocity of particles is calculated as in (12–13) from the HPSO-TVAC method. The position of particles is then updated with the guidance of pseudo-gradient as in (22).

The new position of each particle should also satisfy their upper and lower limits as in (32) and the repairing strategy in Sect. 4.2 is used if any POZ violation is found. The new value of the fitness function is evaluated using (33) based on the new particle's position for determining the newly best position $pbest_d$ for each particle and best global position *gbest* among all particles.

4.3.3 Stopping Criteria

The algorithm of the proposed PGPSO is terminated when the predefined maximum number of iterations $Iter_{\max}$ is reached.

5 Numerical Results

The proposed PGPSO is tested on different systems corresponding to the formulated problems. The algorithm of the PGPSO is coded in Matlab platform and run 100 independent trials for each test case on a 2.1 GHz PC with 2 GB of RAM. In this chapter, the initial operating point is obtained by QP from optimization toolbox in Matlab.

5.1 Selection of Parameters

In the proposed PGPSO method, some parameters are predetermined for dealing with the systems such as cognitive acceleration and social factors, number of particles N_p, particle's velocity limit coefficient R, updating accelerator factor α, and maximum number of iterations $Iter_{\max}$. Among these parameters, the cognitive acceleration and social factors can be easily fixed by $c_{1i} = c_{2f} = 2.5$, $c_{2i} = c_{1f} = 0.2$ [23]. On the one hand, the penalty factor for the slack unit is large enough and set to 10^4 for all systems. On the other hand, the number of particles and maximum number of iterations depend on the dimension and complexity of problems. The number of particles is chosen in the range from twice to twenty times of the problem dimension while the maximum number of iterations is chosen in the range from 100 to 500 iterations. In this chapter, these parameters are chosen by experiments. For each problem, their value is first fixed at the low range and then increased. If the obtained result after one run is considerably improved in

Table 1 Additional parameters of PGPSO

System	No. of units	N_p	$Iter_{max}$	Initial R	α
Valve point effects	40	500	500	0.40	10.0
Multiple fuels	10	20	100	0.15	5.0
Prohibited operating zones	15	200	100	0.10	1.0
Valve point effects and multiple fuels	10	50	200	0.15	1.5
	20	50	500	0.15	2.5
	40	100	500	0.20	3.0
	80	200	500	0.10	2.0
	160	300	500	0.10	2.0

comparison with the previous one, their value will be increased. Otherwise, the obtained value is chosen as the proper value for multiple runs. In this chapter, the value of R will be reduced by 10 % from its initial value for every 100-iteration interval. By experiments, the values for the initial R and α can be chosen in a range of [0.1, 0.5] and [1, 10], respectively. For tuning R, its value is initially set to 0.1 and the step is set to 0.05 in the range [0.1, 0.25] and 0.1 in the range [0.3, 0.5].

Table 2 Solution for 40-unit system with VPE

Unit	P_{gi} (MW)	Unit	P_{gi} (MW)
1	110.8300	21	523.2797
2	110.8017	22	523.2841
3	97.4012	23	523.2808
4	179.7347	24	523.2798
5	92.6126	25	523.2801
6	139.9970	26	523.2837
7	259.6112	27	10.0000
8	284.6113	28	10.0000
9	284.6027	29	10.0037
10	130.0021	30	87.8077
11	168.8009	31	189.9993
12	168.7993	32	190.0000
13	214.7600	33	189.9986
14	304.5194	34	164.8063
15	394.2808	35	164.8018
16	394.2813	36	164.8468
17	489.2789	37	110.0000
18	489.2796	38	109.9999
19	511.2805	39	109.9994
20	511.2800	40	511.2830
Min cost ($/h)	121,415.2447		
Average cost ($/h)	121,998.6771		
Max cost ($/h)	122,746.9205		
Standard deviation ($/h)	329.0243		
Average CPU time (s)	5.895		

The value of α is also tuned by similar way by setting its initial value to 1.0 with the step of 0.5 in the range [1, 3] and 1.0 in the range [4, 10]. By experiments, the fine tuned parameters for each considered test system in this chapter are shown in Table 1.

5.2 Systems with Valve Point Effects

The proposed PGPSO is tested on a large-scale system from [13] comprising 40 generating units supplying to a load demand of 10,500 MW neglecting power loss. The optimal solution obtained by the proposed PGPSO method for this case is given in Table 2.

Table 3 shows a comparison of total cost and computational time from the proposed PGPSO method to those from improved fast EP (IFEP) [13], MSL [8], MPSO [24], evolutionary strategy optimization (ESO) [14], DEC-SQP [30, 31], self-tuning HDE [32], NPSO-LRS [27], NDS [9], SOH_PSO [23], QPSO [22], ABC [17], SA-PSO [28], BBO [19], UHGA [29], improved coordinated aggregation-based PSO (ICA-PSO) [39, 40], FAPSO-NM [33], DE [16], modified differential evolution (MDE) [41], and integration of the variable DE with the fuzzy adaptive PSO (FAPSO-VDE) [42]. Apparently, the PGPSO method obtains a less total cost of 121,415.2447 ($/h) than the total cost of the other methods except the MDE and FAPSO-VDE methods. Moreover, the PGPSO also obtains a higher

Table 3 Comparison of best total cost and CPU time for 40-unit system with VPE

Method	Total cost ($/h)	CPU time (s)
IFEP [13]	122,624.35	1,167.35
MSL [8]	122,406.10	0.078
MPSO [24]	122,252.27	–
ESO [14]	122,122.16	0.261
DEC-SQP [30, 31]	121,741.98	14.26
Self-tuning HDE [32]	121,698.51	6.07
NPSO-LRS [27]	121,664.43	20.74
NDS [9]	121,647.40	4.0471
SOH_PSO [23]	121,501.14	–
QPSO [22]	121,448.21	–
ABC [17]	121,441.03	32.45
SA-PSO [28]	121,430.00	23.89
BBO [19]	121,426.95	1.1749
UHGA [29]	121,424.48	333.68
ICA-PSO [39], [40]	121,422.10	139.92
FAPSO-NM [33]	121,418.30	40
DE [16]	121,416.29	72.94
MDE [41]	121,414.79	–
FAPSO–VDE [42]	121,412.56	22
PGPSO	121,415.24	5.895

quality solution in a faster manner with 5.895 s which is less than those from the other methods except MSL, ESO, NDS, and BBO methods. The computational time from IFEP, MSL, ESO, DEC-SQP, self-tuning HDE, NPSO-LRS, NDS, ABC, BBO, UHGA, ICA-PSO, DE, and FAPSO–VDE were from a Pentium-II 350 MHz with 128 MB of RAM PC, Pentium IV 1.5 GHz with 512 MB of RAM PC, Pentium IV PC, 1.1 AMD Athlon GHz with 112 MB of RAM, Pentium 1.5 GHz with 768 MB of RAM, Pentium IV 1.5 GHz with 128 MB of RAM, Pentium IV 2.6 GHz with 512 MB of RAM, 1.7 GHz with 1 GB of RAM, Pentium IV 2.3 GHz with 512 MB of RAM PC, Pentium IV 2.99 GHz PC, Pentium IV 1.4 GHz PC, Intel 1.67 GHz with 1 GB of RAM PC, and Pentium-IV 3.0 GHz PC, respectively. There is no computational time or computer processor reported for the other methods. In fact, it may not be directly comparable the computational time among the methods due to different computers and programming languages used. However, the comparison of computational time can be considered as a basis for estimation of their efficiency for dealing with the nonconvex ED problems.The test results have indicated that the PGPSO is effective for solving nonconvex ED with valve point effects.

5.3 System with Multiple Fuels

The test system from [10] comprises 10 generating units with multiple fuel options. Various load demands are considered for this system including 2,400, 2,500, 2,600, and 2,700 MW neglecting power loss. The solutions for different

Table 4 Results for 10-unit system with MF

Unit	Load demand of 2,400 MW		Load demand of 2,500 MW		Load demand of 2,600 MW		Load demand of 2,700 MW	
	F_i	P_i (MW)	F_i	P_i (MW)	F_i	P_i (MW)	F_i	P_i (MW)
1	1	189.7807	2	206.4667	2	216.2074	2	218.3351
2	1	202.3332	1	206.5038	1	210.8414	1	211.5971
3	1	253.8521	1	265.8326	1	278.5687	1	280.6878
4	3	233.1025	3	235.9293	3	238.9349	3	239.6504
5	1	241.8250	1	257.9819	1	275.7003	1	278.4149
6	3	232.9748	3	235.9461	3	239.0876	3	239.6019
7	1	253.4319	1	269.0991	1	286.0002	1	288.6804
8	3	233.0516	3	235.9656	3	239.1758	3	239.4526
9	1	320.3100	1	331.3439	1	343.3878	3	428.8039
10	1	239.3382	1	254.9309	1	272.0959	1	274.7759
Min total cost ($/h)		481.7227		526.2389		574.3814		623.8095
Average total cost ($/h)		485.3323		531.7085		576.9959		626.7308
Max total cost ($/h)		507.3870		571.6084		598.7630		646.2601
Standard deviation ($/h)		5.4601		7.9350		4.0521		4.3292
CPU (s)		0.228		0.228		0.229		0.233

Table 5 Comparison of best total costs and average CPU times for 10-unit system with MF

Method	2,400 MW		2,500 MW	
	Cost ($/h)	CPU (s)	Cost ($/h)	CPU (s)
HNUM [5]	488.50	1.08	526.70	1.08
HNN [10]	487.87	~60	526.13	~60
AHNN [11]	481.72	~4	526.230	~4
MPSO [24]	481.723	–	526.239	–
AIS [18]	481.723	–	526.240	–
ALHN [43]	481.723	0.042	526.239	0.043
PGPSO	481.723	0.228	526.239	0.228
Method	2,600 MW		2,700 MW	
	Cost ($/h)	CPU (s)	Cost ($/h)	CPU (s)
HNUM [5]	574.03	1.08	625.18	1.08
HNN [10]	574.26	~60	626.12	~60
AHNN [11]	574.37	~4	626.24	~4
MPSO [24]	574.381	–	623.809	–
AIS [18]	574.381	–	623.809	–
ALHN [43]	574.381	0.047	623.809	0.057
PGPSO	574.381	0.225	623.810	0.233

system load demands are given in Table 4. The total costs obtained by the PGPSO for the system with corresponding load demands of 2,400, 2,500, 2,600, and 2,700 are 481.7227 ($/h), 526.2389 ($/h), 574.3814 ($/h), and 623.8095 ($/h), respectively.

A comparison of the best total costs and average computational times of the PGPSO and HNUM [5], HNN [10], adaptive HNN (AHNN) [11], MPSO [24], AIS [18], and augmented Lagrange Hopfield network (ALHN) [43] is made as in Table 5. For this system, the best total costs from the PGPSO are less than those from HNUM and HNN for the load demand of 2,400 MW and HNUM, HNN, and AHNN for the load demand of 2,700 MW and close to those from the other methods for the remaining cases. For the computational times, the PGPSO is faster than HNUM, HNN, and AHNN and slightly slower than ALHN. The CPU times from the HNUM, HNN, AHNN, and ALHN methods were from a VAX 11/780, IBM PC-386, Compaq 90, and 2.1 GHz PC. There is no CPU time reported for the AIS method. From the obtained results, it has shown that the PGPSO can obtain the better solutions in a faster manner due to its powerful search ability. Accordingly, the proposed PGPSO is effective for solving nonconvex ED with multiple fuels.

Table 6 Results for 15-unit system with POZ

Unit	P_i (MW)	Unit	P_i (MW)
1	454.9644	9	43.2998
2	380.0000	10	160.0000
3	130.0000	11	80.0000
4	130.0000	12	80.0000
5	170.0000	13	25.0721
6	460.0000	14	15.6087
7	430.0000	15	15.7862
8	85.9332		
Power loss (MW)	30.6644		
Total power (MW)	2,660.6644		
Minimum total cost ($/h)	32,705.7533		
Average total cost ($/h)	32,716.8369		
Maximum total cost ($/h)	32,726.0751		
Standard deviation ($/h)	4.0221		
CPU time (s)	1.631		

Table 7 Comparison of best total cost and average CPU time for 15-unit system with POZ

Method	Power loss (MW)	Total power (MW)	Total cost ($/h)	CPU time (s)
GA [20]	38.2782	2,668.40	33,113.00	4.95
PSO [20]	32.4306	2,662.40	32,858.00	2.74
AIS 18	32.4075	2,662.04	32,854.00	–
SOH_PSO [23]	32.2800	2,662.29	32,751.00	0.0936
MPSO [25]	29.9780	2,661.62	32,738.42	–
PSO-MSAF [26]	30.4900	2,660.49	32,713.09	19.15
SA-PSO [28]	30.9080	2,660.90	32,708.00	10.37
PGPSO	30.6644	2,660.66	32,705.75	1.632

5.4 System with Prohibited Operating Zones

The test system consists of 15 units supplying to a load demand of 2,630 MW, considering ramp rate constraints and system power loss [20]. The results obtained by the proposed PGPSO for this system are given in Table 6.

The result obtained by the PGPSO is compared to that from GA and PSO in [20], AIS [18], SOH_PSO [23], MPSO [25], PSO-MSAF [26], and SA-PSO [28] as in Table 7. The proposed method obtains a less total cost of 32,705.7533 ($/h) than the best total cost from the other methods. In addition, the proposed method is faster than GA, PSO, PSO-MSAF, and SA-PSO and slightly slower than SOH_PSO. The GA and PSO, SOH_PSO, and PSO-MSAF methods were implemented on a Pentium III 550 with 256 MB of RAM, Pentium IV 2.8 GHz with 512 MB of RAM, and Pentium IV 2.60 GHz with 512 MB of RAM, respectively. There is no computational time reported for the AIS and MPSO methods. The result

Table 8 Results for 10-unit system with VPE and MF

Unit	F_i	P_i (MW)
1	2	218.5941
2	1	210.9690
3	1	280.6574
4	3	240.1769
5	1	279.6374
6	3	240.1769
7	1	290.0615
8	3	239.3707
9	3	427.7234
10	1	272.6326
Min total cost ($/h)		623.8431
Average total cost ($/h)		624.0979
Max total cost ($/h)		626.5191
Standard deviation ($/h)		0.5428
CPU time (s)		1.494

Table 9 Comparison of best total cost and average CPU time for 10-unit system with VPE and MF

Method	Total cost ($/h)	CPU time (s)
CGA_MU [12]	624.7193	26.64
IGA_MU [12]	624.5178	7.32
PSO-LRS [27]	624.2297	1.81
NPSO [27]	624.1624	0.76
NPSO-LRS [27]	624.1273	1.60
RGA [15]	624.5081	4.1340
DE [15]	624.5146	2.8236
PSO [15]	624.5074	3.3852
PGPSO	623.8431	1.494

comparison has indicated that the PGPSO can obtain better solution quality than many other methods for the nonconvex ED with POZ.

5.5 Systems with Both Valve Point Effects and Multiple Fuels

5.5.1 10-unit System

The test system from [12] comprises 10 units with VPE and MF. This system supplies to a load demand of 2,700 MW neglecting power loss. Table 8 shows the result obtained by the proposed PGPSO for this system.

The best total cost and average computational time from the proposed method are compared to those from other methods such as conventional GA with multiplier updating (CGA_MU) and improved GA with multiplier updating (IGA_MU) in Chiang [12], PSO with local random search (PSO-LRS), new PSO (NPSO), and NPSO-LRS in Selvakumar and Thanushkodi [27], and real-coded GA (RGA), PSO, and DE in Manoharan et al. [15] as in Table 9. Obviously, the total cost obtained from the proposed method for this system is less than that from the other methods. The proposed PGPSO is also faster than the CGA_MU, IGA_MU, PSO-LRS, NPSO-LRS, RGA, PSO, and DE methods and slightly slower than the NPSO method. The computational times for the CGA_MU and IGA_MU methods were from a PIII-700 PC, the computational times for the PSO-LRS, NPSO, and NPSO-LRS methods from a Pentium IV 1.5 GHz with 128 MB of RAM, and the computational times for the CGA, PSO, and DE method from a Pentium IV 1.8 GHz with 1 GB of RAM PC. Test result indicates that the PGPSO can obtain better solution quality than many other methods for nonconvex ED problem with valve point effects and multiple fuels.

Table 10 Results for large-scale systems with VPE and MF

No. of units	20	40	80	160
Min total cost ($/h)	1,247.7326	2,495.6162	4,994.2781	10,004.7260
Average total cost ($/h)	1,248.9623	2,499.6127	5,003.0250	10,032.4883
Max total cost ($/h)	1,259.2242	2,512.9091	5,021.0196	10,107.1708
Standard deviation ($/h)	2.0378	3.8971	6.2272	17.0031
CPU time (s)	4.078	18.645	43.191	91.570

Table 11 Comparison of average total cost and average CPU time for large-scale systems with VPE and MF

Method	No. of units	Total cost ($)	CPU time (s)
CGA_MU [12]	20	1,249.3893	80.48
	40	2,500.9220	157.39
	80	5,008.1426	309.41
	160	10,143.7263	621.30
IGA_MU [12]	20	1,249.1179	21.64
	40	2,499.8243	43.71
	80	5,003.8832	85.67
	160	10,042.4742	174.62
PGPSO	20	1,248.9623	4.078
	40	2,499.6127	18.645
	80	5,003.0250	43.191
	160	10,032.4883	91.570

5.5.2 Large-Scale Systems

The large-scale systems here consisting of 20, 40, 80, and 160 units are based on the basic 10-unit system above. The considered systems are created by duplicating the basic 10-unit system with the load demand adjusted proportionally to the system size. The results obtained by the proposed method for these systems including minimum costs, average costs, maximum costs, standard deviations, and computational times are given in Table 10. For all systems, the difference between the maximum and minimum costs obtained the proposed method is small and the ratio between the standard deviation and the minimum cost is less than 0.17 %. In Table 11, the average total costs and computational times from the PGPSO method are compared to those from CGA_MU and IGA_MU methods in Chiang [12]. The result comparison has shown that the proposed PGPSO can obtain better average total costs with faster average computational times than both CGA_MU and IGA_MU. Therefore, the proposed method is also effective for solving large-scale nonconvex ED problems with valve point effects and multiple fuels.

6 Conclusion

In this chapter, the newly proposed PGPSO method has been efficiently implemented for solving different nonconvex ED problems. The PGPSO method is a novel improvement of PSO by combining the HPSO-TVAC method with the pseudo-gradient to improve its search capability. The pseudo-gradient is efficient for guiding the search direction for each individual in population based methods. In addition, an efficient repairing strategy is also used for handling POZ violation considering ramp rate constraints. With the new improvement, the proposed PGPSO method is more effective than many other methods in solving nonconex ED problems with multiple minima. The proposed method has been tested on different systems with nonconvex generator's characteristics including valve point effects, multiple fuels, and prohibited operating zones. Test results have shown that the proposed PGPSO can obtain better solution quality than many other methods, leading substantial cost savings.

Appendix

The unit data for 40-unit system with valve point effects are given in Table A1.

The unit data for 10-unit system with multiple fuels is given in Table A2.

The unit data for 15-unit system with prohibited zones is given in Table A3 and prohibited zones are given in Table A4.

The unit data for 10-unit system with valve pint effects and multiple fuels is given in Table A5.

Table A1 Unit data for 40-unit system with valve point effects

Unit	a_i ($/h)	b_i ($/MWh)	c_i ($/MW^{2}h)	e_i ($/h)	f_i (1/MW)	P_{imin} (MW)	P_{imax} (MW)
1	94.705	6.73	0.0069	100	0.084	36	114
2	94.705	6.73	0.0069	100	0.084	36	114
3	309.54	7.07	0.02028	100	0.084	60	120
4	369.03	8.18	0.00942	150	0.063	80	190
5	148.89	5.35	0.0114	120	0.077	47	97
6	222.33	8.05	0.01142	100	0.084	68	140
7	287.71	8.03	0.00357	200	0.042	110	300
8	391.98	6.99	0.00492	200	0.042	135	300
9	455.76	6.6	0.00573	200	0.042	135	300
10	722.82	12.9	0.00605	200	0.042	130	300
11	635.2	12.9	0.00515	200	0.042	94	375
12	654.69	12.8	0.00569	200	0.042	94	375
13	913.4	12.5	0.00421	300	0.035	125	500
14	1,760.4	8.84	0.00752	300	0.035	125	500
15	1,728.3	9.15	0.00708	300	0.035	125	500
16	1,728.3	9.15	0.00708	300	0.035	125	500
17	647.85	7.97	0.00313	300	0.035	220	500
18	649.69	7.95	0.00313	300	0.035	220	500
19	647.83	7.97	0.00313	300	0.035	242	550
20	647.81	7.97	0.00313	300	0.035	242	550
21	785.96	6.63	0.00298	300	0.035	254	550
22	785.96	6.63	0.00298	300	0.035	254	550
23	794.53	6.66	0.00284	300	0.035	254	550
24	794.53	6.66	0.00284	300	0.035	254	550
25	801.32	7.1	0.00277	300	0.035	254	550
26	801.32	7.1	0.00277	300	0.035	254	550
27	1,055.1	3.33	0.52124	120	0.077	10	150
28	1,055.1	3.33	0.52124	120	0.077	10	150
29	1,055.1	3.33	0.52124	120	0.077	10	150
30	148.89	5.35	0.0114	120	0.077	47	97
31	222.92	6.43	0.0016	150	0.063	60	190
32	222.92	6.43	0.0016	150	0.063	60	190
33	222.92	6.43	0.0016	150	0.063	60	190
34	107.87	8.95	0.0001	200	0.042	90	200
35	116.58	8.62	0.0001	200	0.042	90	200
36	116.58	8.62	0.0001	200	0.042	90	200
37	307.45	5.88	0.0161	80	0.098	25	110
38	307.45	5.88	0.0161	80	0.098	25	110
39	307.45	5.88	0.0161	80	0.098	25	110
40	647.83	7.97	0.00313	300	0.035	242	550

Table A2 Unit data for 10-unit system with multiple fuels

Unit	Fuel type	a_{ij} ($/h)	b_{ij} ($/MWh)	c_{ij} ($/MW^{2}h)	$P_{ij\min}$ (MW)	$P_{ij\max}$ (MW)
1	1	26.97	−0.3975	0.002176	100	196
	2	21.13	−0.3059	0.001861	196	250
2	2	1.865	−0.03988	0.001138	50	114
	3	13.65	−0.198	0.00162	114	157
	1	118.4	−1.269	0.004194	157	230
3	1	39.79	−0.3116	0.001457	200	332
	3	−2.876	0.03389	0.000804	332	388
	2	−59.14	0.4864	1.18E-05	388	500
4	1	1.983	−0.03114	0.001049	99	138
	2	52.85	−0.6348	0.002758	138	200
	3	266.8	−2.338	0.005935	200	265
5	1	13.92	−0.08733	0.001066	190	338
	2	99.76	−0.5206	0.001597	338	407
	3	−53.99	0.4462	0.00015	407	490
6	2	1.983	−0.03114	0.001049	85	138
	1	52.85	−0.6348	0.002758	138	200
	3	266.8	−2.338	0.005935	200	265
7	1	18.93	−0.1325	0.001107	200	331
	2	43.77	−0.2267	0.001165	331	391
	3	−43.35	0.3559	0.000245	391	500
8	1	1.983	−0.03114	0.001049	99	138
	2	52.85	−0.6348	0.002758	138	200
	3	266.8	−2.338	0.005935	200	265
9	3	14.23	−0.01817	0.000612	130	213
	1	88.53	−0.5675	0.001554	213	370
	3	14.23	−0.01817	0.000612	370	440
10	1	13.97	−0.09938	0.001102	200	362
	3	46.71	−0.2024	0.001137	362	407
	2	−61.13	0.5084	4.16E-05	407	490

Table A3 Unit data for 15-unit system with prohibited zones

Unit	a_i ($/h)	b_i ($/MWh)	c_i ($/MW^{2}h)	$P_{i\min}$ (MW)	$P_{i\max}$ (MW)	$S_{i\max}$ (MW)	UR_i (MW)	DR_i (MW)	P_{i0} (MW)
1	671	10.1	0.000299	150	455	50	80	120	400
2	574	10.2	0.000183	150	455	0	80	120	300
3	374	8.8	0.001126	20	130	30	130	130	105
4	374	8.8	0.001126	20	130	30	130	130	100
5	461	10.4	0.000205	150	470	0	80	120	90
6	630	10.1	0.000301	135	460	0	80	120	400
7	548	9.8	0.000364	135	465	50	80	120	350
8	227	11.2	0.000338	60	300	50	65	100	95
9	173	11.2	0.000807	25	162	30	60	100	105

(continued)

Table A3 (continued)

Unit	a_i ($/h)	b_i ($/MWh)	c_i ($/MW^{2}h)	$P_{i\min}$ (MW)	$P_{i\max}$ (MW)	$S_{i\max}$ (MW)	UR_i (MW)	DR_i (MW)	P_{i0} (MW)
10	175	10.7	0.001203	25	160	30	60	100	110
11	186	10.2	0.003586	20	80	20	80	80	60
12	230	9.9	0.005513	20	80	0	80	80	40
13	225	13.1	0.000371	25	85	20	80	80	30
14	309	12.1	0.001929	15	55	40	55	55	20
15	323	12.4	0.004447	15	55	40	55	55	20

Table A4 Prohibited zones for 15-unit system

Unit	Prohibited zone 1	Prohibited zone 2	Prohibited zone 3
2	(185 225)	(305 335)	(420 450)
5	(180 200)	(305 335)	(390 420)
6	(230 255)	(365 395)	(430 455)
12	(30 40)	(55 65)	

Table A5 Unit data for 10-unit system with valve point effects and multiple fuels

Unit	Fuel type	a_{ij} ($/h)	b_{ij} ($/MWh)	c_{ij} ($/MW^{2}h)	e_{ij} ($/h)	f_{ij} (1/MW)	$P_{ij\min}$ (MW)	$P_{ij\max}$ (MW)
1	1	26.97	−0.3975	0.002176	0.02697	−3.975	100	196
	2	21.13	−0.3059	0.001861	0.02113	−3.059	196	250
2	2	1.865	−0.03988	0.001138	0.001865	−0.3988	50	114
	3	13.65	−0.198	0.00162	0.01365	−1.98	114	157
	1	118.4	−1.269	0.004194	0.1184	−12.69	157	230
3	1	39.79	−0.3116	0.001457	0.03979	−3.116	200	332
	3	−2.876	0.03389	0.000804	−0.00288	0.3389	332	388
	2	−59.14	0.4864	1.18E-05	−0.05914	4.864	388	500
4	1	1.983	−0.03114	0.001049	0.001983	−0.3114	99	138
	2	52.85	−0.6348	0.002758	0.05285	−6.348	138	200
	3	266.8	−2.338	0.005935	0.2668	−23.38	200	265
5	1	13.92	−0.08733	0.001066	0.01392	−0.8733	190	338
	2	99.76	−0.5206	0.001597	0.09976	−5.206	338	407
	3	−53.99	0.4462	0.00015	−0.05399	4.462	407	490
6	2	1.983	−0.03114	0.001049	0.001983	−0.3114	85	138
	1	52.85	−0.6348	0.002758	0.05285	−6.348	138	200
	3	266.8	−2.338	0.005935	0.2668	−23.38	200	265
7	1	18.93	−0.1325	0.001107	0.01893	−1.325	200	331
	2	43.77	−0.2267	0.001165	0.04377	−2.267	331	391
	3	−43.35	0.3559	0.000245	−0.04335	3.559	391	500

(continued)

Table A5 (continued)

Unit	Fuel type	a_{ij} ($/h)	b_{ij} ($/MWh)	c_{ij} ($/MW^{2}h)	e_{ij} ($/h)	f_{ij} (1/MW)	$P_{ij\min}$ (MW)	$P_{ij\max}$ (MW)
8	1	1.983	−0.03114	0.001049	0.001983	−0.3114	99	138
	2	52.85	−0.6348	0.002758	0.05285	−6.348	138	200
	3	266.8	−2.338	0.005935	0.2668	−23.38	200	265
9	3	14.23	−0.01817	0.000612	0.01423	−0.1817	130	213
	1	88.53	−0.5675	0.001554	0.08853	−5.675	213	370
	3	14.23	−0.01817	0.000612	0.01423	−0.1817	370	440
10	1	13.97	−0.09938	0.001102	0.01397	−0.9938	200	362
	3	46.71	−0.2024	0.001137	0.04671	−2.024	362	407
	2	−61.13	0.5084	4.16E-05	−0.06113	5.084	407	490

References

1. Chowdhury, E.H., Rahrnan, S.: A review of recent advances in economic dispatch. IEEE Trans. Power Syst. **5**, 1248–1259 (1990)
2. Xia, X., Elaiw, A.M.: Optimal dynamic economic dispatch of generation: a review. Electr. Power Syst. Res. **80**, 975–986 (2010)
3. Wood, A.J., Wollenberg, B.F.: Power Generation, Operation, and Control, 2nd edn. Wiley, New York (1996)
4. Fink, L.H., Kwatny, H.G., Mcdonald, J.P.: Economic dispatch of generation via valve-point loading. IEEE Trans. Power Appar. Syst. **88**, 805–811 (1969)
5. Lin, C.E., Viviani, G.L.: Hierarchical economic dispatch for piecewise quadratic cost functions. IEEE Trans. Power Appar. Syst. **103**, 1170–1175 (1984)
6. Lee, F.N., Breipohl, A.M.: Reserve constrained economic dispatch with prohibited operating zones. IEEE Trans. Power Syst. **8**, 246–254 (1993)
7. Fan, J.Y., McDonald, J.D.: A practical approach to real time economic dispatch considering unit's prohibited operating zones. IEEE Trans. Power Syst. **9**, 1737–1743 (1994)
8. Hemamalini, S., Simon, S.P.: Maclaurin series-based Lagrangian method for economic dispatch with valve-point effect. IET Gener. Transm. Distrib. **3**, 859–871 (2009)
9. Lin, W.-M., Gowa, H.-J., Tsai, M.-T.: Combining of direct search and signal-to-noise ratio for economic dispatch optimization. Energy Conver. Manag. **52**, 487–493 (2011)
10. Park, J.H., Kim, Y.S., Eom, I.K., Lee, K.Y.: Economic load dispatch for piecewise quadratic cost function using Hopfield neural network. IEEE Trans. Power Syst. **8**, 1030–1038 (1993)
11. Lee, K.Y., Sode-Yome, A., Park, J.H.: Adaptive Hopfield neural networks for economic load dispatch. IEEE Trans. Power Syst. **13**, 519–526 (1998)
12. Chiang, C.-L.: Improved genetic algorithm for power economic dispatch of units with valve-point effects and multiple fuels. IEEE Trans. Power Syst. **20**, 1690–1699 (2005)
13. Sinha, N., Chakrabarti, R., Chattopadhyay, P.K.: Evolutionary programming techniques for economic load dispatch. IEEE Trans. Evol. Comput. **7**, 83–94 (2003)
14. Pereira-Neto, A., Unsihuay, C., Saavedra, O.R.: Efficient evolutionary strategy optimization procedure to solve the nonconvex economic dispatch problem with generator constraints. IEE Proc. Gener. Transm. Distrib. **152**, 653–660 (2005)
15. Manoharan, P.S., Kannan, P.S., Baskar, S., Iruthayarajan, M.W.: Penalty parameter-less constraint handling scheme based evolutionary algorithm solutions to economic dispatch. IET Gener. Transm. Distrib. **2**, 478–490 (2008)
16. Noman, N., Iba, H.: Differential evolution for economic load dispatch problems. Elec. Power Syst. Res. **78**, 1322–1331 (2008)

17. Hemamalini, S., Simon, S.P.: Artificial bee colony algorithm for economic load dispatch problem with non-smooth cost functions. Elec. Power Comp. Syst. **38**, 786–803 (2010)
18. Panigrahi, B.K., Yadav, S.R., Agrawal, S., Tiwari, M.K.: A clonal algorithm to solve economic load dispatch. Elec. Power Syst. Res. **77**, 1381–1389 (2007)
19. Bhattacharya, A., Chattopadhyay, P.K.: Biogeography-based optimization for different economic load dispatch problems. IEEE Trans. Power Syst. **25**, 1064–1077 (2010)
20. Gaing, Z.L.: Particle swarm optimization to solving the economic dispatch considering the generator constraints. IEEE Trans. Power Syst. **18**, 1187–1195 (2003)
21. Mahor, A., Prasad, V., Rangnekar, S.: Economic dispatch using particle swarm optimization: a review. Renew. Sustain. Energy Rev. **13**, 2134–2141 (2009)
22. Meng, K., Wang, H.G., Dong, Z.Y., Wong, K.P.: Quantum-inspired particle swarm optimization for valve-point economic load dispatch. IEEE Trans. Power Syst. **25**, 215–222 (2010)
23. Chaturvedi, K.T., Pandit, M., Srivastava, L.: Self-organizing hierarchical particle swarm optimization for nonconvex economic dispatch. IEEE Trans. Power Syst. **23**, 1079–1087 (2008)
24. Park, J.-B., Lee, K.-S., Shin, J.-R., Lee, K.Y.: A particle swarm optimization for economic dispatch with nonsmooth cost functions. IEEE Trans. Power Syst. **20**, 34–42 (2005)
25. Neyestani, M., Farsangi, M.M., Nezamabadi-pour, H.: A modified particle swarm optimization for economic dispatch with non-smooth cost functions. Eng. Apps. AI **23**, 1121–1126 (2010)
26. Subbaraj, P., Rengaraj, R., Salivahanan, S., Senthilkumar, T.R.: Parallel particle swarm optimization with modified stochastic acceleration factors for solving large scale economic dispatch problem. Electr. Power Energy Syst. **32**, 1014–1023 (2010)
27. Selvakumar, A.I., Thanushkodi, K.: A new particle swarm optimization solution to nonconvex economic dispatch problems. IEEE Trans. Power Syst. **22**, 42–51 (2007)
28. Kuo, C.-C.: A novel coding scheme for practical economic dispatch by modified particle swarm approach. IEEE Trans. Power Syst. **23**, 1825–1835 (2008)
29. He, D.-K., Wang, F.-L., Mao, Z.-Z.: Hybrid genetic algorithm for economic dispatch with valve-point effect. Elec. Power Syst. Res. **78**, 626–633 (2008)
30. dos Santos Coelho, L., Mariani, V.C.: Combining of chaotic differential evolution and quadratic programming for economic dispatch optimization with valve-point effect. IEEE Trans. Power Syst. **21**, 989–996 (2006)
31. dos Santos Coelho, L., Mariani, V.C.: Correction to Combining of chaotic differential evolution and quadratic programming for economic dispatch optimization with valve-point effect. IEEE Trans. Power Syst. **21**, 1465 (2006)
32. Wang, S.-K., Chiou, J.-P., Liu, C.-W.: Non-smooth/non-convex economic dispatch by a novel hybrid differential evolution algorithm. IET Gener. Transm. Distrib. **1**, 793–803 (2007)
33. Niknam, T.: A new fuzzy adaptive hybrid particle swarm optimization algorithm for non-linear, non-smooth and non-convex economic dispatch problem. Appl. Energy **87**, 327–339 (2010)
34. Ratnaweera, A., Halgamuge, S.K., Watson, H.C.: Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients. IEEE Trans. Evol. Comput. **8**, 240–255 (2004)
35. Wen, J.Y., Wu, Q.H., Jiang, L., Cheng, S.J.: Pseudo-gradient based evolutionary programming. Electron. Lett. **39**, 631–632 (2003)
36. Walter, D.C., Sheble, G.B.: Genetic algorithm solution of economic load dispatch with valve point loading. IEEE Trans. Power Syst. **8**, 1325–1332 (1993)
37. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings of the IEEE Conference Neural Networks (ICNN'95), vol. 4, pp. 1942–1948 (1995)
38. Pham, D.T., Jin, G.: Genetic algorithm using gradient-like reproduction operator. Electron. Lett. **31**, 1558–1559 (1995)

39. Vlachogiannis, J.G., Lee, K.Y.: Economic load dispatch: a comparative study on heuristic optimization techniques with an improved coordinated aggregation-based PSO. IEEE Trans. Power Syst. **24**, 991–1001 (2009)
40. Vlachogiannis, J.G., Lee, K.Y.: Closure to discussion on Economic load dispatch: a comparative study on heuristic optimization techniques with an improved coordinated aggregation-based PSO. IEEE Trans. Power Syst. **25**, 591–592 (2010)
41. Amjady, N., Sharifzadeh, H.: Solution of non-convex economic dispatch problem considering valve loading effect by a new modified differential evolution algorithm. Int. J. Electr. Power Energy Syst. **32**, 893–903 (2010)
42. Niknam, T., Mojarrad, H.D., Meymand, H.Z.: A novel hybrid particle swarm optimization for economic dispatch with valve-point loading effects. Energy Convers. Manage. **52**, 1800–1809 (2011)
43. Dieu, V.N., Ongsakul, W., Polprasert, J.: The augmented Lagrange Hopfield network for economic dispatch with multiple fuel options. Math. Comp. Model. **57**, 30–39 (2013)