Hindawi
Complexity
Volume 2018, Article ID 7276585, 15 pages
https://doi.org/10.1155/2018/7276585

Research Article

An Elitist Transposon Quantum-Based Particle Swarm Optimization Algorithm for Economic Dispatch Problems

Angus Wu[1] and Zhen-Lun Yang[2]

[1]*Department of Electronic Engineering, City University of Hong Kong, Tat Chee Ave, Kowloon Tong, Hong Kong*
[2]*School of Information Engineering, Guangzhou Panyu Polytechnic, Guangzhou 511483, China*

Correspondence should be addressed to Angus Wu; angus.wu@cityu.edu.hk

Received 23 February 2018; Revised 7 May 2018; Accepted 21 May 2018; Published 17 July 2018

Academic Editor: Fernando Lezama

Population-based optimization algorithms are useful tools in solving engineering problems. This paper presents an elitist transposon quantum-based particle swarm algorithm to solve economic dispatch (ED) problems. It is a complex and highly nonlinear constrained optimization problem. The proposed approach, double elitist breeding quantum-based particle swarm optimization (DEB-QPSO), makes use of two elitist breeding strategies to promote the diversity of the swarm so as to enhance the global search ability and an improved efficient heuristic handling technique to manage the equality and inequality constraints of ED problems. Investigating on 15-unit, 40-unit, and 140-unit widely used test systems, through performance comparison, the proposed DEB-QPSO algorithm is able to obtain higher-quality solutions efficiently and stably superior than the other the state-of-the-art algorithms.

1. Introduction

Economic dispatch (ED) of electric power generation is used to determine an optimal combination of power output from the units in the system for a minimal total generation cost meeting the load demand while satisfying all equality and inequality constraints of the units and system. The constraints involved discontinuous prohibited zones, unit power limits, and ramp rate limits making the practical ED problem a highly constrained nonconvex and nonlinear optimization problem [1]. The cost function of ED problems can be represented by a quadratic function and solved by conventional methods such as the gradient method, dynamic programming, and the lambda-iteration method [2, 3]. However, none of these conventional optimization methods is able to provide an optimal solution as they are fast but easily getting stuck at the local optima as confirmed with past experience of researchers.

In recent decades, a wide variety of metaheuristic optimization methods such as genetic algorithm (GA) [4, 5], artificial immune system (AIS) [6, 7], particle swarm optimization (PSO) [8–16], differential evolution (DE) [17–19], gravitational search algorithm (GSA) [20], Tabu Search (TS) [21, 22], neural network (NN) [23, 24], evolutionary programming (EP) [25], bacterial foraging algorithm (BFA) [26], biogeography-based optimization (BBO) [27], and other population-based optimization algorithms [28–32] have been applied with success in solving the ED problems and been able to obtain better solutions compared to using conventional optimization methods.

Recently, a variant of PSO with guaranteed global convergence ability, quantum-behaved particle swarm optimization (QPSO) algorithm, is proposed by Sun et al. [33, 34]. QPSO outperforms PSO in global search ability and is a promising optimizer for complex problems [35–37]. QPSO demonstrates its superiority in solving ED problems comparing to other population-based optimization algorithms [38, 39]. Although various QPSO approaches have been successful in solving ED problems as reported in literature, they still lack the efficient mechanism to treat the constraints effectively [39]. The most commonly used method to handle constraints in ED problem with QPSO is the use of penalty functions [39, 40]. The simple implementation of combining the constraints with the objective function is the advantage of this approach. However, an additional tuning parameter called penalty factor is needed to penalize those solutions violating the constraints. It is rather difficult to choose the appropriate penalty factor for the penalty function approaches. A small penalty factor is not effective to handle the ED problem. Conversely, a large penalty factor will make the ED problem feasible but is distorting the solution space. As a result, it will converge to a weak local optimum. Recently, some heuristic constraint handling strategies have been proposed to modify infeasible solutions to satisfy the equality constraints, but their heavy computational requirement imposes a challenge for any evolutionary algorithms based on those heuristic strategies to find the global optimal solutions efficiently [41].

To overcome the existing deficiencies of QPSO for the ED problem, this paper proposes a double elitist breeding transposon QPSO (DEB-QPSO) algorithm based on our recent work on EB-QPSO [42] with the extension of an improved constraint handling technique and a cooperative update method for the pbests and gbest. The proposed approach makes use of two elitist breeding methods with transposon to enhance the diversity of the population in the QPSO mechanism. The transposon operators can improve the global search ability by preventing the premature convergence through increased diversity of the population as demonstrated in our previous work [42]. The current work is to extend our EB-QPSO technique to solve the ED problems. Moreover, one of the proposed elitist breeding schemes aims to assist the constraint handling while improving the diversity of the population. An improved dedicated efficient heuristic handling technique is proposed to manage the equality and inequality constraints of ED problems. The proposed algorithm is applied on three different widely used ED test problems and compared to various the state-of-the-art population-based optimization algorithms. The rest of this paper is organized as follows: the mathematical formulation of the ED problem is given in Section 2. Section 3 briefly describes the PSO, QPSO, and EB-QPSO algorithms that related to the proposed approach. The proposed DEB-QPSO algorithm for solving ED problems is presented in Section 4. Section 5 gives the implementation of the proposed algorithm for solving ED problems. Section 6 provides the case studies and results of the DEB-QPSO algorithm for three nonconvex ED problems and is compared to the state-of-the-art approaches from literature. Conclusion is given in Section 7.

2. Problem Formulation

The objective of ED is to reduce the operation cost of the system while fulfilling the load demand within the limit of constraints. The nonsmooth/nonconvex ED problem takes into account valve-point loading effects, prohibited operating zones and multifuel options along with system power demand, transmission loss, and operational limit constraints. The overall ED problem can be formulated as a nonlinear optimization programming problem:

$$\text{Minimize} \quad C_T = \sum_{i=1}^{Ng} F_i(P_i), \tag{1}$$

$$F_i(P_i) = \mathrm{a}_i + b_i P_i + c_i P_i^2, \tag{2}$$

where C_T, $F_i(P_i)$, and N_g are the total fuel cost, cost function of generator i, and the number of generators in the system, respectively; a_i, b_i, and c_i are the cost coefficients of the ith generator and P_i is the power output of the ith generator.

The generating units with multiple valves in steam turbines are available. The opening and closing of these valves may add the ripples in the cost function which makes the objective function highly nonlinear [5]. The cost function in (2) is modified as

$$F_i(P_i) = a_i + b_i P_i + c_i P_i^2 + \left| e_i \sin\left(f_i \times \left(P_{i,\min} - P_i\right)\right)\right|, \tag{3}$$

where $P_{i,\min}$ is the minimum output of the ith generator, e_i and f_i are two coefficients of the ith generator with valve-point loading effect.

ED problem is subjected to the following constraints:

(i) Power balance constraint

$$\sum_{i=1}^{N_g} P_i - P_L - P_D = 0, \tag{4}$$

where P_L and P_D are the total transmission network losses and the total load demand, respectively.

Normally, P_L is represented by way of Kron's loss formula [1] given as

$$P_L = \sum_{i=1}^{N_g}\sum_{j=1}^{N_g} P_i B_{ij} P_j + \sum_{i=1}^{N_g} B_{0i} P_i + B_{00}, \tag{5}$$

where B_{ij}, B_{i0}, and B_{00} are known as the loss coefficients determined by the situation of a specified power system.

(ii) Power output limit and amp rate limit constraint

Considering the inequality constraints of power output limit and ramp rate limit simultaneously, the generator operation constraint can be expressed as follows:

$$\max\left\{P_{i,\min}, P_i^0 - DR_i\right\} \le P_i \le \min\left\{P_{i,\max}, P_i^0 + UR_i\right\}, \tag{6}$$

where $P_{i,\min}$, $P_{i,\max}$, P_i^0, UR_i, and DR_i are the minimum output, maximum output, previous output power, the upramp limit, and downramp limit of the ith generator, respectively.

(iii) Prohibited operating zone constraint

The feasibility operation zones of a unit with prohibited operation zones lead to additional constraints on the unit operating range as follows:

$$P_i \in \begin{cases} P_i^{\min} \le P_i \le P_{i,1}^{l}, \\ P_{i,k-1}^{u} \le P_i \le P_{i,k}^{l}, \\ P_{i,n_{pz}}^{u} \le P_i \le P_i^{\max}, \\ \quad k = 2, 3, \ldots, n_{pz}, \end{cases} \tag{7}$$

where n_{gz}, $P_{i.k}^{l}$, and $P_{i.k}^{u}$ are the number of prohibited zones, the lower bound of the kth prohibited zone of the ith generator, and upper bound of the kth prohibited zone of the ith generator, respectively.

3. PSO, QPSO, and EB-QPSO

Particle swarm optimization (PSO) was proposed by Kennedy and Eberhart [34, 43] and is acknowledged as one of the most popular stochastic algorithms. In the past two decades, the PSO algorithm has undergone many improvements or modifications in an effort to compete more effectively on solving complicated problems [34, 43].

3.1. The Original PSO Algorithm. The PSO algorithm is inspired by the social behavior of bird flocking. Each particle is defined by a position vector $\mathbf{x} = (x_1, x_2, \ldots, x_D)$ which signifies a solution responsible for the exploration of the search space. Let N denote the swarm size and D be the dimensionality of the search space; during the search process, the position of each particle is evolved through the velocity and position equations:

$$v_i^{t+1} = wv_i^t + c_1 r_1 \left(\text{pbest}_i - \mathbf{x}_i^t\right) + c_2 r_2 \left(\text{gbest} - \mathbf{x}_i^t\right), \tag{8}$$

$$\mathbf{x}_i^{t+1} = v_i^{t+1} + \mathbf{x}_i^t, \tag{9}$$

where v_i^t and $\mathbf{x}_i^t$ are the velocity and position of the ith $(i = 1, 2, \ldots, N)$ particle in generation t. pbest_i, the personal best, is the previous position having the best objective function value of the ith particle. gbest called global best particle is the position of the best particle among all particles in the swarm. r_1 and r_2 are two uniformly distributed random numbers generated in the range $[0, 1]$. c_1 and c_2 are the learning factors called acceleration coefficients, and w is the inertia weight introduced into (8) by Shi and Eberhart [34, 43, 44] which is decreased linearly from 0.9 to 0.4 through the evolutionary process to enhance the capacity of exploring the solution space. Without loss of generality, for minimization, with the objective function, f, the pbest_i is updated according to

$$\text{pbest} = \begin{cases} \mathbf{x}_i^t & \text{if } f\left[\mathbf{x}_i^t\right] < f\left[p_i^{t-1}\right], \\ \text{pbest}_i^{t-1} & \text{if } f\left[\mathbf{x}_i^t\right] > f\left[p_i^{t-1}\right]. \end{cases} \tag{10}$$

Consequently, gbest is found by

$$\begin{aligned} \text{gbest}^t &= \text{pbest}_g^t, \\ \text{pbest}_g^t &= \arg\ \min_{1 \le i \le N} \left\{ f\left[\text{pbest}_i^t\right] \right\}. \end{aligned} \tag{11}$$

Although PSO converges fast and many attempts have been made to improve the performances of PSO [42, 43], it is prone to be trapped into local optima and not guaranteed to be global convergent as demonstrated in [45].

3.2. QPSO. A widely used PSO using alternative particle evolution formulae is QPSO. Inspired by the quantum mechanics, Sun et al. developed the quantum-behaved particle swarm optimization algorithm based on the trajectory analysis of the PSO, which is theoretically proved to be global convergent [33, 34, 46]. In the evolutionary process, the position of each particle is updated with the following rules:

$$\begin{aligned} \mathbf{x}_i^{t+1} &= p_i^t + \alpha \left|\mathbf{x}_i^t - \text{mbest}^t\right| \ln\left(\frac{1}{u_i^t}\right), \quad \text{if randv} \ge 0.5, \\ \mathbf{x}_i^{t+1} &= p_i^t - \alpha \left|\mathbf{x}_i^t - \text{mbest}^t\right| \ln\left(\frac{1}{u_i^t}\right), \quad \text{if randv} < 0.5, \end{aligned} \tag{12}$$

where p_i^t, mbest^t, and α are the local attractor of the particle in iteration t, the mean best position in iteration t, and the contraction-expansion coefficient, respectively; both u_i^t and randv are random numbers generated using the uniform probability distribution functions in the range of $[0, 1]$. The local attractor is defined as

$$p_i^t = \varphi^t \times \text{pbest}_i^t + \left(1 - \varphi^t\right) \times \text{gbest}^t, \tag{13}$$

where φ^t is a uniformly distributed random parameter chosen within the interval $[0, 1]$, $\text{pbest}_\text{i}^\text{t}$ and gbest^t are the personal best of the ith particle and global best of the swarm, respectively. The mean best position is defined as the mean of the pbest position of all particles:

$$\text{mbest}^t = \frac{1}{N} \sum_{i=1}^{N} \text{pbest}_i^t. \tag{14}$$

α is the contraction-expansion coefficient used to control convergence rate of QPSO, which is usually adjusted with a time-varying decreasing method [47] defined as follows:

$$\alpha = \frac{\alpha_1 + (T - t) \times (\alpha_0 - \alpha_1)}{T}, \tag{15}$$

where T is the maximum iteration number and t is the current search iteration number. α_0 and α_1 are the initial and final values of α, respectively.

QPSO is theoretically guaranteed the global convergence of the algorithm; however, upon the assumption of infinite number of search iteration, such requirement is impractical in solving complex engineering problems like ED problems.

3.3. EB-QPSO. To alleviate the problem of QPOS, numerous strategies have been proposed in recent literatures to improve the exploration efficiency and quality of solution. These strategies can be classified as improvements by parameter selection, control swarm diversity, cooperative methods, using probability distribution function, novel search methods, and hybrid methods [32, 46, 48, 49, 51–57] Recently, a EB-QPSO algorithm based on transposon was proposed [42, 58]. The basic idea of the approach is to make better use of the elitists consisting of the pbests and

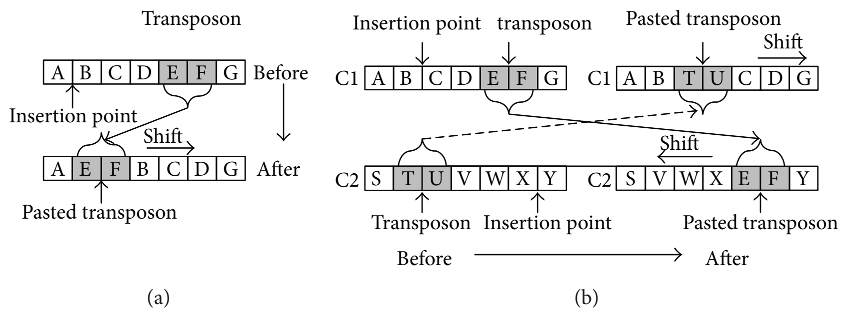

Figure 1: Cut-and-paste transposon operator. (a) Cut-and-paste in same chromosome. (b) Cut-and-paste in different chromosomes.

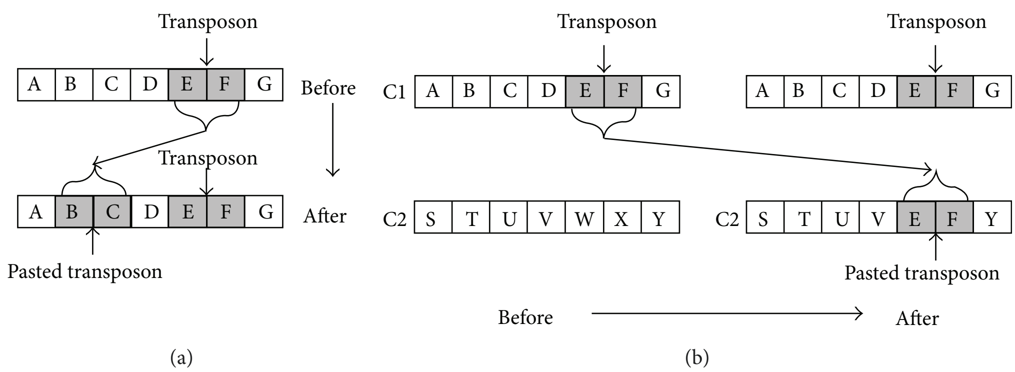

Figure 2: Copy-and-paste transposon operator. (a) Copy-and-paste in same chromosome. (b) Copy-and-paste in different chromosomes.

gbest in aiding to aggrandize the diversity of the swarm that is essential to the exploration and exploitation for the search of the global optima [59, 60].

An elitist exploration strategy, namely, elitist transposon breeding, is incorporated with the basic evolutionary processes of QPSO in the EB-QPSO algorithm. In the elitist transposon breeding scheme, an elitist pool consisting of pb ests and gbest is constructed. New particles are generated from the elitist pool with the transposon operators, having the ability to enhance the diversity of solutions, to explore the elitist memory and extract some more potential essences from the elitist individuals and thus to improve the search efficiency. Moreover, the update of elitists with the new-bred better-fitted individuals will provide a more efficient and precise search guidance for the swarm.

Transposon operators were firstly proposed by Tang et al. [50] and mainly used in multiobjective evolutionary algorithms and applied in population-based optimization algorithm in our works [42, 58, 61]. A transposon is made of consecutive genes located in the randomly assigned position in each chromosome while the transposon operators are lateral movement operations that happen in one chromosome or between different ones. In general, there exist two types of transposon operators, cut-and-paste and copy-and-paste, which are shown in Figures 1 and 2. The transposon operations conducted within an individual chromosome or on a different chromosome are chosen randomly. Moreover, the size of each transposon can be greater than one and is decided by a parameter called jumping

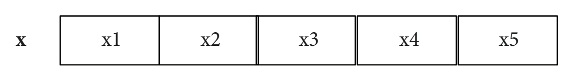

Figure 3: An example of the particle is represented as a chromosome.

percentage while the number of transposons is also a predefined parameter. Another parameter, the jumping rate, is assigned to determine the probability of the activation of transposon operations.

As demonstrated in Figure 3, each particle which can be regarded as a chromosome consists of the same number of genes as the size of its position vector and each gene holds a real number of the corresponding decision variable.

Comparing with other the state-of-the-art PSO and QPSO variants, EB-QPSO performs more competitively in solving unconstrained optimization problems in terms of better global search capability and faster convergence rate as demonstrated in our recent work [42, 58] and has been applied in solving practical problems like cancer gene classification [62, 63].

4. DEB-QPSO for ED Problems

In this section, we will propose a DEB-QPSO algorithm based on our recent work of using elitist breeding to solve the ED problem. A double elitist breeding strategy is designed to improve the search efficiency and manage the constraint requirement of the complex ED problems. In addition, an improved heuristic technique infeasible particle

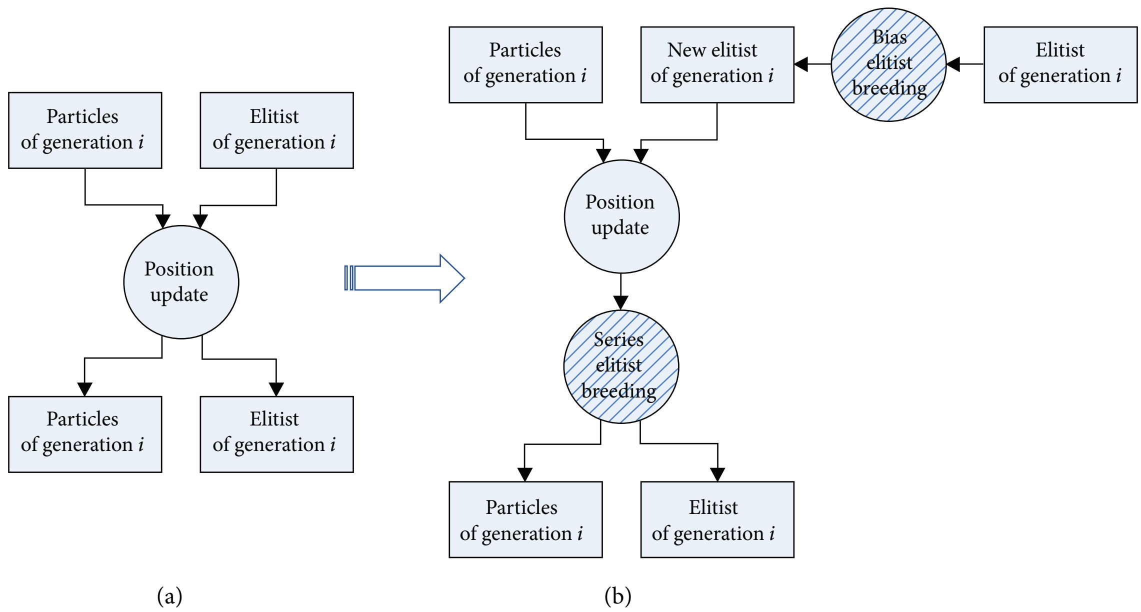

FIGURE 4: Integration of the double elitist breeding into the evolutionary processes of QPSO.

repositioning is proposed to treat the equality and inequality constraints of ED problems efficiently. Moreover, a novel update method for the pbests and gbest of the swarm is proposed to cooperate with the constraint handling technique. In the rest of the section, the double elitist breeding strategy catered for solving constrained optimization problem is described and is followed by how to use the proposed approach to solve ED problem.

4.1. Elitist Breeding Strategies. The elitists have a major effect on the exploration behavior of the swarm, thus impinging the exploration performance. Obviously, making good use of the elitists is beneficial to promote the exploration for optimal solution. In DEB-QPSO, an elitist pool, epool, consisting of pbests and gbest is constructed. Two elitist breeding strategies are used to improve the global search ability of the algorithm. Generally, the main idea of breeding is to make good use of the elitists to generate new particles through transposon operations. There are two types of elitist breeding operations used: the bias elitist breeding and the series elitist breeding differentiated according to their execution sequences in the proposed DEB-QPSO algorithm and the selection of the type of elitists for breeding. Figure 4 shows the relationship between the elitist breeding and the normal algorithmic operation of QPSO. In Figure 4(a), it gives the relationship between particles and the elitists of the swarm in the original QPSO while Figure 4(b) illustrates the integration of two elitist breeding with the exploration operations of QPSO.

4.2. Bias Elitist Breeding. The bias elitist breeding aims at improving the search efficiency of the algorithm through promoting the particle diversity and assists the constraint handling capability. The bias elitist breeding is not executed on every search cycle but once at the interval of every λ iteration. It places impact to particles by substituting the elitist individuals in memory with better-fitted new-bred individuals. Breeding operation will be conducted on every particle in the swarm. New particles will be generated through the transposon with an elitist selected randomly from the original elitist pool made up of the pbests and the gbest of the swarm. Only the feasible elitists are selected for transposon. If the newly generated particle has a better fitness evaluated by the objective function, it will become the personal best of that particle and be kept in the elitist pool; otherwise, the original personal best of that particle will be kept in the elitist pool. In other words, the pbest of each particle and the gbest will be substituted by the corresponding better solutions in the newly generated subswarm. Such elitist breeding scheme is biased towards the generation of elitists in the feasible zone. The idea is illustrated in Figure 5.

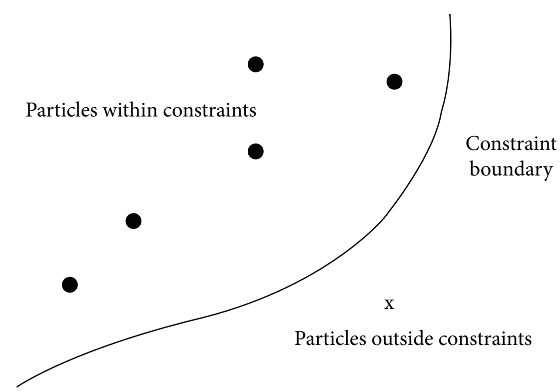

FIGURE 5: Particles participated in transposon: an infeasible particle undergoes transposon with a feasible elitist randomly selected from the elitist pool.

```
Procedure of Bias elitist breeding
for i = 1 to swarm size N
    epool_eb = transposon (epool, particle[i];);
    evaluate (epool_eb);
    if f (epool_eb[i]) < f (pbest[i]);
        pbest [i] = epool_eb [i];
    endif
endfor
```

Pseudocode 1. The pseudocodes for bias elitist breeding operation.

For an infeasible particle, located outside the constraint boundary, undergoing the bias elitist breeding with a randomly selected elitist located within the constraint boundary, it is more likely to have a feasible particle generated for the infeasible particle through the transposon with a feasible elitist chosen from epool. In such a way, the bias elitist breeding not only promotes the diversity of the swarm but also is able to assist the constraint handling capability in order to enhance the global search exploration in return. The pseudocodes of bias elitist breeding operation are given in Pseudocode 1.

4.3. Series Elitist Breeding. The series elitist breeding is performed right after the position update procedure of each particle. It will be conducted for every search cycle. Different from the bias elitist breeding, the newly updated particle undergoes transposon with an elitist selected randomly among all elitists from epool to generate a trial vector so as to effectively exploit and explore the search space. The purpose of the series elitist breeding is to promote the diversity of the swarm. The epool is updated accordingly with the newly generated particle. The procedure of series elitist breeding is the same as the bias one, but all the elitists participated in the random selection of transposon.

4.4. Constraint Handling with Particle Repositioning. To keep the particle search in the correct direction, it is essential to satisfy the equality and inequality constraints simultaneously. The standard penalty function method is not effective in handling the equality constraints. Researches have focused on the heuristic strategies to modify infeasible solutions to satisfy the equality constraints. Emphasis is on adjusting the value of the elements in each solution in every search iteration meticulously to satisfy the constraints [9, 64, 65]; obviously, such strategies are computationally expensive and inefficient in finding the optimum solution. As mentioned, the infeasible particle which resulted from the position update may have a chance to migrate to the feasible zone through the series elitist breeding. Therefore, an improved heuristic technique based on the heuristic strategy presented in [64] and combined with the complementary update method for pbests and gbest for infeasible particle repositioning to satisfy the constraints is proposed. If the infeasible particle cannot be corrected through the series elitist breeding, it will be repositioned to a new location satisfying the constraints through the method described as follows:

Step 1. If necessary, to satisfy the inequality constraint of power output limit given by (6), modify the value of the jth element in the ith individual as follows:

$$P_{i,j} = \begin{cases} P_{i,j}, & \text{if } P_{j,\min}' \le P_{i,j} \le P_{j,\max}', \\ P_{j,\min}', & \text{if } P_{i,j} < P_{j,\min}', \\ P_{j,\max}', & \text{if } P_{i,j} > P_{j,\max}'. \end{cases} \quad (16)$$

Step 2. If any element in the ith individual falls within its kth prohibited zone, the value of this element will be adjusted to satisfy the inequality constraint given by (7) as follows:

$$P_{i,j} = \begin{cases} P_{i,k}^{l}, & \text{if } \left(P_{j,k}^{u} - P_{i,j}\right) > \left(P_{i,j} - P_{j,k}^{l}\right), \\ P_{i,k}^{u}, & \text{if } \left(P_{j,k}^{u} - P_{i,j}\right) \le \left(P_{i,j} - P_{j,k}^{l}\right). \end{cases} \quad (17)$$

Step 3. Set $P_i' = P_i$ and set each of the repairing flag RF_j of element j to 0.

Step 4. Calculate P_L (i.e., transmission network loss) using the coefficient formula given by (5).

Step 5. Calculate the equality constraint violation using the following formula:

$$P_{cv} = \sum_{i=1}^{N_g} P_i - P_L - P_D. \quad (18)$$

If the absolute value of P_{cv} is less than the predefined demand tolerance ε, then go to Step 8; otherwise, go to Step 6.

Step 6. From the current individual, randomly select an element j with $RF_j = 0$ (it means the element was not selected so far) and set $RF_j = 1$; if all the elements have been selected before, then go to Step 8.

Step 7. Adjust the value of element j to satisfy the power balance constraint given by (4) as follows:

$$P_{i,j} = \begin{cases} P_{i,j} - \min\left(P_{cv}, \left(P_{i,j} - P_{j,\min}'\right) \times \text{randv}\right), & \text{if } P_{cv} > 0, \\ P_{i,j} + \max\left(P_{cv}, \left(P_{j,\max}' - P_{i,j}\right) \times \text{randv}\right), & \text{if } P_{cv} \le 0, \end{cases} \quad (19)$$

where randv is a random number uniformly distributed on $[0, 1]$.
If element j falls within its kth prohibited zone, the value of this element will be adjusted to satisfy the prohibited zone constraint as Step 2. Repeat the actions listed in Step 5.

Step 8. If $\max_{\forall j}\{|P_{i,j} - P_{i,j}'|\} < \delta$, then go to Step 9; otherwise, go to Step 3. Here, δ is a temporal adaptive solution convergence tolerance and can be calculated as follows:

$$\delta = \frac{\delta_1 + (T - t) \times (\delta_0 - \delta_1)}{T}, \quad (20)$$

where T is the maximum iteration number and t is the current search iteration number; δ_0 and δ_1 are the initial and final values of δ, respectively.

Step 9. End.

The improved constraint handling method lays on the time-varying δ together with the cooperative update method for pbests and gbest described below to reduce the computational cost and enhance the search efficiency. To handle the constraints effectively, the solution convergence tolerance δ is normally set to a small fixed value in those approaches adopting the heuristic technique as in [64] and thereby leads to heavy computational cost. Apparently, it is unnecessary to use the meticulous method to handle the constraints for each of the individual particles in the whole search procedure. Hence, the time-varying decrement δ is proposed to improve the efficiency of constraint handling. In the early iterations, more infeasible solutions are allowed to appear so as to increase the diversity of the search. Combined with the elitist breeding strategies, the exploration ability of the solution algorithm is thereby enhanced. Conversely, in the later iterations, a stricter criterion is beneficial for generating the feasible solutions so as to enhance the convergence speed of the search.

4.5. pbests *and* gbest *Update.* It is clear that the equality constraint is not guaranteed to be satisfied through the heuristic repairing procedure. To cooperate with the constraint handling method, a novel update method for pbests and gb est is introduced following the constraint handling step. The method is as shown below.

Step 1. Calculate P_L using coefficient (5).

Step 2. Calculate the absolute value of equality constraint violation (called constr) according to the formula as follows:

$$\text{constr} = abs\left(\sum_{i=1}^{N_g} P_i - P_L - P_D\right). \quad (21)$$

If constr is less than the demand tolerance ε, then set constr to 0; otherwise, calculate the probability (called prob) to accept the infeasible solutions as the elitist (pbest and gbest) as follows:

$$\text{prob} = \frac{t}{T}, \quad (22)$$

where T is the maximum iteration number and t is the current search iteration number.Then recontribute the value of constr with the equation below:

$$\text{constr} = \begin{cases} 0, & \text{if randv} > \text{prob}, \\ \text{constr}, & \text{if randv} \le \text{prob}, \end{cases} \quad (23)$$

where randv is a random value chosen uniformly within the interval $[0, 1]$.

Step 3. Update pbest of each particle with the following rules:

(1) If constr of the current particle is less than constr of its pbest, set the current particle as its updated pbest.

(2) If constr of the current particle is equal to constr of its pbest and the objective function value of the current particle is less than the objective function value of its pbest, set the current particle as its updated pbest.

(3) Otherwise, keep the pbest of the current particle unchanged.

Step 4. Update gbest with the following rule: select the pbest with the smallest constr as the new gbest of the current iteration; if more than one pbest have the same smallest value of constr, use the pbest with the smallest objective function value as the new pbest.

Step 5. Stop the pbest and gbest update procedure.

Obviously, the probability of accepting infeasible solution as elitists is decreasing with respect to the search iteration. Such scheme aims to enhance the solution diversity in the early search iterations and promotes the search efficiency in the feasible area in the later search stage.

5. Implementation of the DEB-QPSO Algorithm

The decision variables in ED problems are the real power output of the units in the systems. A particle is a set of real number elements corresponding to the units' output represented particle by the position vector $\mathbf{x} = (x_1, x_2, \dots, x_D)$ where D is the number of units, which signifies a solution in the search responsible for the exploration of the search space. The proposed DEB-QPSO algorithm is summarized as in Table 1:

5.1. Initialization of Population. According to (6), the minimum and maximum outputs of the ith generator are defined as follows:

$$\begin{aligned} P_{i,\min}' &= \max\left\{P_{i,\min}, P_i^0 - DR_i\right\}, \\ P_{i,\max}' &= \min\left\{P_{i,\max}, P_i^0 + UR_i\right\}. \end{aligned} \quad (24)$$

Table 1: The process of DEB-QPSO.

Step 1. Randomly initialize the position of a population while satisfying the constraints.
Step 2. Update the positions of particles according to (12).
Step 3. Perform the series elitist breeding operation.
Step 4. If necessary, repair the position of particles to satisfy the constraints.
Step 5. Update pbests and gbest.
Step 6. Perform the bias elitist breeding operation when the criterion is met.
Step 7. Go to step 2 until the stopping criterion is met.

Therefore, a set of particles is initialized randomly as follows:

$$P_{i,j}^{\text{init}} = P_{j,\min}{}' + \text{randv}_{i,j} \times \left(P_{j,\max}{}' - P_{j,\min}{}' \right), \tag{25}$$

where $\text{randv}_{i,j}$ is the random number generated using the uniform probability distribution function in the range $[0, 1]$.

5.2. Position Update. For each particle, the position update is conducted according to the QPSO algorithm signified by (8).

5.3. Series Elitist Breeding. The series elitist breeding is performed on each particle with the elitist randomly selected from epool. New subswarm is generated. Since the resulting subswarm is not always guaranteed to satisfy the constraints, the improved particle repositioning is conducted.

5.4. Update of pbests *and* gbest. The pbests of each particle at iteration $t + 1$ is updated according to the described updating procedures. Obviously, gbest is set as the best evaluated position among all the pbests.

5.5. Bias Elitist Breeding. To bias the swarm towards the feasible region, each particle is going through the bias elitist breeding for every λ iteration.

5.6. Stopping Criteria. The proposed DEB-QPSO algorithm is terminated if the iteration reaches a predefined maximum number.

6. Case Studies and Results

The proposed DEB-QPSO approach is applied to three different widely used test case power systems posing different difficulties to optimization algorithms: (i) a 15-unit system with prohibited operating zones, ramp rate limits, and transmission network losses, (ii) a 40-unit system with valve-point effects, prohibited operating zones, and ramp rate limits, and (ii) a 140-unit Korean power system with valve-point effects, prohibited operating zones, and ramp rate limits.

To evaluate the solution quality and robustness fairly, 50 independent runs are conducted for each case. The setup for the proposed DEB-QPSO algorithm is as follows:

Table 2: Best solution obtained using the proposed DEB-QPSO (case I).

Bus number	Output	Bus number	Output
P_1 (MW)	454.9997	P_9 (MW)	60.5887
P_2 (MW)	380	P_{10} (MW)	159.9958
P_3 (MW)	129.9998	P_{11} (MW)	80
P_4 (MW)	130	P_{12} (MW)	79.9992
P_5 (MW)	170	P_{13} (MW)	25
P_6 (MW)	459.9998	P_{14} (MW)	15
P_7 (MW)	430	P_{15} (MW)	15
P_8 (MW)	68.7781		
ΣP_i (MW)		2660. 3611	
P_{loss} (MW)		30.3611	
F_{total} (\$/h)		32701.1557	

(1) The maximum object function evaluation numbers (FEs) are 6000, 20,000, and 20,000 for test systems 1, 2, and 3, respectively.

(2) The population size is 20.

(3) The contraction-expansion coefficient α decreases linearly from 0.6 to 0.5.

(4) CR in serial elitist breeding is fixed at 0.6.

(5) λ is 2.

(6) ε is set to $1 \times 10^{-10} \times$ (total power load demand), and it is much stricter than the parameter setting in [24].

(7) δ_0 and δ_1 are set at 1×10^{-2} and 1×10^{-3}, respectively.

To make a direct comparison, simulation experiments of the three test systems are conducted with the QPSO and its two variants, QPSO-DM(1) and QPSO-DM(2), proposed to solve the ED problem in [39] with the same population size and FEs as set in the DEB-QPSO algorithm. In addition, to assess the efficiency of the proposed elitist breeding strategies and the proposed constraint handling method, the proposed constraint handling and the one proposed in IPSO [64] are applied, respectively, to the original QPSO under the same parameter settings as in the DEB-QPSO algorithm to form two other algorithms, namely, QPSO-EDP(1) and QPSO-EDP(2), for comparison. Furthermore, the computational results found by some other the state-of-the-art methods reported in the literature are compared as well. All the simulations are conducted under the computing environment with a notebook PC, 4 GB RAM, Core i3 2.13GHz CPU clock speed, Microsoft Windows 7, and MATLAB 2010a.

For each test case, there are two tables showing the simulation results: one for the best fuel cost values obtained by DEB-QPSO with the corresponding generation power outputs after 50 independent runs (Tables 2, 3, and 4 for case I, case II, and case III, resp.) and the other for the result summary of the best, average, and

Table 3: Best solution obtained using the proposed DEB-QPSO (case II).

Bus number	Output	Bus number	Output	Bus number	Output	Bus number	Output
P_1 (MW)	110.7998	P_{11} (MW)	168.7821	P_{21} (MW)	523.2794	P_{31} (MW)	190.0000
P_2 (MW)	110.7998	P_{12} (MW)	168.0552	P_{22} (MW)	523.2794	P_{32} (MW)	190.0000
P_3 (MW)	97.3999	P_{13} (MW)	214.7598	P_{23} (MW)	523.2794	P_{33} (MW)	190.0000
P_4 (MW)	179.7331	P_{14} (MW)	400.0000	P_{24} (MW)	523.2794	P_{34} (MW)	164.7998
P_5 (MW)	87.7999	P_{15} (MW)	394.2794	P_{25} (MW)	523.2794	P_{35} (MW)	164.7998
P_6 (MW)	140.0000	P_{16} (MW)	304.5196	P_{26} (MW)	523.2794	P_{36} (MW)	164.7998
P_7 (MW)	259.5997	P_{17} (MW)	489.2794	P_{27} (MW)	10.0000	P_{37} (MW)	110.0000
P_8 (MW)	284.5997	P_{18} (MW)	489.2794	P_{28} (MW)	10.0000	P_{38} (MW)	110.0000
P_9 (MW)	284.5997	P_{19} (MW)	511.2794	P_{29} (MW)	10.0000	P_{39} (MW)	110.0000
P_{10} (MW)	130.0000	P_{20} (MW)	511.2794	P_{30} (MW)	87.7999	P_{40} (MW)	511.2794
ΣP_i (MW)		10500.0000				F_{total} ($/h)	121472.7668

Table 4: Best solution obtained for the 140-unit system using DEB-QPSO (case III).

Bus number	Output	Bus number	Output	Bus number	Output	Bus number	Output	Bus number	Output
P_1 (MW)	117.2970	P_{29} (MW)	500.9966	P_{57} (MW)	103.8112	P_{85} (MW)	115.0175	P_{113} (MW)	94.1953
P_2 (MW)	188.7133	P_{30} (MW)	501	P_{58} (MW)	198.0444	P_{86} (MW)	207.0494	P_{114} (MW)	94.0038
P_3 (MW)	189.8084	P_{31} (MW)	506	P_{59} (MW)	311.5666	P_{87} (MW)	207.0172	P_{115} (MW)	244
P_4 (MW)	190	P_{32} (MW)	505.9437	P_{60} (MW)	285.0454	P_{88} (MW)	177.3728	P_{116} (MW)	244.2227
P_5 (MW)	168.5109	P_{33} (MW)	505.9441	P_{61} (MW)	163.5463	P_{89} (MW)	176.1600	P_{117} (MW)	244.0369
P_6 (MW)	188.1695	P_{34} (MW)	505.9737	P_{62} (MW)	95.4018	P_{90} (MW)	175.0217	P_{118} (MW)	95.1762
P_7 (MW)	489.9914	P_{35} (MW)	499.8883	P_{63} (MW)	160.7681	P_{91} (MW)	175.1082	P_{119} (MW)	95.0278
P_8 (MW)	489.9733	P_{36} (MW)	499.9484	P_{64} (MW)	165.9874	P_{92} (MW)	579.9638	P_{120} (MW)	116.0736
P_9 (MW)	495.9999	P_{37} (MW)	240.9973	P_{65} (MW)	487.1185	P_{93} (MW)	644.9552	P_{121} (MW)	175.0168
P_{10} (MW)	495.9738	P_{38} (MW)	240.9804	P_{66} (MW)	197.9387	P_{94} (MW)	983.9236	P_{122} (MW)	2.1429
P_{11} (MW)	495.9997	P_{39} (MW)	773.9365	P_{67} (MW)	488.8266	P_{95} (MW)	978	P_{123} (MW)	4.0436
P_{12} (MW)	495.9950	P_{40} (MW)	768.9871	P_{68} (MW)	490	P_{96} (MW)	681.9999	P_{124} (MW)	15
P_{13} (MW)	505.9779	P_{41} (MW)	3.0000	P_{69} (MW)	130.1504	P_{97} (MW)	719.9820	P_{125} (MW)	9.0005
P_{14} (MW)	508.9738	P_{42} (MW)	3.0091	P_{70} (MW)	234.8483	P_{98} (MW)	718	P_{126} (MW)	12.0286
P_{15} (MW)	505.9993	P_{43} (MW)	249.7749	P_{71} (MW)	137.3286	P_{99} (MW)	719.9486	P_{127} (MW)	10.0056
P_{16} (MW)	504.9864	P_{44} (MW)	242.9736	P_{72} (MW)	325.2732	P_{100} (MW)	963.9992	P_{128} (MW)	112.1078
P_{17} (MW)	505.9858	P_{45} (MW)	249.9237	P_{73} (MW)	195.6645	P_{101} (MW)	957.9839	P_{129} (MW)	4.0005
P_{18} (MW)	505.8171	P_{46} (MW)	249.3941	P_{74} (MW)	175.3521	P_{102} (MW)	1006.9750	P_{130} (MW)	5.0025
P_{19} (MW)	504.8906	P_{47} (MW)	244.7004	P_{75} (MW)	186.0201	P_{103} (MW)	1005.9920	P_{131} (MW)	5
P_{20} (MW)	504.9952	P_{48} (MW)	248.0422	P_{76} (MW)	175.1273	P_{104} (MW)	1012.9910	P_{132} (MW)	50.3038
P_{21} (MW)	504.9956	P_{49} (MW)	249.8546	P_{77} (MW)	175.5258	P_{105} (MW)	1019.9920	P_{133} (MW)	5.0258
P_{22} (MW)	504.9881	P_{50} (MW)	249.7768	P_{78} (MW)	330.9662	P_{106} (MW)	953.9933	P_{134} (MW)	42.0002
P_{23} (MW)	504.9728	P_{51} (MW)	169.2291	P_{79} (MW)	530.8824	P_{107} (MW)	952	P_{135} (MW)	42.0500
P_{24} (MW)	504.9863	P_{52} (MW)	165.0001	P_{80} (MW)	530.9256	P_{108} (MW)	1006	P_{136} (MW)	41.0118
P_{25} (MW)	536.9815	P_{53} (MW)	165.7448	P_{81} (MW)	366.5174	P_{109} (MW)	1012.9980	P_{137} (MW)	17.0102
P_{26} (MW)	536.7416	P_{54} (MW)	165.5105	P_{82} (MW)	56.01368	P_{110} (MW)	1021	P_{138} (MW)	7
P_{27} (MW)	548.9839	P_{55} (MW)	180.0231	P_{83} (MW)	115.1456	P_{111} (MW)	1014.9750	P_{139} (MW)	7.1937
P_{28} (MW)	548.7910	P_{56} (MW)	180	P_{84} (MW)	115.0067	P_{112} (MW)	94.1349	P_{140} (MW)	26.9094
ΣP_i (MW)				49342.0000		F_{total} ($/h)		1559989.4523	

worst cost found by the proposed DEB-QPSO and its FEs together with the corresponding result of the other state-of-the-art methods in literatures (Tables 5, 6, and 7 for case I, case II, and case III, resp.). In addition, to access the effectiveness of the proposed constraint handling method in reducing the computational time, the average

Table 5: Results obtained by optimization methods (case I).

Methods	Best cost ($/h)	Worst cost ($/h)	Mean cost ($/h)	FEs	CPU time (s)
PSO [11]	32858.00	33331.00	33105.00	20,000	/
GA [11]	33113.00	33337.00	33228.00	20,000	/
SA-PSO [12]	32708.00	32789.00	32732.00	20,000	/
IPSO [13]	32,709	—	32784.5	10,000	/
DSPSO-TSA [14]	32715.06	32730.39	32724.63	6000	/
MTS [22]	32716.87	32796.13	32767.4	100,000	/
IA_EDP [7]	**32698.20***	32823.78	32750.22	20,000	/
QPSO [39]	33014.21	33342.30	33148.77	6000	/
QPSO-DM(1) [39]	32970.85	33406.27	33163.26	6000	/
QPSO-DM(2) [39]	32927.93	33353.23	33156.34	6000	/
QPSO-EDP(1)	32707.99	32718.39	32712.02	6000	4.27
QPSO-EDP(2)	32710.14	32719.27	32716.94	6000	6.35
DEB-QPSO	**32701.16**	**32701.18**	**32701.17**	6000	4.68

*In the case of IA_EDP, power balance constraint is not satisfied.

Table 6: Results obtained by optimization methods (case II).

Methods	Best cost ($/h)	Worst cost ($/h)	Mean cost ($/h)	FEs	CPU time (s)
NAPSO [15]	121491.0662	121491.5261	121491.2756	100,000	/
FAPSO [15]	122261.3706	122597.5196	122471.0751	100,000	/
PSO [15]	124875.8523	125368.9204	125162.7011	100,000	/
IABC [16]	121491.2751	121582.3865	121539.4175	100,000	/
IABC-LS [16]	121488.7636	121582.2525	121526.0333	100,000	/
QPSO [39]	126615.4498	128679.5909	128621.1241	20,000	/
QPSO-DM(1) [39]	131291.9433	131586.1562	131368.4071	20,000	/
QPSO-DM(2) [39]	131318.4412	133393.5961	131739.5250	20,000	/
QPSO-EDP(1)	121681.46	121708.01	121693.42	20,000	5.38
QPSO-EDP(2)	121875.23	121907.26	121895.49	20,000	7.96
DEB-QPSO	**121472.77**	**121483.67**	**121477.52**	20,000	6.15

Table 7: Results obtained by optimization methods (case III).

Methods	Best cost ($/h)	Worst cost ($/h)	Mean cost ($/h)	FEs	CPU time (s)
IPSO [64]	1657962.73	1657962.73	1657962.73	300,000	/
CQGSO [40]	1657962.72	1657962.77	1657962.74	120,000	/
DEL [18]	1657962.71	—	1658001.70	225,000	/
DE [19]	1566264.99	1566308.37	1566285.56	250,000	/
IDE [19]	1564648.66	1564682.73	1564663.54	250,000	/
QPSO [39]	1795570.26	1896439.75	1837102.23	20,000	/
QPSO-DM(1) [39]	1826727.13	1926994.21	1882621.38	20,000	/
QPSO-DM(2) [39]	1836848.28	1925476.87	1883407.62	20,000	/
QPSO-EDP(1)	1562612.57	1564828.72	1564105.64	20,000	7.21
QPSO-EDP(2)	1563052.83	1565314.23	1564482.81	20,000	10.39
DEB-QPSO	**1559989.45**	**1560739.23**	**1560215.43**	20,000	8.48

CPU time required by the DEB-QPSO, QPSO-EDP(1), and QPSO-EDP(2) algorithms are listed in Tables 5, 6, and 7 for case I, case II, and case III, respectively.

Moreover, the convergence characteristics of the median results obtained in the 50 runs with the proposed DEB-QPSO algorithm and the compared QPSO-EDP(1) and QPSO-

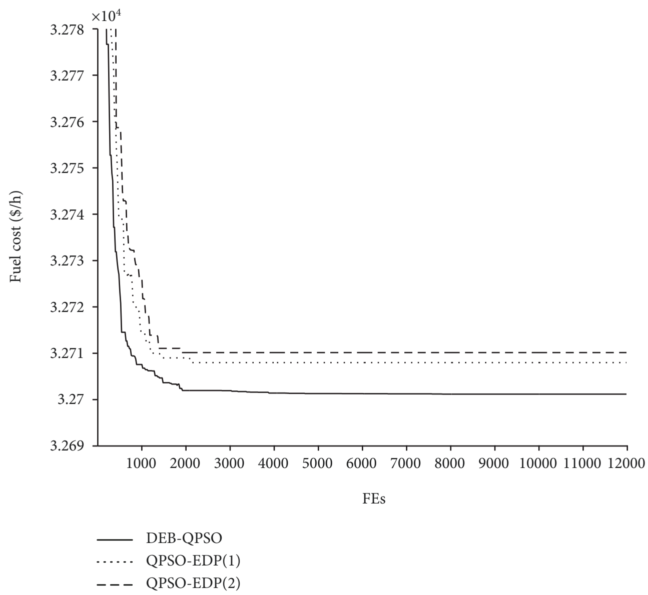

Figure 6: Convergence properties for case I.

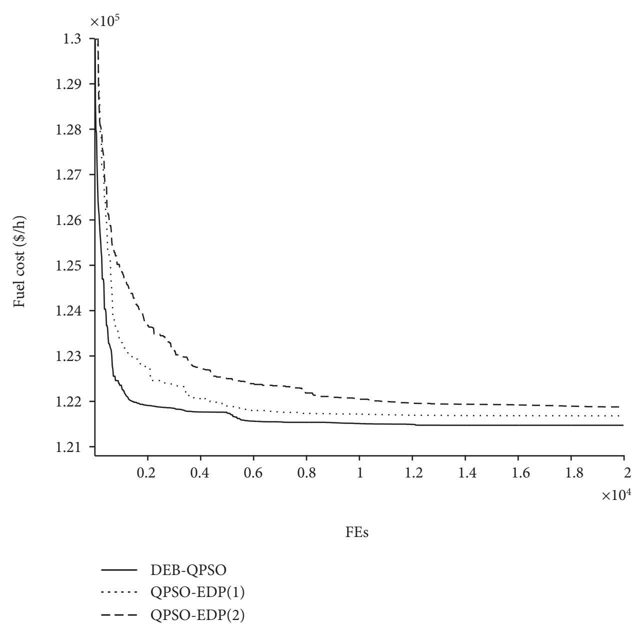

Figure 7: Convergence properties for case II.

EDP(2) algorithms for the three test systems are illustrated in Figures 6, 7, and 8, respectively.

6.1. Case I: 15-Unit System. The test system consists of 15 generating units. The expected load demand is 2630 MW. The system parameters and the B coefficients can be found in [10]. The main difficulties of this system for any optimization algorithm are the nonlinear and noncontinuous decision space and the power balance constraint with network transmission losses.

From Table 5, it clearly shows that the proposed method obtains the best result with lowest FEs than other techniques

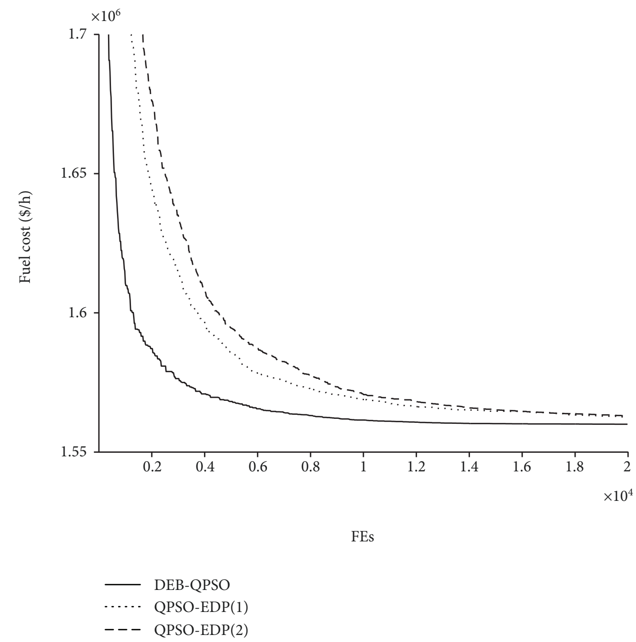

Figure 8: Convergence properties for case III.

except for IA_EDP [7] in the 15-generating-unit test systems. However, it should be noted that the exact power loss computed from the best solution found by IA_EDP [7] is actually 30.2825 instead of 30.0187 as reported in the corresponding literature, which shows that the total generated power of the schedule is much less than the total load demand plus its total line loss; obviously, the best solution reported in [7] is in fact infeasible. It can be concluded that the elitist breeding strategy of the proposed DEB-QPSO algorithm is not only capable of locating the optimal solution but also capable of being computationally effective as fewer functional evaluations are required comparing to other algorithms in the 15-generating-unit test system.

Moreover, the robustness of DEB-QPSO is confirmed with the evidence that the difference between worst cost and the best cost obtained by DEB-QPSO is no more than 0.01% of the best cost. The impact of the two key components of DEB-QPSO towards the search of optimal solution is also revealed individually from the simulation results. On one hand, it can be concluded from the results obtained by DEB-QPSO and QPSO-EDP(1) that the proposed elitist breeding strategies are beneficial for improving the global search capability of QPSO since QPSO-EDP(1) employed the same constraint handling as in the proposed DEB-QPSO. On the other hand, the efficiency of the proposed constraint handling method is validated not only with the results obtained by QPSO-EDP(1), which are better than those obtained by QPSO-EDP(2) using the IPSO's constraint handling method in [64], but also required less computational time to locate the solutions. Furthermore, the extra processing for elitist breeding is computational effective as the DEB-QPSO only increases less than 10% of the computational time when comparing with QPSO-EDP(1).

Besides, the simulation results also reveal that algorithms with heuristic constraint handling such as QPSO-EDP(1), QPSO-EDP(2), IPSO, and the proposed DEB-QPSO have better performance and computational effectiveness than those using penalty function for constraint handling.

It can be observed from Figure 6 that the DEB-QPSO algorithm converges to the result very close to the final optimal solution in early iterations and has a better convergence property than QPSO-EDP(1) and QPSO-EDP(2) algorithms in test case I.

6.2. Case II: 40-Unit System. The test system consists of 40 thermal units and 5 of which exhibit prohibited zones. The transmission losses are not considered and the expected load demand is 10,500 MW. All the generators in this system are subjected to valve-point effects resulting in a solution space with multiple minima. Because of the large dimension and multiple minima, it is hard to locate the global minimum. The fuel cost function coefficients and active power generation limits for this system can be obtained from Table 7 in reference [15] and the prohibited operation zones from Table 12 in reference [15].

It can be observed from Table 4 that the DEB-QPSO algorithm has generated very satisfactory stable solutions in the 40-generating-unit test system. The DEB-QPSO algorithm is able to obtain the best costs among the compared approaches with the lowest FEs; besides, the best cost and the worst cost are all within ±0.01% of the cost mean. Similarly, the efficiencies of the proposed elitist breeding strategies and the constraint handling method are both demonstrated positively in the test results in terms of the solution obtained and the computational efforts required to reach to the solution comparing to the compared algorithms. Moreover, the QPSO-EDP(1) algorithm with the proposed constraint handling outperforms the QPSO-EDP(2) algorithm while the DEB-QPSO algorithm has better solution than both QPSO-EDP(1) and QPSO-EDP(2) algorithms as in case I. It is obvious that the proposed DEB-QPSO algorithm has the best convergence performance among the three compared algorithms as shown in Figure 7.

6.3. Case III: 140-Unit System. To demonstrate the capability of the DEB-QPSO algorithm to the large-scale power systems, the proposed method is evaluated on a Korean power system consisting of 140 generators with ramp rate limits with the hydro and pump storage plants not being considered. Twelve of the generating units have the cost function with valve-point effects, and four generating units have the prohibited operating zones.

Since the 140-unit system is a larger system with more nonlinear elements and so as local minima; it is far more difficult to find the global solution than case II. System parameters of the test system are taken from [64] with the load demand set at 49,342 MW. It can be observed in Table 7 that the proposed DEB-QPSO algorithm outperforms the compared methods. Moreover, even the worst result found by the DEB-QPSO algorithm is less costly than the best result of other compared methods, which reveals that the proposed algorithm is able to solve the large-scale ED problems with valve-point effect and prohibited zones effectively.

In addition, the robustness of the DEB-QPSO algorithm is obviously demonstrated by the fact that all the costs obtained by the DEB-QPSO algorithm are within ±0.01% of the mean value. Comparison of the computational time needed for the ED problem by the DEB-QPSO, QPSO-EDP(1), and QPSO-EDP(2) algorithms supports the efficiencies of the proposed elitist breeding strategies and constraint handling method. In Figure 8, the proposed DEB-QPSO algorithm exhibits better convergence properties in solving the large-scale nonconvex ED problems.

7. Conclusion

This paper proposes a DEB-QPSO approach for solving nonconvex, nonsmooth, and nonlinear ED problems. It combines the basic evolutionary processes of QPSO with two elitist breeding strategies, and an efficient improved heuristic constraint handling technique is proposed to solve the ED problems. The bias elitist breeding combined with the series elitist breeding is devised to improve the search efficiency of the algorithm. To handle the constraints, the improved heuristic technique for repairing position of particles and a novel pbests and gbest update method are proposed. These strategies reduce the computational efforts and improve the search efficiency of the solution algorithm for solving ED problems. Such characteristics are demonstrated consistently in all test cases.

The proposed DEB-QPSO method was tested on the ED problems of three widely used power system instances of 15 units, 40 units, and 140 units, respectively, with nonconvex, nonsmooth, and nonlinear characteristics of the generators such as valve-points prohibited operating zones with ramp rate limits as well as transmission network losses. The results of the case studies clearly illustrate the superior features of the proposed DEB-QPSO method such as high-quality solutions, robustness properties, and computational effectiveness. Comparing with other algorithms, the DEB-QPSO algorithm can locate better solution effectively. It is mainly because the proposed elitist breeding scheme can aggrandize the diversity of the swarm that is essential to the exploration and exploitation for the search of the global optima. The framework of the DEB-QPSO algorithm can be used as an efficient optimizer providing satisfactory solutions for ED problems with various features. Future researches will be followed to perfect the DEB-QPSO algorithm for solving ED problems under dynamic environment.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work is supported by the CityU SRG Research Grant no. 7004541.

References

[1] A. J. Wood and B. F. Wollenberg, *Power Generation, Operation, and Control*, John Wiley & Sons, 2012.

[2] S. Granville, "Optimal reactive dispatch through interior point methods," *IEEE Transactions on Power Systems*, vol. 9, no. 1, pp. 136–146, 1994.

[3] Z.-X. Liang and J. D. Glover, "A zoom feature for a dynamic programming solution to economic dispatch including transmission losses," *IEEE Transactions on Power Systems*, vol. 7, no. 2, pp. 544–550, 1992.

[4] C.-L. Chiang, "Improved genetic algorithm for power economic dispatch of units with valve-point effects and multiple fuels," *IEEE Transactions on Power Systems*, vol. 20, no. 4, pp. 1690–1699, 2005.

[5] D. C. Walters and G. B. Sheble, "Genetic algorithm solution of economic dispatch with valve point loading," *IEEE Transactions on Power Systems*, vol. 8, no. 3, pp. 1325–1332, 1993.

[6] S. Hemamalini and S. P. Simon, "Dynamic economic dispatch using artificial immune system for units with valve-point effect," *International Journal of Electrical Power & Energy Systems*, vol. 33, no. 4, pp. 868–874, 2011.

[7] V. S. Aragón, S. C. Esquivel, and C. A. Coello Coello, "An immune algorithm with power redistribution for solving economic dispatch problems," *Information Sciences*, vol. 295, pp. 609–632, 2015.

[8] B. K. Panigrahi, V. Ravikumar Pandi, and S. Das, "Adaptive particle swarm optimization approach for static and dynamic economic load dispatch," *Energy Conversion and Management*, vol. 49, no. 6, pp. 1407–1415, 2008.

[9] Y. Zhang, D.-w. Gong, N. Geng, and X.-y. Sun, "Hybrid barebones PSO for dynamic economic dispatch with valve-point effects," *Applied Soft Computing*, vol. 18, pp. 248–260, 2014.

[10] J. Cai, X. Ma, L. Li, and P. Haipeng, "Chaotic particle swarm optimization for economic dispatch considering the generator constraints," *Energy Conversion and Management*, vol. 48, no. 2, pp. 645–653, 2007.

[11] Z.-L. Gaing, "Particle swarm optimization to solving the economic dispatch considering the generator constraints," *IEEE Transactions on Power Systems*, vol. 18, no. 3, pp. 1187–1195, 2003.

[12] C.-C. Kuo, "A novel coding scheme for practical economic dispatch by modified particle swarm approach," *IEEE Transactions on Power Systems*, vol. 23, no. 4, pp. 1825–1835, 2008.

[13] A. Safari and H. Shayeghi, "Iteration particle swarm optimization procedure for economic load dispatch with generator constraints," *Expert Systems with Applications*, vol. 38, no. 5, pp. 6043–6048, 2011.

[14] S. Khamsawang and S. Jiriwibhakorn, "DSPSO–TSA for economic dispatch problem with nonsmooth and noncontinuous cost functions," *Energy Conversion and Management*, vol. 51, no. 2, pp. 365–375, 2010.

[15] T. Niknam, H. D. Mojarrad, and H. Z. Meymand, "Non-smooth economic dispatch computation by fuzzy and self adaptive particle swarm optimization," *Applied Soft Computing*, vol. 11, no. 2, pp. 2805–2817, 2011.

[16] S. Özyön and D. Aydin, "Incremental artificial bee Colony with local search to economic dispatch problem with ramp rate limits and prohibited operating zones," *Energy Conversion and Management*, vol. 65, pp. 397–407, 2013.

[17] L. dos Santos Coelho and V. C. Mariani, "Combining of chaotic differential evolution and quadratic programming for economic dispatch optimization with valve-point effect," *IEEE Transactions on Power Systems*, vol. 21, no. 2, pp. 989–996, 2006.

[18] L. dos Santos Coelho, T. C. Bora, and V. C. Mariani, "Differential evolution based on truncated Lévy-type flights and population diversity measure to solve economic load dispatch problems," *International Journal of Electrical Power & Energy Systems*, vol. 57, pp. 178–188, 2014.

[19] M. Basu, "Improved differential evolution for economic dispatch," *International Journal of Electrical Power & Energy Systems*, vol. 63, pp. 855–861, 2014.

[20] U. Güvenç, Y. Sönmez, S. Duman, and N. Yörükeren, "Combined economic and emission dispatch solution using gravitational search algorithm," *Scientia Iranica*, vol. 19, no. 6, pp. 1754–1762, 2012.

[21] W.-M. Lin, F.-S. Cheng, and M.-T. Tsay, "An improved Tabu search for economic dispatch with multiple minima," *IEEE Transactions on Power Systems*, vol. 17, no. 1, pp. 108–112, 2002.

[22] S. Pothiya, I. Ngamroo, and W. Kongprawechnon, "Application of multiple Tabu search algorithm to solve dynamic economic dispatch considering generator constraints," *Energy Conversion and Management*, vol. 49, no. 4, pp. 506–516, 2008.

[23] J. H. Park, Y. S. Kim, I. K. Eom, and K. Y. Lee, "Economic load dispatch for piecewise quadratic cost function using Hopfield neural network," *IEEE Transactions on Power Systems*, vol. 8, no. 3, pp. 1030–1038, 1993.

[24] R. Naresh, J. Dubey, and J. Sharma, "Two-phase neural network based modelling framework of constrained economic load dispatch," *IEE Proceedings - Generation, Transmission and Distribution*, vol. 151, no. 3, pp. 373–378, 2004.

[25] N. Sinha, R. Chakrabarti, and P. K. Chattopadhyay, "Evolutionary programming techniques for economic load dispatch," *IEEE Transactions on Evolutionary Computation*, vol. 7, no. 1, pp. 83–94, 2003.

[26] B. K. Panigrahi and V. Ravikumar Pandi, "Bacterial foraging optimisation: Nelder–Mead hybrid algorithm for economic load dispatch," *IET Generation, Transmission & Distribution*, vol. 2, no. 4, pp. 556–565, 2008.

[27] A. Bhattacharya and P. K. Chattopadhyay, "Biogeography-based optimization for different economic load dispatch problems," *IEEE Transactions on Power Systems*, vol. 25, no. 2, pp. 1064–1077, 2010.

[28] L. D. Le, L. D. Ho, D. N. Vo, and P. Vasant, "Hybrid differential evolution and gravitational search algorithm for nonconvex economic dispatch," in *Proceedings of the 18th Asia Pacific Symposium on Intelligent and Evolutionary Systems - Volume 2*, pp. 89–103, Springer, 2015.

[29] N. Jayakumar, S. Subramanian, S. Ganesan, and E. B. Elanchezhian, "Grey wolf optimization for combined heat and power dispatch with cogeneration systems," *International Journal of Electrical Power & Energy Systems*, vol. 74, pp. 252–264, 2016.

[30] D. N. Vo, P. Schegner, and W. Ongsakul, "Cuckoo search algorithm for non-convex economic dispatch," *IET Generation, Transmission & Distribution*, vol. 7, no. 6, pp. 645–654, 2013.

[31] L. Srivastava and H. Singh, "Hybrid multi-swarm particle swarm optimisation based multi-objective reactive power dispatch," *IET Generation, Transmission & Distribution*, vol. 9, no. 8, pp. 727–739, 2015.

[32] M. R. AlRashidi and M. E. El-Hawary, "A survey of particle swarm optimization applications in electric power systems," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 4, pp. 913–918, 2009.

[33] J. Sun, B. Feng, and W. Xu, "Particle swarm optimization with particles having quantum Behavior," in *Proceedings of the 2004 Congress on Evolutionary Computation (IEEE Cat. No.04TH8753)*, pp. 325–331, Portland, OR, USA, 2004.

[34] J. Sun, C.-H. Lai, and X.-J. Wu, *Particle Swarm Optimisation: Classical and Quantum Perspectives*, CRC Press, 2011.

[35] S. M. Mikki and A. A. Kishk, "Quantum particle swarm optimization for electromagnetics," *IEEE Transactions on Antennas and Propagation*, vol. 54, no. 10, pp. 2764–2775, 2006.

[36] S. Li, R. Wang, W. Hu, and J. Sun, "A new QPSO based BP neural network for face detection," in *Fuzzy Information and Engineering*, pp. 355–363, Springer, 2007.

[37] S. N. Omkar, R. Khandelwal, T. V. S. Ananth, G. Narayana Naik, and S. Gopalakrishnan, "Quantum behaved particle swarm optimization (QPSO) for multi-objective design optimization of composite structures," *Expert Systems with Applications*, vol. 36, no. 8, pp. 11312–11322, 2009.

[38] H. M. Dubey, B. K. Panigrahi, and M. Pandit, "Bio-inspired optimisation for economic load dispatch: a review," *International Journal of Bio-Inspired Computation*, vol. 6, no. 1, pp. 7–21, 2014.

[39] J. Sun, W. Fang, D. Wang, and W. Xu, "Solving the economic dispatch problem with a modified quantum-behaved particle swarm optimization method," *Energy Conversion and Management*, vol. 50, no. 12, pp. 2967–2975, 2009.

[40] M. Moradi-Dalvand, B. Mohammadi-Ivatloo, A. Najafi, and A. Rabiee, "Continuous quick group search optimizer for solving non-convex economic dispatch problems," *Electric Power Systems Research*, vol. 93, pp. 93–105, 2012.

[41] Y. Wang, B. Li, G. M. Jing, P. Wang, and J. Wang, "Research of constraint handling techniques for economic load dispatch of power System," in *IEEE Congress on Evolutionary Computation*, pp. 1–8, Barcelona, Spain, 2010.

[42] Z.-L. Yang, A. Wu, and H.-Q. Min, "An improved quantum-behaved particle swarm optimization algorithm with elitist breeding for unconstrained optimization," *Computational Intelligence and Neuroscience*, vol. 2015, Article ID 326431, 12 pages, 2015.

[43] K. E. Parsopoulos and M. N. Vrahatis, *Particle Swarm Optimization and Intelligence: Advances and Applications*, IGI Global, 2010.

[44] Y.-l. Zheng, L.-h. Ma, L.-y. Zhang, and J.-x. Qian, "Empirical study of particle swarm optimizer with an increasing inertia weight," in *The 2003 Congress on Evolutionary Computation, 2003. CEC '03*, pp. 221–226, Canberra, ACT, Australia, 2003.

[45] F. Van den Bergh, *An Analysis of Particle Swarm Optimizers, [Ph.D. thesis]*, University of Pretoria, Pretoria, South Africa, 2006.

[46] J. Sun, X. Wu, V. Palade, W. Fang, C.-H. Lai, and W. Xu, "Convergence analysis and improvements of quantum-behaved particle swarm optimization," *Information Sciences*, vol. 193, pp. 81–103, 2012.

[47] J. Sun, W. Fang, X. Wu, V. Palade, and W. Xu, "Quantum-behaved particle swarm optimization: analysis of individual particle behavior and parameter selection," *Evolutionary Computation*, vol. 20, no. 3, pp. 349–393, 2012.

[48] W. Fang, J. Sun, Y. Ding, X. Wu, and W. Xu, "A review of quantum-behaved particle swarm optimization," *IETE Technical Review*, vol. 27, no. 4, pp. 336–348, 2010.

[49] J. Sun, W. Xu, and W. Fang, "A diversity-guided quantum-behaved particle swarm optimization algorithm," in *Asia-Pacific Conference on Simulated Evolution and Learning*, pp. 497–504, Springer, Berlin, Heidelberg, 2006.

[50] K. S. Tang, T. M. Chan, R. J. Yin, and K. F. Man, *Multiobjective Optimization Methodology: A Jumping Gene Approach*, CRC Press, 2018.

[51] J. Sun, W. Xu, and J. Liu, "Parameter selection of quantum-behaved particle swarm optimization," in *Advances in Natural Computation*, pp. 543–552, Springer, Berlin, Heidelberg, 2005.

[52] W. Fang, J. Sun, and X. Wen-bo, "Improved quantum-behaved particle swarm optimization algorithm based on differential evolution operator and its application," *Journal of System Simulation*, vol. 20, no. 24, pp. 6740–6744, 2008.

[53] K. Yang and H. Nomura, "Quantum-behaved particle swarm optimization with chaotic search," *IEICE Transactions on Information and Systems*, vol. E91-D, no. 7, pp. 1963–1970, 2008.

[54] W. Fang, J. Sun, and W. Xu, "Analysis of mutation operators on quantum-behaved particle swarm optimization algorithm," *New Mathematics and Natural Computation*, vol. 05, no. 02, pp. 487–496, 2009.

[55] J. Sun, W. Fang, V. Palade, X. Wu, and W. Xu, "Quantum-behaved particle swarm optimization with Gaussian distributed local attractor point," *Applied Mathematics and Computation*, vol. 218, no. 7, pp. 3763–3775, 2011.

[56] Y. Li, R. Xiang, L. Jiao, and R. Liu, "An improved cooperative quantum-behaved particle swarm optimization," *Soft Computing*, vol. 16, no. 6, pp. 1061–1069, 2012.

[57] M. Xu, L. Zhang, B. du, L. Zhang, Y. Fan, and D. Song, "A mutation operator accelerated quantum-behaved particle swarm optimization algorithm for hyperspectral endmember extraction," *Remote Sensing*, vol. 9, no. 3, p. 197, 2017.

[58] Z. Yang, A. Wu, H. Liao, and J. Xu, "An elitist promotion quantum-behaved particle swarm optimization algorithm," in *2016 8th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, pp. 347–350, Hangzhou, China, 2016.

[59] S. Cheng, Y. Shi, and Q. Quande, "Population diversity based study on search information propagation in particle swarm optimization," in *2012 IEEE Congress on Evolutionary Computation*, pp. 1–8, Brisbane, QLD, Australia, 2012.

[60] J. Chen, B. Xin, Z. Peng, L. Dou, and J. Zhang, "Optimal contraction theorem for exploration–exploitation tradeoff in search and optimization," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 39, no. 3, pp. 680–691, 2009.

[61] Z. Yang, A. Wu, and H. Min, "A multi-objective PSO algorithm with transposon and elitist seeding approaches," in *2013 Sixth International Conference on Advanced Computational Intelligence (ICACI)*, pp. 64–69, Hangzhou, China, 2013.

[62] P. Chaudhari and H. Agarwal, "Improving feature selection using elite breeding QPSO on gene data set for cancer classification," in *Intelligent Engineering Informatics*, pp. 209–219, Springer, Singapore, 2018.

[63] H. Agarwal and P. Chaudhari, "Enhancing classification accuracy using elite breeding QPSO on gene dataset," in *2017 International Conference on Advances in Computing, Communication and Control (ICAC3)*, pp. 1–6, Mumbai, India, 2017.

[64] J.-B. Park, Y.-W. Jeong, J.-R. Shin, and K. Y. Lee, "An improved particle swarm optimization for nonconvex economic dispatch problems," *IEEE Transactions on Power Systems*, vol. 25, no. 1, pp. 156–166, 2010.

[65] M. N. Abdullah, A. H. A. Bakar, H. Mokhlis, and J. J. Jamian, "An efficient constraint handling approach for economic load dispatch problem with non-smooth cost function," *Applied Mechanics and Materials*, vol. 785, pp. 490–494, 2015.


Advances in Operations Research
Advances in Decision Sciences
Journal of Applied Mathematics
The Scientific World Journal
Journal of Probability and Statistics
International Journal of Mathematics and Mathematical Sciences
Hindawi
Submit your manuscripts at
www.hindawi.com
Journal of Optimization
International Journal of Engineering Mathematics
International Journal of Analysis
Journal of Complex Analysis
Advances in Numerical Analysis
Mathematical Problems in Engineering
International Journal of Differential Equations
Discrete Dynamics in Nature and Society
International Journal of Stochastic Analysis
Journal of Mathematics
Journal of Function Spaces
Abstract and Applied Analysis
Advances in Mathematical Physics