Electrical Power and Energy Systems 43 (2012) 779–787

Contents lists available at SciVerse ScienceDirect

Electrical Power and Energy Systems

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

Solve environmental economic dispatch of Smart MicroGrid containing distributed generation system – Using chaotic quantum genetic algorithm

Gwo-Ching Liao *

Fortune Institute of Technology, Department of Electrical Engineering, No. 1-10, Nwongchang Rd., Daliao District, Kaohsiung City 83160, Taiwan

ARTICLE INFO

Article history:
Received 30 December 2010
Received in revised form 5 June 2012
Accepted 9 June 2012
Available online 15 July 2012

Keywords:
Smart MicroGrid
Renewable energy
Distributed generation
Photovoltaic
Wind power
Water turbine

ABSTRACT

With the decreasing of the fossil fuel energy resources and the increasing energy load demand Distributed Generation (DG) technologies have received more attention Smart MicroGrid (SMG) systems integrate the power generation advantages from new and renewable energy power generation systems connected to the standard grid. SMG can enhance the comprehensively cascaded energy utilization and also provide an effective complementary network that improves power supply reliability and power quality. SMG has become one of the most up-to-date and important topics in the field of power systems all over the world. According to distributed generation SMG characteristics, such as Photo Voltaic (PV), Wind Power (WP), Water Turbine (WT), Fuel Cell (FC), gas turbine and micro-gas turbine, considering different fuel efficiency, operation and maintenance costs, the greenhouse gas emission levels of distributed generation with various types and capacity a novel SMG model environmental and economic dispatch is presented that considers generation cost and emission costs. This paper uses the quantum genetic algorithm to confirm the accuracy and validity of a mathematic model using actual examples compared with other optimization approaches used to solve the economic dispatch problem. The superiority and usability of the proposed approach is discussed.

1. Introduction

Energy resources are the foundation of all livelihood and human development. We want to use regional rich clean and renewable energy sources fully to provide "the green electrical power" to our users. Energy conservation is the goal Distributed Generation produced the technology to obtain more values and the greater application from the energy generating system. The DG unit has the advantages low investment cost nimble electricity genera in a compatible environment obtaining widespread application in the distribution grid. It therefore energizes the generation distribution system using the "Smart MicroGrid" [1–4] with the main electrical grid incorporation. The main electrical grid mutual support this model displays the best system effectiveness.

Smart MicroGrid technology is the synthesis of new power electronic technology, distributional generation, renewable energy source and stored energy technology. SMG has these principal advantages: (1) The SMG provides effective DG unit integration inherited the merit of all DG units alone. (2) The SMG is an independence module that will not have an adverse effect on the main grid and would not revise the main grid revolution strategy. (3) The SMG may isolate and safeguard the power supply, enhancing the power supply reliability when the previous-level power distribution grid breaks down. The SMG environmental economic dispatch [5], satisfies the load under reasonable and effective output of each DG unit. SMG allows the entire MicroGrid's generating cost, emissions cost and the total cost to achieve the minimum value.

In the past, many methods were deployed to solve the economic dispatch problem. For example: Dynamic Programming (DP) [6,7], with the advantage of discontinuous or non-monotonically increasing cost function and the disadvantage of computational memory requirement and exponentially increased computational time when the number of units increases. Because there are multiple local minimums at this point of time only suboptimal solutions can be found. On the other hand, many optimization methods have been developed in the Artificial Intelligence field. There are also some scholars who applied random optimization methods on the economic dispatch problem, such as Simulated Annealing (SA) [8–10], Genetic Algorithm (GA) [11–17], Evolutionary Programming (EP) [18,19], Evolutionary Strategy (ES) [20], Particle Swarm Optimization (PSO) [21–27], Ant Colony System (ACS) [28–30]. These methods are effective optimization techniques with the capability to find the global optimal solution. When looking for the optimal solution using the SA method, a probabilistic approach is used to accept the candidate solutions in order to avoid local optimal solution trap. However, the related parameters are not easy to set up and the computational time is long. It is not easy

* Tel.: +886 7 7889888; fax: +886 7 7889777.
E-mail address: liaogwo@ms68.hinet.net

0142-0615/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.ijepes.2012.06.040

to implement when applied to the large electrical power systems. The GA method has also been successfully used in solving the economic dispatch problem. However, its disadvantage is its long computational time and the lack of guarantee that a global optimal solution can always be found. The EP method has also been used successfully in solving the economic dispatch problem. However, when encountering larger systems, its long computational time is still its main drawback. The PSO and ACS methods similarly have major limitations in the numerical technique problem dimensions, large computational time and complexity in programming.

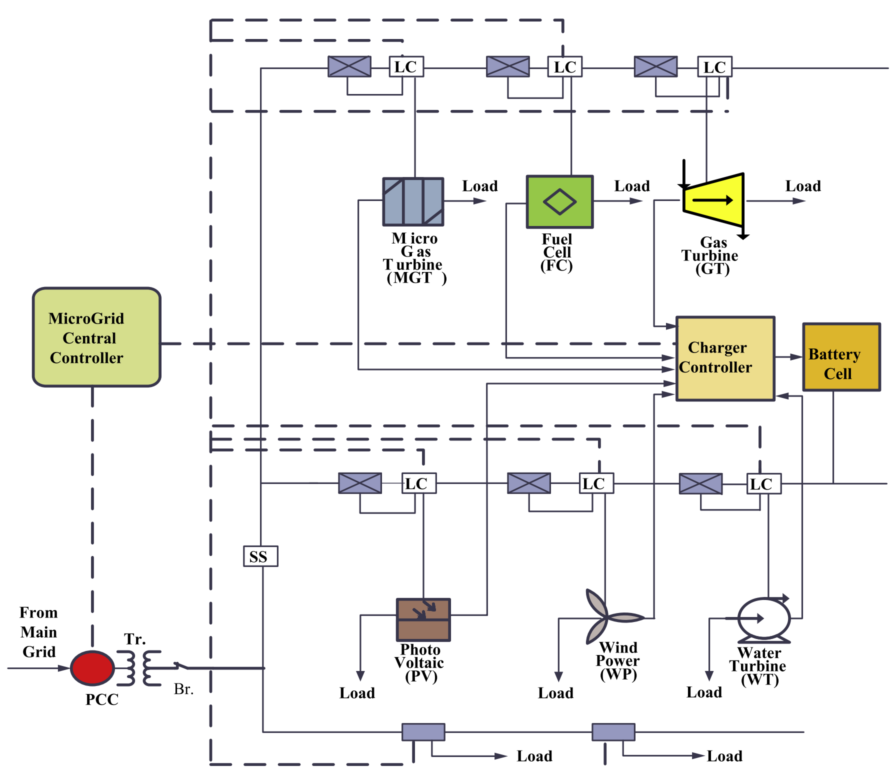
The Chaotic Quantum Genetic Algorithm (CQGA) [31–38] is adopted in this paper to borrow the Quantum computing concept (such as quantum bit and quantum superposition state). It uses quantum bits to encode the chromosomes. This algorithm uses the quantum probability vector encoding mechanism and adopts the genetic algorithm crossover update strategy to effectively improve the global search ability and escape the local optimal solution trap by using chaotic algorithms [39,40], so that it can achieve the real objective of the global optimal solutions.

2. The base structure of Smart MicroGrid [1–5]

The base structure of Smart MicroGrid is shown in Fig. 1. In the figure, include several elements of DG unit. These systems and elements provide power to the load in-together. The entire SMG relative to main grid or Smart Grid (SG) is a whole. To relative to the transformer substation of previous rank of SG through a breaker. The DG unit of SMG can include several form of sources, those are renewable (Photo Voltaic-PV, Wind Power-WP and Water Turbine-WT), non-renewable source (Fuel Cell-FC, Gas Turbine and Micro-Gas Turbine). Another, it can through union generation of heart and electricity, or the style of union generation of heart, cool and electricity to supply heart or cool to the users. It can raise the efficient of multi degree used of sources. In addition, if the main grid is out of work, or the quality of electricity cannot achieve the standard of system. In the condition the SMG can access independent operation under the style of isolated model to guarantee itself of SMG and the normal operation of main grid. Then, enhance the reliability and security of power supply. Therefore, isolated operation is the most important ability of SMG. Look from the main grid, the SMG is just as the load or generator in the electric grid, is a modular of the whole system. In addition look from the user situation, MG is an independent operation system it can satisfy the demands of the electric quality and reliability that is raised from the users.

3. A mathematical model of a Smart MicroGrid environmental protection and economic dispatch system

The Smart MicroGrid environmental protection and economic dispatch [5] system adjusts the existing DG unit output in the microgrid. It satisfies the power balance and the load demand by solving the minimum generating cost values and greenhouse emissions cost.



Note: PCC (Point of Common Coupling), SS (State Switch), LC (Local Controller).

—— : power line.

……… : Communication control line.

Fig. 1. The base structure of Smart MicroGrid.

3.1. Objective function

$$\text{Min } F(P_i) = \sum_{i=1}^{n} C_{g,i}(P_i) + \sum_{i=1}^{n} C_{e,i}(P_i) \tag{1}$$

where $F(\cdot)$ is the total operation cost (US \$), n is the unit number of DG unit, $C_{g,i}$ is generation cost of No. i unit of DG, $C_{e,i}$ is emission cost of No. i unit of DG, P_i is generation power of No. i unit of DG.

3.1.1. Generation cost

The cost of DG unit power generation includes both fuel, operation and maintenance:

$$C_{g,i}(P_i) = C_{f,i}(P_i) + C_{o,i}(P_i) \tag{2}$$

where $C_{g,i}(P_i)$ is generation cost of No. i unit of DG.

3.1.1.1. Fuel cost

$$C_{f,i}(P_i) = K_{f,i} \times P_i \tag{3}$$

where $C_{f,i}(P_i)$ is fuel cost of No. i unit of DG (US \$), $K_{f,i}$ is fuel coefficient of No. i unit of DG (US \$/kW h); it means the fuel consumption fuel expense for generating 1 kW h of DG unit. P_i is the generation power of No. i unit of DG.

3.1.1.2. Operation and maintenance cost.

$$C_{o,i}(P_i) = K_{o,i} \times P_i \tag{4}$$

where $C_{o,i}(P_i)$ is the operating and maintenance cost for No. i unit of DG (US \$), $K_{o,i}$ is the operating and maintenance coefficient for No. i unit of DG (US \$/kW h), is the operating and maintenance cost for generating 1 kW h of DG unit.

3.1.2. Emissions cost

$$C_{e,i}(P_i) = K_{e,i} \times M_e \times P_i \tag{5}$$

where $C_{e,i}(P_i)$ is the emissions cost of No. i unit of DG (US \$), $K_{e,i}$ is the emissions coefficient for No. i unit of DG (kg/kW h); it means Generates 1 kW h electricity every time the emission greenhouse gas weight of the No. i unit of DG. M_e is the greenhouse gas emissions cost (US \$/kg), it means the discharge fee which every time emission 1 kg greenhouse gas must pay.

3.2. Equality limiting condition

At the same time of environmental and economic dispatch, it essentially satisfies condition of the power balance limit.

$$P_D = \sum_{i=1}^{n} P_i \tag{6}$$

where P_D is the total load demand.

3.3. Inequality limiting condition

To guarantee steady DG running each DG satisfies the capacity limiting condition. This means the DG power generation satisfies the upper or lower value limit for the output of each unit.

$$P_i^{min} \leqslant P_i \leqslant P_i^{max} \tag{7}$$

where P_i^{max}, P_i^{min} is respectively the upper or lower limit value of No. i unit of DG.

4. The chaotic quantum genetic algorithms apply to economic dispatch problems [31–38]

4.1. The chaotic quantum genetic algorithm concept

Quantum Evolutionary Algorithm [31] (QEA) developed from the basic quantum information science is an evolutionary algorithm based on the quantum computing concept. It incorporates concepts such as superposition state in quantum computing and adopts the unique coding format to achieve better experimental results on the combinatorial optimization problem. However, when dealing with Multi-modal function optimization using QEA, in particular, the high dimensional multi-modal function optimization problem, it is prone to fall into the local optimal and its computational efficiency is not high.

For the above-mentioned deficiencies of QEA, this research integrated the global optimization ability of genetic algorithms, local searching capability based quantum probability model, the sensitive dependence of chaotic algorithms to initial value, and the traverse of the search space to establish a new improved quantum evolution algorithm, that is "Chaotic Quantum Genetic Algorithm".

4.2. The steps for the chaotic quantum genetic algorithm in economic dispatch

Step 1: Initialize population generated using chaotic algorithms.

4.2.1. Chaotic system [39]

Chaos is a ubiquitous, nonlinear phenomenon in nature. Its behavior is complex, and similar to the random, exists a delicate inner regularity. Chaotic system optimization in creating chaos variables generally uses the Logistic mapping, which

$$x_{k+1} = \mu x_k(1 - x_k) \tag{8}$$

where μ is the chaotic attractor, when $\mu = 4$, the system enters into a chaotic state, resulting in chaotic variables x_k ($k = 1,2,3,\ldots$), with value in the range of (0,1).

4.2.2. Qubit

In CQGA, the smallest information unit is qubit, a qubit state can be 0 or 1, and its state can be expressed as

$$|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle \tag{9}$$

where α, β represent two complex $|\alpha|^2 + |\beta|^2 = 1$ of the probability of corresponding states; $|\alpha|^2$ and $|\beta|^2$ represent the probability of qubit in the state of 0 and 1, respectively.

4.2.3. Quantum chromosome

Frequently used coding methods in Evolutionary Algorithm (EA) are binary, decimal, and symbolic coding. In CQGA, a new coding method based on quantum bit is adopted, i.e. using a pair of complex numbers to define a quantum bit. A system with m quantum bits can be described as

$$\left[\begin{array}{c|c|c|c|} \alpha_1 & \alpha_2 & \cdots\cdots & \alpha_m \\ \beta_1 & \beta_2 & \cdots\cdots & \beta_m \end{array}\right] \tag{10}$$

where $|\alpha h|^2 + |\beta h|^2 = 1$ ($h = 1,2,\ldots,m$). This representation method can be used to express any linear superposition of states. For example: a three quantum bits system with the following probability amplitudes:

$$\left[\begin{array}{c|c|c|} \frac{1}{\sqrt{2}} & \frac{\sqrt{3}}{2} & \frac{1}{2} \\ \frac{1}{\sqrt{2}} & \frac{1}{2} & \frac{\sqrt{3}}{2} \end{array}\right] \tag{11}$$

The system state can be expressed as

$$\frac{\sqrt{3}}{4\sqrt{2}}|000\rangle + \frac{3}{4\sqrt{2}}|001\rangle + \frac{1}{4\sqrt{2}}|010\rangle + \frac{\sqrt{3}}{4\sqrt{2}}|011\rangle + \frac{\sqrt{3}}{4\sqrt{2}}|100\rangle + \frac{3}{4\sqrt{2}}|101\rangle + \frac{1}{4\sqrt{2}}|110\rangle + \frac{\sqrt{3}}{4\sqrt{2}}|111\rangle \tag{12}$$

The above result shows $|000\rangle$, $|001\rangle$, $|010\rangle$, $|011\rangle$, $|100\rangle$, $|101\rangle$, $|110$, $|111$ that the probability of occurrence are $\frac{3}{32}, \frac{9}{32}, \frac{1}{32}, \frac{3}{32}, \frac{3}{32}, \frac{9}{32}, \frac{1}{32}$ and $\frac{9}{32}$ respectively.

4.2.4. Chaotic quantum populations

Using the following m units of the Logistic mapping to generate m units of chaotic variables [39]:

$$x_{o+1,h} = \mu_h \times x_{o,h} \times (1 - x_{o,h}) \tag{13}$$

where $\mu_h = 4$, and h is the sequence number of chaotic variables. If $o = 0$, using a set different initial values to a given m units of chaotic variables and uses Eq. (13) to produce m units of chaotic variables ($h = 1,2,\ldots,m$). Using the first solution qubit of the m-chaotic variable initialization populations, let $o = 1,2,\ldots,N-1$, and based on the previous method to produce another $N-1$ solutions, and these N solutions will consist of the initial populations. Using the o solution of P_o as an example, the initialization results are:

$$P_o = \left[\begin{array}{c|c|c|} \alpha_{o,1} & \alpha_{o,2} & \ldots\ \alpha_{o,m} \\ \beta_{o,1} & \beta_{o,2} & \ldots\ \beta_{o,m} \end{array}\right] \tag{14}$$

$$\alpha_{o,h} = \cos(2 \cdot x_{o,h} \cdot \pi); \beta_{o,h} = \sin(2 \cdot x_{o,h} \cdot \pi)$$

Step 2: The encoding and individual measurements to the power output of generator in populations.

CQGA and EA are similar probability algorithm. Their algorithm flow is shown in Fig. 2, $G(t) = \{Q_1^t, Q_2^t, \ldots, Q_g^t \ldots, Q_l^t\}$, ($g = 1,2,\ldots,l$). Where g is the size of population. In $Q_l(t) = \{q_1^t, q_2^t, \ldots, q_o^t \ldots, q_n^t\}$, n is the number of generating units, t for the evolution of algebra, q_o^t for the binary code of the power generation of the oth generator, in which the chromosome is defined as

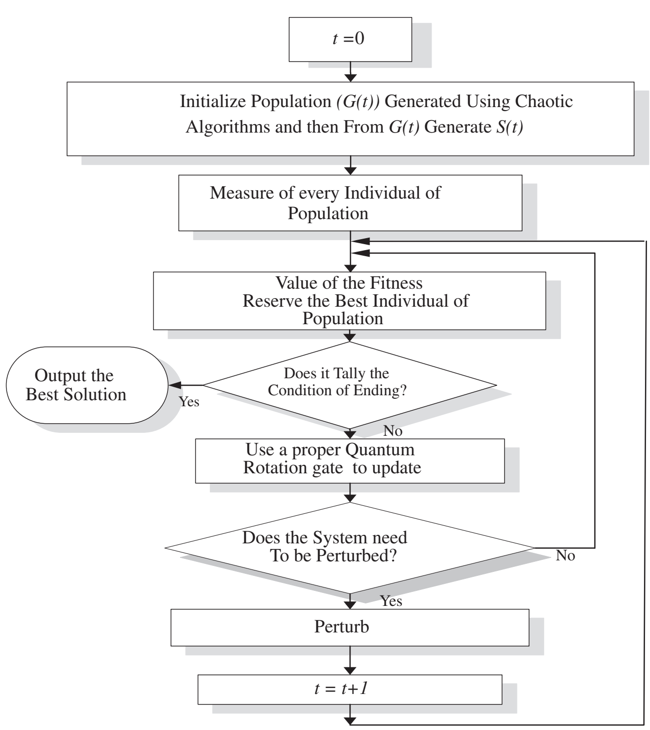


Fig. 2. The flowchart of CQGA Algorithm.

$$q_o^t = \left[\begin{array}{c|c|c|} \alpha_1^t & \alpha_2^t & \ldots\ \alpha_m^t \\ \beta_1^t & \beta_2^t & \ldots\ \beta_m^t \end{array}\right] \tag{15}$$

($o = 1,2,\ldots,n$) (m for the length of quantum chromosome)

In the "initial population $G(t)$", if α_h^t, β_h^t ($h = 1,2,\ldots,m$) and all q_o^t in $Q_l(t)$ have been initialized, it means that all of the possible linear superposition states could emerge with the same probability.

In the step of "from $G(t)$ produces $S(t)$", by observing the state to generate a general solution set of $S(t)$, in which in the t generation, $S(t) = \{P_1^t, P_2^t, \ldots, P_g^t \ldots, P_l^t\}$ and $P_l(t) = \{x_1^t, x_2^t, \ldots, x_o^t \ldots, x_n^t\}$, for each x_o^t ($o = 1,2,\ldots,n$) is the length of m of the string ($z_1, z_2, \ldots, z_h, \ldots, z_m$), it is obtained by the extent of qubit $|\alpha_h^t|^2$ or $|\beta_h^t|^2$ ($h = 1,2,\ldots,m$), corresponding to the binary case is a random process to generate a [0,1] number. If it is larger than $|\alpha_h^t|^2$, 1 will be taken, otherwise the 0.

Step 3: Perform individual measurement to each object in $S(t)$.

Use a fitness evaluation function to evaluate each individual object in $S(t)$ and keep the best object in the generation. If a satisfactory solution is obtained, stop the algorithm; otherwise, continue to the 4th step.

Step 4: Use a proper quantum rotation gate $U(t)$ to update $S(t)$.

The traditional genetic algorithm uses mating, and mutation, etc. operations to maintain the diversity of the population. Quantum genetic algorithm [33] applies logic gate to the probability amplitude of quantum state to maintain the diversity of the population. Therefore, the update method using a quantum gate is the key to the quantum genetic algorithm. In the traditional genetic algorithm, binary system, adaptation values, and probability amplitude comparison method are used for update using a quantum gate. This update method using quantum gate is suitable to find solutions for combinatorial optimization problems with known optimal solution in principle. However, for the actual optimization problems, in particular, those multi-variable continuous function optimization problems, their optimal solutions are not known beforehand in principle. Therefore, here, a quantum rotation gate of quantum logic gate is adopted for the new quantum genetic algorithm.

$$U = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \tag{16}$$

where θ is the quantum gate rotation angle. Its value is taken as

$$\theta = \omega \cdot f(\alpha_i, \beta_i) \tag{17}$$

$$\omega = \pi \cdot \exp\left(-\frac{t}{iter_{max}}\right) \tag{18}$$

We define "ω" as a variable related to the generation of the evolution so that it will adjust the size of the grid self-adaptively. Where t is the evolution generation, π is an angle, $iter_{max}$ is a constant depending on the complexity of the optimization problem. The purpose of the function $f(\alpha_i, \beta_i)$ is to make the algorithm search along the optimal direction. The search strategies listed in Table 1 are adopted here. Its principle is to make the current search solution approach the optimal solution gradually and, thereby, determine

Table 1
Look up table for $f(\alpha_i, \beta_i)$ function.

d_1 and d_2		$f(\alpha_i, \beta_i)$	
$d_1 > 0$	$d_2 > 0$	$\lvert\xi_1\rvert > \lvert\xi_2\rvert$	$\lvert\xi_1\rvert < \lvert\xi_2\rvert$
True	True	+1	−1
True	False	+1	+1
False	True	−1	−1
False	False	−1	+1

Table 2
The total load reference data of each time period for all the sources at the Dth day in summer.

Time (h)	Loads (MW)	Time (h)	Loads (MW)
1	223	13	663
2	231	14	612
3	203	15	667
4	196	16	678
5	215	17	600
6	223	18	576
7	361	19	567
8	572	20	515
9	686	21	437
10	711	22	417
11	722	23	396
12	752	24	333

the direction of the quantum rotation gate. In Table 1, α_1 and β_1 are the probability amplitude for the optimal solution found, $d_1 = \alpha_1 \times \beta_1$, $\xi_1 = \tan^{-1}(\beta_1/\alpha_1)$, α_2 and β_2 is the probability amplitude for the current solution, $d_2 = \alpha_2 \times \beta_2$, $\xi_2 = \tan^{-1}(\beta_2/\alpha_2)$. When d_1 and d_2 are all larger than 0 at the same time, It means that the current solution and the optimal solution found are both in the first or the third quadrant. When $|\xi_1| > |\xi_2|$, the current solution should be rotated counterclockwise, which has a value of "+1"; otherwise, its value should be "−1". By the same token, the other three scenarios can be deduced.

In this way, the procedure for applying the quantum rotation gate to all the probability amplitudes for individual object in the population, i.e. using quantum rotation gate $U(t)$ to update $S(t)$, we have

$$S(t+1) = U(t) \times S(t) \tag{19}$$

$U(t)$ is the tth generation quantum rotation gate, $S(t)$ is the tth generation probability amplitude of a certain object, and $S(t+1)$ is the $(t+1)$th generation probability amplitude of the corresponding object.

Step 5: Perturbation.

In order to solve the problem of CQGA, problem being prone to be trapped in local extreme value better, we perturb the population. It is found that by using the CQGA analysis that when the best individual of the current generation is a local extreme value, it is very hard for the algorithm to extricate itself. Therefore, when the best individual does not change in successive generations, the algorithm is trapped in the local extreme. At this point of time, a perturbation should be applied to the population to extricate itself out of the local optimal and start a new search. The approach is to maintain the optimal solution in this generation, making the remaining individuals similar to the genetic algorithm crossover operation. Two individuals then form a pair. The two-point crossover operation forms a linear combination into new individual using two different parents. A new population is built until the number is same as the original population. The operation then ends.

The complete CQGA computation flowchart is shown in Fig. 2:

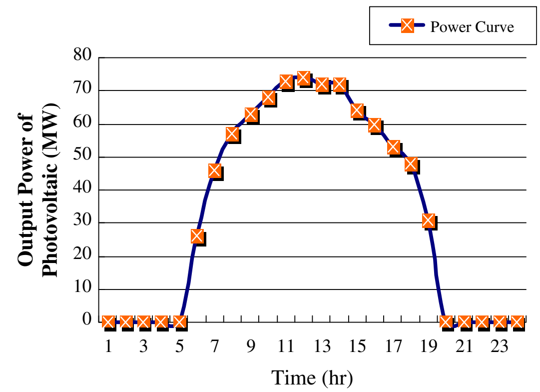


Fig. 4. The output power of each time period for the photovoltaic in a particular day (the Dth day).

5. Simulation and results

5.1. Studies Case 1

There is one wind farm and five kinds of different power sources (can see from Fig. 1). Among them, the total load reference data for each time period for all six kinds of sources at the Dth day in summer are listed in Table 2. The wind farm contains number 20 wind turbine generators of the same model operating in parallel. The rated total effective power output is 100 MW. The wind power generation curve at each time period within the study period at the Dth day in summer is shown in Fig. 3. It got from using the forecasted wind power beforehand through Neural Network and then converted into electrical power. The corresponding minimum output power is 15 MW and the maximum output power is 100 MW. Fig. 4 shows the output power of each time period for the photovoltaic at Dth day in summer. The output power was cal-

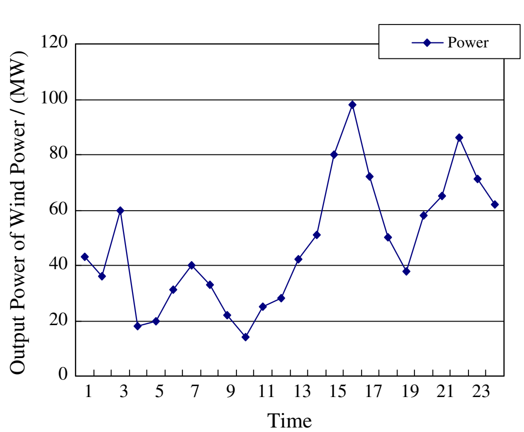


Fig. 3. The output power of each time period for the wind farm in a particular day (the Dth day).

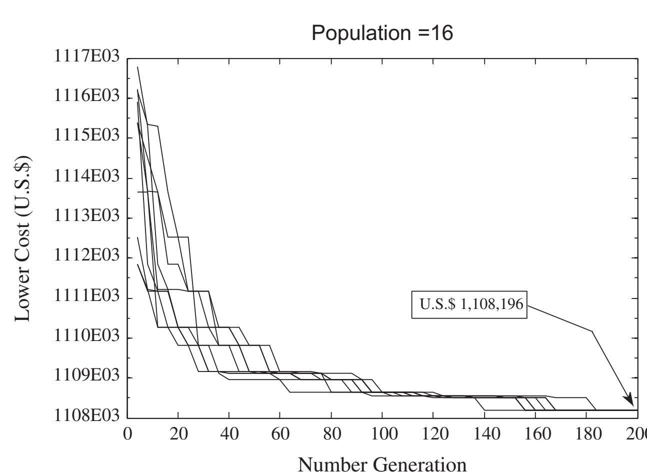


Fig. 5. The convergence scenario for the total cost of power generation using the CQGA method (the number of population is 16, Generation = 200, computation 10 times in total).

Table 3
The effects of the magnitude of rotation angle and the population size on cost and CPU time.

Population size	$\theta \times \pi$ (radians)	Time (s)	Cost (US $)		
			Best	Worst	Mean
4	0.01	10.37	1,103,277	1,114,351	1,108,814
	0.02	10.38	1,103,256	1,114,335	1,108,795
	0.03	10.36	1,103,323	1,114,456	1,108,895
	0.04	10.37	1,103,413	1,114,532	1,108,973
	0.05	10.38	1,103,478	1,114,612	1,109,045
8	0.01	21.22	1,103,179	1,114,118	1,108,649
	0.02	21.25	1,103,123	1,114,123	1,108,677
	0.03	21.23	1,103,231	1,114,231	1,108,731
	0.04	21.24	1,103,289	1,114,335	1,108,812
	0.05	21.25	1,103,356	1,114,423	1,108,889
12	0.01	32.56	1,102,612	1,113,955	1,108,284
	0.02	32.55	1,102,654	1,113,934	1,108,294
	0.03	32.57	1,102,723	1,113,978	1,108,351
	0.04	32.56	1,102,798	1,114,056	1,108,427
	0.05	32.58	1,102,865	1,114,099	1,108,482
16	0.01	42.04	1,1025,20	1,113,883	1,108,202
	0.02	42.03	1,102,517	1,113,876	1,108,196
	0.03	42.07	1,102,675	1,113,943	1,108,309
	0.04	42.09	1,102,734	1,113,978	1,108,356
	0.05	42.12	1,102,812	1,113,994	1,108,403
20	0.01	53.33	1,102,521	1,113,873	1,108,197
	0.02	53.37	1,102,501	1,113,852	1,108,177
	0.03	53.35	1,102,632	1,113,912	1,108,272
	0.04	53.57	1,1026,89	1,113,921	1,108,277
	0.05	53.51	1,102,723	1,113,972	1,108,348
24	0.01	63.12	1,102,489	1,113,708	1,108,099
	0.02	63.18	1,102,483	1,113,712	1,108,098
	0.03	63.13	1,102,521	1,113,767	1,108,144
	0.04	63.16	1,102,631	1,113,823	1,108,227
	0.05	63.11	1,102,723	1,113,912	1,108,318
28	0.01	74.67	1,102,317	1,113,614	1,107,966
	0.02	74.62	1,102,324	1,113,624	1,107,974
	0.03	74.61	1,102431	1,113,676	1,108,053
	0.04	74.68	1,102,507	1,113,746	1,108,126
	0.05	74.69	1,102608	1,113,837	1,108,223

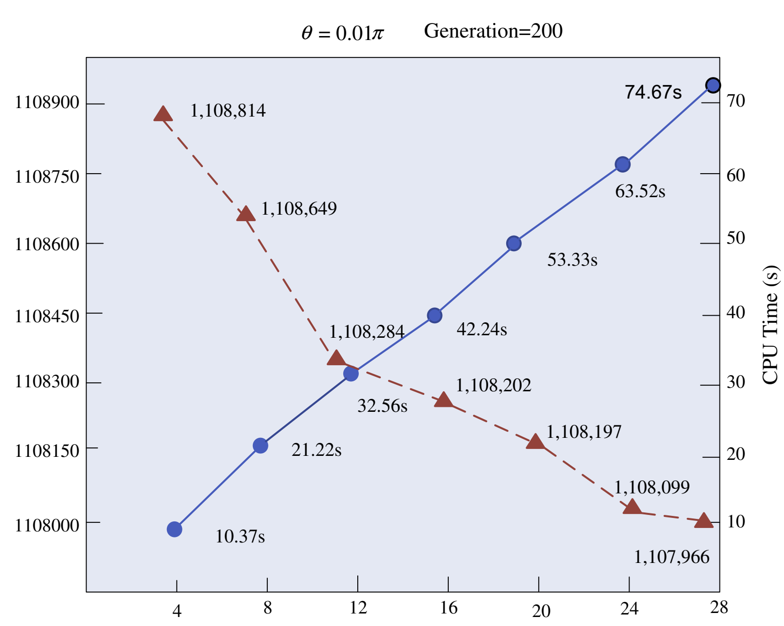


Fig. 6. The effects of θ value and population size on cost and CPU time (while $\theta = 0.01\pi$ and Generation = 200).

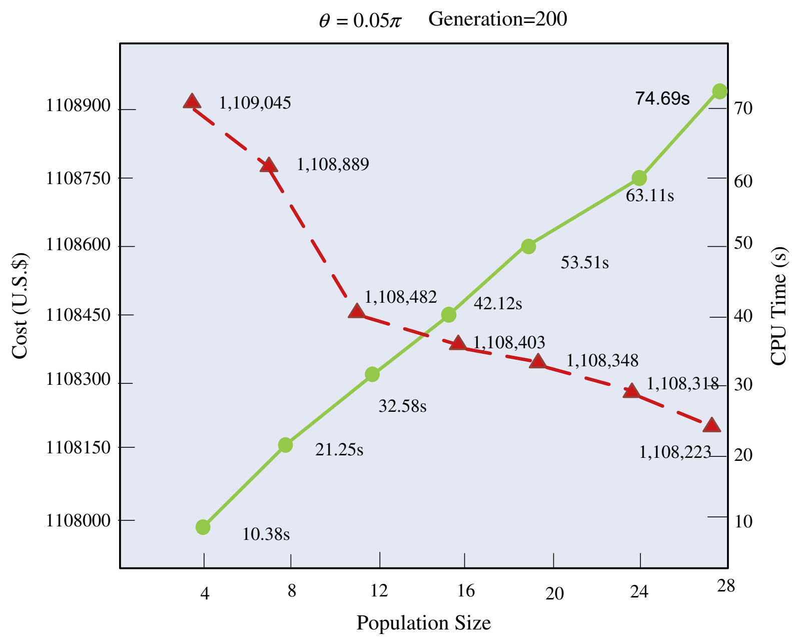


Fig. 7. The effects of θ value and population size on cost and CPU time (while $\theta = 0.05\pi$ and Generation = 200).

Table 4
The effects of generation number on cost and CPU time (while population size is 16 and $\theta = 0.02\pi$).

Generation number	CPU time (s)	Cost (US $)		
		Best	Worst	Mean
100	23.39	1,102,834	1,114,123	1,108,479
200	42.03	1,102,517	1,113,876	1,108,196
300	63.78	1,102,211	1,113,714	1,107,963
400	81.89	1,101,943	1,113,212	1,107,578
500	106.27	1,101,745	1,113,011	1,107,378
600	133.77	1,101,612	1,112,657	1,107,135

culated using the same method mentioned in Fig. 3. Fig. 5 shows the convergence scenario for the power generation operating cost at *D*th day in summer. When using the CQGA method to solve the Dynamic Economic Dispatch problem, the constraints are: six kinds of DG source condition, their evolution generation is set to 200, the population size is 16, the number of computation is 10 (therefore, there are number 10 cost curves in the figure). The convergence scenario for the operating cost converges to the minimum cost, which is about US $ 1,108,196 (see Fig. 5).

5.2. Studies Case 2

In Table 3 is the magnitude of the rotation angle in radians and the Population Size are studied to settle the optimal value. From Table 3 the effect of the Population Size investigated by varying the size from 4 to 28, with a step size of 4. The value of θ is from 0.01π to $\theta = 0.05\pi$ with step size of 0.01π.

As observed the comparison from Figs. 6 and 7 the CQGA algorithm convergence cost is relative to the angle of θ. In Fig. 6 the running condition is set $\theta = 0.01\pi$ and Generation = 200. In Fig. 7 the running condition is set $\theta = 0.05\pi$ and Generation = 200. Since large angles may cause premature convergence, small angles generally produce better solutions. According to our test while $\theta = 0.02\pi$ the cost can always converge to the optimal value weather the Population Size is set at any value. From Table 3, in Figs. 6 and 7 a large Population Size can slightly improve the mean cost value, but it will increase the CPU time.

Observed from Table 4 when the Generation number is large the convergence cost will be reduced. As an example, while Generation = 200, the mean cost = US $ 1,108,196, the CPU time = 42.03(s), another while Generation = 600, the mean cost = US $ 1,107,135, the CPU time = 133.77(s). The cost reduces only US $ 1061, but the CPU time increase 91.74(s). Obviously, selected Generation number = 200 is better for the solution.

5.3. Studies Case 3

Table 5 shows the total power generation cost and CPU run time on the *D*th day in summer under above operating conditions while using different methods (i.e. CQGA, QGA, EP, GA and DP). From the table, we can obtain a clear result through the comparison of the average convergence of power generation cost – i.e. CQGA can save

Table 5
The different total Generation Cost and computation time of CPU on *D*th day in summer while using the five different kinds of algorithms (those are DP, GA, EP, QGA and CQGA).

Different algorithm	Convergence cost of generation (US $)			CPU time of convergence (s)		
	Min.	Max.	Ave.	Min.	Max.	Ave.
DP	1,158,135	1,172,573	1,165,354	123.73	145.56	134.65
GA	1,143,576	1,155,998	1,149,787	96.92	116.67	106.80
EP	1,131,364	1,142,773	1,137,068	83.86	95.73	89.79
QGA	1,116,993	1,128,371	1,122,682	62.64	74.19	68.42
CQGA	1,102,517	1,113,876	1,108,196	35.17	48.88	42.03

Note: DP-Dynamic Programming, GA-Genetic Algorithm, EP-Evolutionary Programming, QGA-Quantum Genetic Algorithm.

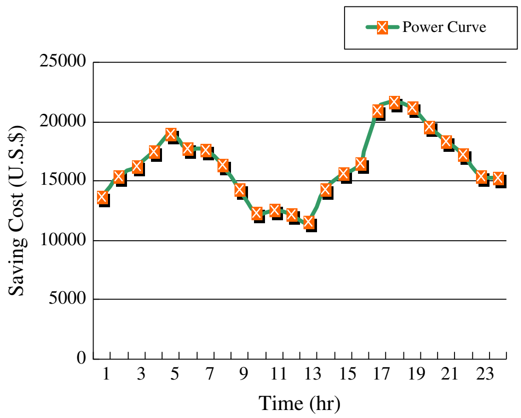


Fig. 8. The saving cost that solves the Dynamic Economic Dispatch problem in a summer working day while using the DG sources and CQGA method compare with the condition that power buy directly from main grid.

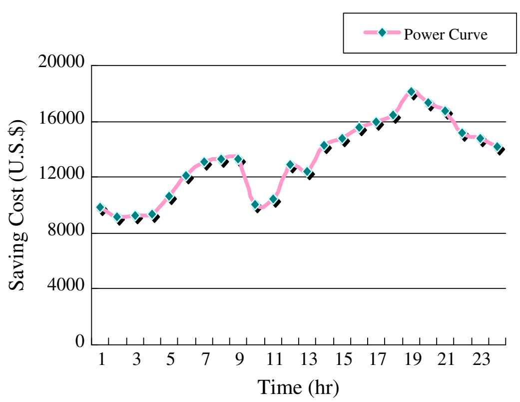


Fig. 9. The saving cost that solves the Dynamic Economic Dispatch problem in a summer weekend day while using the DG sources and CQGA method compare with the condition that power buy directly from main grid.

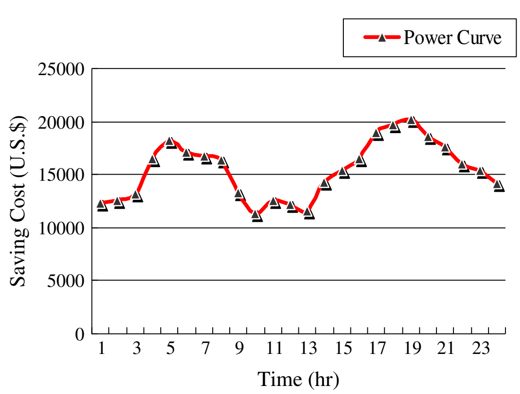


Fig. 10. The saving cost that solves the Dynamic Economic Dispatch problem in a winter working day while use DG sources and CQGA method compare with the condition that power is bought directly from the main grid.

more than QGA [32–35], EP [18], GA [11–14], DP [6] for US $ 14,484, US $ 28,872, US $ 41,591 and US $ 57,058 respectively. Another, the CPU time of CQGA is lower than QGA, EP, GA, DP for 26.39(s), 47.76(s), 64.77(s) and 92.62(s) respectively.

Figs. 8–11 show the saving cost that solves the Dynamic Economic Dispatch in summer or winter when use DG sources and the algorithm mentioned in this paper compare with the condition that power buying directly from the main grid. Fig. 8 is 1 day in

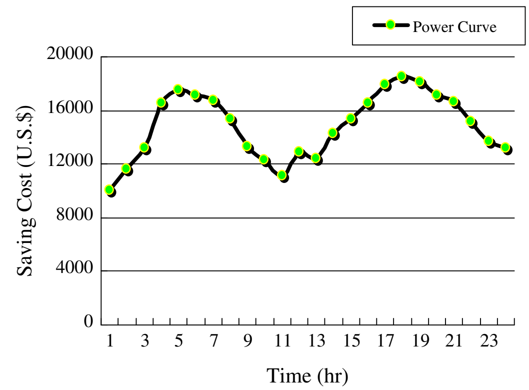


Fig. 11. The saving cost that solves the Dynamic Economic Dispatch problem in a summer weekend day while use DG sources and CQGA method compare with the condition that power is bought directly from the main grid.

summer working day. From the figure we have that the dispatch cost saving is from US $ 11,576 (minimum) to US $ 21,675 (maximum). In Fig. 9 is 1 day in summer weekend day. From the figure we have that the dispatch cost saving is from US $ 9,134 to US $ 18,145. In Fig. 10 is 1 day in winter working day. From the figure we have that it can save the dispatch cost from US $ 11,312 to US $ 20,234. Fig. 11 is 1 day in winter weekend day. From the figure we have that it can save the dispatch cost from US $ 10,316 to US $ 19,475. The comprehensive survey can save cost from US $ 219,216 to US $ 485,616 in 1 day. The total savings in 1 month is from US $ 6,576,480 to US $ 14,568,480. It is really a big number. Therefore using the CQGA method to solve Environmental Economic Dispatch of Smart MicroGrid Containing Distributed Generation problem is really a outstanding and feasible method.

6. Conclusions

The chaotic quantum genetic algorithm method was used in this paper. The quantum bit coding was adopted for the problem coding. Using this algorithm presents several advantages as: (1) The quantum chromosome is generated using quantum encoding. Because the quantum probability amplitude means that a quantum chromosome carries information about multiple states, a chromosome will be in a quantum superposition state of many determined states before an observation is made. Therefore, generating a new object through the quantum probability amplitude, the deciding variable, in a sense, is no longer fixed information; rather, it has become a kind of superposition for different information scenarios. Therefore it can produce a richer population than the simplistic genetic method. It is quite easy to induce mutations with current best individual object information so that the population will evolve toward a good schema with high probability to speed up the convergence. (2) Using the chaotic system, the population possesses inherent diversity, i.e. it is not prone to fall into the local optimal trap during the optimization process.

The algorithm proposed here to solve the environmental economic dispatch problem for Smart MicroGrid power systems is economically significant and quite practical, which is an excellent algorithm.

References

[1] Hernandez-Aramburo CA, Green TC, Mugniot N. Fuel consumption minimization of a microgrid. IEEE Trans Ind Appl 2005;41(3):673–81.
[2] Dimeas L, Hatziargyriou ND. Operation of a multiagent system for microgrid control. IEEE Trans Power Syst 2005;20(3):1447–55.
[3] PecasLopes JA, Moreira CI, Madureira AG. Defining control strategies for microgrid islanded operation. IEEE Trans Power Syst 2006;21(2):916–24.

[4] Chen CL, Lee TY, Jan RM. Optimal wind–thermal coordination dispatch in isolated power system with large integration of wind capacity. Energy Convers Manage 2006;47:3456–72.
[5] Ahn SJ, Moon SI. Economic scheduling of distributed generators in a microgrid considering various constraints. IEEE PES general meeting; 2009. p. 1–6.
[6] Ross DW, Kim S. Dynamic economic dispatch of generation. IEEE Trans Power Apparat Syst 1980;99(6):2060–88.
[7] Shoults RR, Venkatesh SV, Helmick SD, Lolla MJ. A dynamic programming based method for developing dispatch curves when incremental heat rate curves are non-monotonically increasing. IEEE Trans Power Syst 1986;PWRS-1(1):10–6.
[8] Mantawy H, Abdel-Magid YL, Selim SZ. A simulated annealing algorithm for unit commitment. IEEE Trans Power Syst 1998;13(1):197–204.
[9] Wong KP, Fung CC. Simulated annealing based economic dispatch algorithm. IEE Proc – C 1993;140(6):509–15.
[10] Zhuang F, Glaiana FD. Unit commitment by simulated annealing. IEEE Trans Power Syst 1990;5(1):311–8.
[11] Liao GC, Tsao TP. Application of a fuzzy neural network combined with a chaos genetic algorithm and simulated annealing to short-term load forecasting. IEEE Trans Evol Comput 2006;10(3):330–40.
[12] Chen PH, Chang HC. Large scale economic dispatch by genetic algorithm. IEEE Trans Power Syst 1995;10(4):1919–26.
[13] Basu M. Dynamic economic emission dispatch using non-dominated sorting genetic algorithm II. Int J Electr Power Energy Syst 2008;30(2):140–9.
[14] He D, Wang F, Mao Z. A hybrid genetic algorithm approach based on differential evolution for dispatch with valve-point effect. Int J Electr Power Energy Syst 2008;30(1):31–8.
[15] Kazarlis SA, Bakirtzis AG, Petridis V. A genetic algorithm solution to the unit commitment problem. IEEE Trans Power Syst 1996;11(1):83–92.
[16] Liao GC. Integrated isolation niche and immune genetic algorithm for solving bid-based dynamic economic dispatch. Int J Electr Power Energy Syst 2012;42(1):264–75.
[17] Lee JC, Lin WM, Liao GC, Tsao TP. Quantum genetic algorithm for dynamic economic dispatch with valve-point effects and including wind power system. Int J Electr Power Energy Syst 2011;33(2):189–97.
[18] Wong KP, Yuryevich J. Evolutionary programming based algorithm for environmentally constrained economic dispatch. IEEE Trans Power Syst 1998;13(2):301–6.
[19] Yuryevich J, Wong KP. Evolutionary programming based optimal power flow algorithm. IEEE Trans Power Syst 1999;14(4):1245–50.
[20] Gaul AJ, Handschin E, Lehmkoster C. Establishing a rule base for a hybrid ES/XPS approach to load management. IEEE Trans Power Syst 1998;13(1):86–93.
[21] Niknam T, Mojarrad HD, Nayeripour M. A new fuzzy adaptive particle swarm optimization for non-smooth economic dispatch. Energy 2010;35(4):1764–78.
[22] Vahidinasab V, Jadid S. Joint economic and emission dispatch in energy markets: a multiobjective mathematical programming approach. Energy 2010;35(3):1497–504.
[23] Gong DW, Zhang Y, Qi CL. Environmental/economic power dispatch using a hybrid multi-objective optimization algorithm. Int J Electr Power Energy Syst 2010;32(5):337–44.
[24] Abido MA. Multiobjective particle swarm optimization for environmental/economic dispatch problem. Electr Power Syst Res 2009;79(7):1105–13.
[25] Meng K, Wang HG, Dong ZY, Wong KP. Quantum-inspired particle swarm optimization for valve-point economic load dispatch. IEEE Trans Power Syst 2010;25(1):1215–22.
[26] Lu, Sun C, Lu Z. An improved quantum-behaved particle swarm optimization method for short-term combined economic emission hydrothermal scheduling. Energy Convers Manage 2010;51(3):561–71.
[27] Saber AY. Economic dispatch using particle swarm optimization with bacterial foraging effect. Int J Electr Power Energy Syst 2012;34(1):38–46.
[28] Pothiya S, Ngamroo I, Kongprawechnon W. Ant colony optimization for economic dispatch problem with non-smooth cost functions. Int J Electr Power Energy Syst 2010;32(5):478–87.
[29] Niu D, Wang Y, Wu DD. Power load forecasting using support vector machine and ant colony. Expert Syst Appl 2010;37(3):2531–9.
[30] Cai J, Li Q, Li L, Peng H, Yang Y. A fuzzy adaptive chaotic ant swarm optimization for economic dispatch. Int J Electr Power Energy Syst 2012;34(1):154–60.
[31] Sailesh Babu GS, Das DB, Patvardhan C. Real parameter quantum evolutionary algorithm for economic load dispatch. IET Proc – Gener Transm Distrib 2008;2(1):22–31.
[32] Tylavsky DJ, Heydt GT. Quantum computing power system simulation. IEEE power engineering society general meeting 2003, vol. 2; 2003. p. 13–7.
[33] Coelho LDS, Mariani VC. Particle swarm approach based on quantum mechanics and harmonic oscillator potential well for economic load dispatch with valve-point effects. Energy Convers Manage 2008;49(11):3080–5.
[34] Zhisheng Z. Quantum-behaved particle swarm optimization algorithm for economic load dispatch of power system. Expert Syst Appl 2010;37(2):1800–3.
[35] Sun J, Fang W, Wang D, Xu W. Solving the economic dispatch problem with a modified quantum-behaved particle swarm optimization method. Energy Convers Manage 2009;50(12):2967–75.
[36] Zhao S, Xu G, Tao T, Liang L. Real-coded chaotic quantum-inspired genetic algorithm for training of fuzzy neural networks. Comput Math Appl 2009;57(11–12):2009–15.
[37] Coelho LDS, Mariani VC. Reactive power and voltage control based on general quantum genetic algorithms. Expert Syst Appl 2009;36(3):6118–26 [Part 2].
[38] Li P, Li S. Quantum-inspired evolutionary algorithm for continuous space optimization based on Bloch coordinates of qubits. Neurocomputing 2008;72(1–3):581–91.
[39] Coeho LS, Mariani VC. Chaotic artificial immune approach applied to economic dispatch of electric energy using thermal units. Chaos Solitons Fract 2009;40(5):2376–83.
[40] Coeho LS, Mariani VC. Combining of chaotic differential evolution and quadratic programming for economic dispatch optimization with valve-point effect. IEEE Trans Power Syst 2006;21(2):989–96.