

Research Article

Improved Quantum Artificial Fish Algorithm Application to Distributed Network Considering Distributed Generation

Tingsong Du,^{1,2} Yang Hu,¹ and Xianting Ke¹

¹Department of Mathematics, Science College, China Three Gorges University, Yichang 443002, China

²Hubei Province Key Laboratory of System Science in Metallurgical Process, Wuhan University of Science and Technology, Wuhan 430081, China

Correspondence should be addressed to Tingsong Du; tingsongdu@ctgu.edu.cn

Received 28 May 2015; Revised 2 August 2015; Accepted 10 August 2015

Academic Editor: José Alfredo Hernandez

Copyright © 2015 Tingsong Du et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

An improved quantum artificial fish swarm algorithm (IQAFSA) for solving distributed network programming considering distributed generation is proposed in this work. The IQAFSA based on quantum computing which has exponential acceleration for heuristic algorithm uses quantum bits to code artificial fish and quantum revolving gate, preying behavior, and following behavior and variation of quantum artificial fish to update the artificial fish for searching for optimal value. Then, we apply the proposed new algorithm, the quantum artificial fish swarm algorithm (QAFSA), the basic artificial fish swarm algorithm (BAFSA), and the global edition artificial fish swarm algorithm (GAFSA) to the simulation experiments for some typical test functions, respectively. The simulation results demonstrate that the proposed algorithm can escape from the local extremum effectively and has higher convergence speed and better accuracy. Finally, applying IQAFSA to distributed network problems and the simulation results for 33-bus radial distribution network system show that IQAFSA can get the minimum power loss after comparing with BAFSA, GAFSA, and QAFSA.

1. Introduction

Distributed generation (DG) is small-scale and radial generating facilities, which are placed in the vicinity of the load, delivering electricity to consumers independently. Compared with the traditional and centralized power, DG has many advantages [1, 2] such as being near to the center of the users and having high energy efficiency and lower cost of construct. Studies show that DG grid connection has significant impact on the distributed network, including power flow, voltage profile, system losses, and reliability, the extend of which has something to do with the location of DG intimately [3–5]. On the one hand, the suitable installation position and capacity can improve the voltage quality effectively and reduce the active loss; on the other hand, unsuitable configuration turns out to be just the opposite wish and threat to the safe and stable operation of the power network. There are many approaches for deciding the optimum sizing and sitting of DG units in distribution systems. Some of them rely on conventional optimization methods and others use artificial intelligence-based optimization methods [6].

The new methods of distribution network programming, Genetic Algorithm (GA) [7, 8], Ant Colony Algorithm (ACA) [9], Particle Swarm Algorithm (PSA) [10], and Chaos Algorithm (CA), have advantages and disadvantages, respectively. The robustness of the GA is strong in addition to its ability to be trapped in a local minimum. Initial particles generated by CA have property of ergodicity but slower convergence speed. Therefore, how to effectively combine the merits of different intelligent algorithms to improve the performance among search algorithm is a worth studying direction [11–13].

Basic artificial fish swarm algorithm (BAFSA) is a kind of intelligence optimization algorithm based on animal behavior by Li et al. in 2002 [14]. According to the characteristics of the fish swarm and its animal autonomous model, it simulates behavior of fish to achieve the purpose of the group global optimization by each individual in the local optimization. The main fish behaviors are the following: foraging, huddling, following, and being random. In almost all the cases, the BAFSA is easy to avoid falling into local optimum and obtain the global optimum. Although the algorithm with the merits of strong robustness and good convergence performance

is not sensitive to initial value and parameter selection, it has the weakness with search efficiency such as the poor ability to keep balance of exploration and development, late blind search, slow arithmetic speed, and low accuracy of optimization results.

Quantum computing that is different from the traditional calculation model of classical physics has incomparable advantages such as quantum super parallelism and exponential storage capacity [15]. The combination of quantum computing and intelligent optimization algorithm injected new life into the intelligent optimization computing source, by using quantum computing in a new mode of representing and processing information, and so forth [16, 17]. Intelligent optimization algorithm can be designed from another angle to enrich the theory of intelligent computing and improve the traditional method of intelligent search performance as a whole.

In the present paper, the improved quantum artificial fish swarm algorithm (IQAFSA) is proposed in consideration of the slow speed and low accuracy of BAFSA and randomness and blindness of quantum computing [18, 19]. The proposed algorithm improves the coding way of the quits artificial fish and uses quantum revolving door, artificial fish following, artificial fish preying, and variation of update strategy to complete self-renewal, resulting in a new artificial fish. IQAFSA is then applied to solve high dimensional and complex nonconvex programming and distributed network programming considering distributed generation. By simulation experiment among BAFSA, the global edition artificial fish swarm algorithm (GAFSA) has been put forward in [20] and quantum artificial fish swarm algorithm (QAFSA). The simulation results indicate that IQAFSA has great convergence and superiority in function optimization; meanwhile, they illustrate the validity and feasibility of IQAFSA applied to optimal configuration of DG in distributed network system.

The rest of this paper is organized as follows. We devote Section 2 to a discussion of those aspects of basic idea of BAFSA. The detail on quantum computing is then discussed in Section 3. The update of the quantum bit needs to make use of the transformation of quantum gates. In Section 4, the QAFSA has been formulated. The IQAFSA given in Section 5 is encoded by quantum bits and updated by quantum revolving gate. Artificial fish perform preying behavior and following behavior. In Section 6, the effectiveness of the algorithm compared with those of BAFSA, GAFSA, and QAFSA for three multidimensional and box constraints programming systems is demonstrated except for one two-dimensional box constraint programming. Section 7 deals with the distributed network modeling and the effect of DGs on system losses. Finally, Section 8 is devoted to drawing of the conclusions.

2. Description of BAFSA

Artificial fish complete update and obtain the optimal value mainly through the following four behaviors: being random, preying, swarming, and following in the process of iterative calculation.

2.1. AF-Random. Random behavior is to randomly select a new state x_{next} in its visual field and then move a step in the direction. It is actually a default behavior.

2.2. AF-Prey. Preying is a kind of the basic behavior of artificial fish, which move to the direction of food with high concentration. The AF-preying behavior is described as follows:

$$X_{i\text{next}} = \begin{cases} x_{ik} + \text{Random}(\text{step}) \frac{x_{jk} - x_{ik}}{\|X_j - X_i\|}, & Y(X_j) \text{ is better than } Y(X_i) \\ x_{ik} + \text{Random}(\text{step}), & \text{else,} \end{cases} \quad (1)$$

where x_{jk} is k th element of state vector; X_j is k th elements of X_i ; $x_{i\text{next}}$ is the next step state vector; $\text{Random}(\text{step})$ is a random number between 0 and step $k = 1, 2, \dots, n$.

2.3. AF-Swarm. AF-swarm refers to the fact that every fish moves to the center of the adjacent partners in the process of swimming as fast as possible and avoids overcrowding. Suppose that X_i is the current state of artificial fish i . n_f is the number of partners and X_c is the central position in the current field ($d_{ij} < \text{Visual}$). If $Y_c/n_f > \delta Y_i$, it shows that there are more food around the partner and its not too crowded. Then, the fish moves a step forward to the central position of this partner. The AF-swarming behavior is described as follows:

$$X_i^{t+1} = X_i^t + \frac{X_c - X_i^t}{\|X_c - X_i^t\|} \cdot \text{step} \cdot \text{Rand}(\cdot). \quad (2)$$

Otherwise, the fish performs preying behavior.

2.4. AF-Follow. AF-follow illustrates that each artificial fish moves to the current optimal direction in the range of vision. Suppose that X_i is the current state of artificial fish i . Y_j is the greatest partner in the current field ($d_{ij} < \text{Visual}$). If $Y_j/n_f > \delta Y_i$, it turns out to be that there are more food around X_j while being not too crowded. Then, the fish moves a step forward to the direction of X_j . The AF-following behavior is described as follows:

$$X_i^{t+1} = X_i^t + \frac{X_j - X_i^t}{\|X_j - X_i^t\|} \cdot \text{step} \cdot \text{Rand}(\cdot). \quad (3)$$

Otherwise, the fish performs preying behavior.

In the problem of function optimization. Suppose that in a D -dimensional search space goal, the number of artificial fish is n . The state of the individual artificial fish can be denoted by vector $X = (x_1, x_2, \dots, x_n)$ and x_i ($i = 1, 2, \dots, n$) which is the optimization variable. The food concentration of the artificial fish in the current location can be expressed as $Y = f(X)$, where Y denotes the objective function value. $d_{ij} = \|X_i - X_j\|$ is the distance between the fish i and the fish j . The idea of BAFSA is that it firstly initializes a swarm of artificial fish randomly; secondly the fish complete update by four basic behaviors mentioned above. Each of the artificial fish explores the current environment conditions (including

the change of the objective function and partners). Then, select an appropriate behavior and move in the direction of optimal areas. Finally, artificial fish gather around several local optimal, especially in some better optimal areas, which are generally able to rally more artificial fish, namely, the optimal value of objective function.

3. Quantum Computing

Physical media acting as the information storage unit are a two-state quantum systems in quantum computing, to be referred to as quantum bit. A quantum bit can be in a state of quantum superposition represented by $|\varphi\rangle = \alpha|0\rangle + \beta|1\rangle$, where α and β denote quantum bit probability amplitude. The probability of $|0\rangle$ is represented by $|\alpha|^2$, and the probability of $|1\rangle$ is represented by $|\beta|^2$; furthermore, $|\alpha|^2 + |\beta|^2 = 1$. Each quantum state in system can be expressed as the superposition of the basic state; as $|\alpha|^2$ and $|\beta|^2$ approach to be 1 or 0, the quantum bit will converge to a specific state.

The update of the quantum bit needs to make use of quantum gates which include the xor gate, the controlled xor gate, Hadamard transform gate, and the revolving gate. The quantum revolving gate [21, 22] was adopted in this paper to change its phase by interference and the basic state probability amplitude. The revolving gate is described as follows:

$$U(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}, \quad (4)$$

where θ denotes the rotation angle of the quantum gate; the update process is as follows:

$$\begin{bmatrix} \alpha' \\ \beta' \end{bmatrix} = U(\theta) \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix}. \quad (5)$$

The rotation angle of quantum gate is adjusted dynamically with the evolutionary process, whilst rotation angle is set larger at the beginning of the algorithm. With the increase of

evolution generation, the rotation angle is gradually reduced by formula $\theta = k \cdot s(\alpha_i, \beta_i)$, where k is a coefficient influencing the speed and $s(\alpha_i, \beta_i)$ is the direction of rotation angle. Emphasis here is on the fact that if k is very large, the search step length will be very big in each iteration in the process of calculation. It is easy to fall into local optimal value. On the other hand, if the convergence speed is slow, the computation time will be too long. In this paper, $k = 10 \cdot \exp(-t/T)$, where t denotes the current evolution generation, and T is the biggest evolution generation. $s(\alpha_i, \beta_i)$ is to make the algorithm get the optimal solution, the principle of which is to use the current solution to approach to the best solution best, determining the direction of rotation of the quantum revolving gate. An adjusting strategy which is general and has nothing to do with the problem [23] is adopted in this paper. These strategies are described in Table 1.

4. Description of QAFSA

The artificial fish is encoded with quantum bits in QAFSA. It mainly uses update strategy of quantum revolving gate for self-renewal to get new artificial fish. Compared with BAFSA, the diversity of its fish becomes richer. Despite the small scale of the fish, it does not affect the convergence of the algorithm. At the same time, the algorithm has faster convergence speed as well as higher accuracy.

Suppose that there is a quantum of artificial fish $Q(t) = \{q_1^t, q_2^t, \dots, q_n^t\}$, where n denotes the population size, t is evolution generation, and q_j^t represents a quantum artificial fish, which is encoded by quantum bit as follows:

$$q_j^t = \begin{bmatrix} \alpha_1^t & \alpha_2^t & \dots & \alpha_m^t \\ \beta_1^t & \beta_2^t & \dots & \beta_m^t \end{bmatrix}, \quad j = 1, 2, \dots, n, \quad (6)$$

where m denotes the quantum number. An artificial fish encoding with multiple quantum bits can be represented as [19]

$$q_j^t = \begin{bmatrix} \alpha_{11}^t & \alpha_{12}^t & \dots & \alpha_{1k}^t & \alpha_{21}^t & \alpha_{22}^t & \dots & \alpha_{2k}^t & \dots & \alpha_{m1}^t & \alpha_{m2}^t & \dots & \alpha_{mk}^t \\ \beta_{11}^t & \beta_{12}^t & \dots & \beta_{1k}^t & \beta_{21}^t & \beta_{22}^t & \dots & \beta_{2k}^t & \dots & \beta_{m1}^t & \beta_{m2}^t & \dots & \beta_{mk}^t \end{bmatrix}, \quad (7)$$

where m is variable dimension of artificial fish optimal control, k is quantum bit number of optimal control variables, α_{xy} and β_{xy} ($1 \leq x \leq m$, $1 \leq y \leq k$) are complex constants, and $|\alpha_{xy}|^2 + |\beta_{xy}|^2 = 1$.

The quantum bits coding (α, β) of each individual is initialized to $(1/\sqrt{2}, 1/\sqrt{2})$. A measurement is carried out for the initialization of the artificial fish in order to obtain a set of definite solution $P(t) = p_1^t, p_2^t, \dots, p_n^t$, where p_j^t is j th solution in t th generation population, which is performed by binary string of length m , each of which is 0 or 1 and gotten by the probability of quantum bits. The measurement

process is randomly generating a number between 0 and 1. If this number is greater than $|\alpha_i^t|^2$, the result value is 1, otherwise 0. In each iteration, firstly, measure the population to get a set of determinate solutions $P(t)$, and then calculate the fitness value of each solution. Using quantum revolving gate to adjust individual in the population to obtain the updated population, record the current optimal solution, and compare with the current target value. If it is better than the current target, let the new optimal solution be a target for the next iteration. Otherwise, keep the current target value the same.

TABLE 1: Changing the angle value.

| best _{<i>i</i>} | $s(\alpha_i, \beta_i)$ | | | |
|--------------------------|------------------------|------------------------|----------------|---------------|
| | $\alpha_i \beta_i > 0$ | $\alpha_i \beta_i < 0$ | $\alpha_i = 0$ | $\beta_i = 0$ |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |
| 1 | +1 | -1 | 0 | ± 1 |
| 1 | -1 | +1 | ± 1 | 0 |
| 0 | -1 | +1 | ± 1 | 0 |
| 0 | +1 | -1 | 0 | ± 1 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 |

5. Description of IQAFSA

Despite the fact that QAFSA can reach high precision and convergence speed, it may get some inferior solutions while producing better one. The fish diversity is not very rich and the convergence speed can be further improved. Therefore, the IQAFSA given in this paper is encoded by quantum bits and updated by quantum revolving gate. Make it perform preying behavior and following behavior. And swap quantum probability amplitude of artificial fish having poor results can realize the variation, and then the optimal solution is obtained [24].

At the early stage of the quantum artificial fish optimization, each time the artificial fish has achieved an update, it will turn into the implementation of the tailgating behavior of AFSA, which can improve the convergence speed of artificial fish. At the late stage of the artificial fish quantum optimization, each time the artificial fish has realized an update, it will turn into the implementation of the preying behavior of AFSA, which can improve the accuracy of optimization. Having been updated by quantum revolving gate, preying behavior, and following behavior, for those artificial fish having poor optimization results, it can enrich the diversity of the artificial fish by swapping the quantum bit probability amplitude value, namely, (α, β) , and reversing the individual evolutionary direction as a whole, which can prevent the individual evolution falling into a local optimum. Finally, the variation of artificial fish is brought about. Here a point that should be stressed is that the *Pauli-X* gate is used below to realize mutation operation:

$$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \beta \\ \alpha \end{bmatrix}. \quad (8)$$

Because the quality of preying and following behavior has close relation with vision and step of artificial fish. The results in the literature [22] show that the larger range of vision is, the stronger global search ability of artificial fish is and faster convergence speed is. On the contrary, the smaller range of vision is, the stronger local search ability of artificial fish is. Whilst the larger size of step is, the faster convergence speed is; however, it will turn out to be an oscillation phenomenon. On the other hand, the smaller size of step is, the faster convergence speed is, but there will be the high precision

of the solution. Consequently, the dynamic adjustments of artificial fish vision and step size are as follows:

$$\begin{aligned} \text{visual} &= \text{visual} \cdot a + \text{visual}_{\min}, \\ \text{step} &= \text{step} \cdot a + \text{step}_{\min}, \end{aligned} \quad (9)$$

$$a = \exp \left[-30 \cdot \left(\frac{t}{T} \right)^s \right].$$

IQAFSA steps are as follows.

Step 1. Initialize the quantum of artificial fish $Q(t_0) = q_1^{t_0}, q_2^{t_0}, \dots, q_n^{t_0}$ in the feasible region, and all optimal control variables $(\alpha_i^{t_0}, \beta_i^{t_0})$ of the artificial fish populations are initialized to $(1/\sqrt{2}, 1/\sqrt{2})$. Set up parameters such as the artificial fish vision, step length, crowded degree, maximum attempts, and evolution generation.

Step 2. Calculate the fitness of each artificial fish in the population $Q(t)$; the determinate solution $P(t)$ can be calculated.

Step 3. Conduct fitness evaluation for the determinate solutions and put the optimal artificial fish and its fitness value in the bulletin board.

Step 4. On the basis of the above adjustment strategy, firstly use the quantum revolving gate $U(\theta)$ and the corresponding rotation angle adjustment policy to update quantum bit of artificial fish $Q(t+1)$, and perform preying behavior and following behavior for artificial fish through the fish evolution process. Finally, new artificial fish is gotten.

Step 5. Perform mutation for artificial fish, choose q ($q < n$) artificial fish having minimum fitness, and update its quantum bit probability amplitude by using formula (6).

Step 6. If evolution generation achieves the maximum, then output the result on the bulletin board as the final optimal solution; otherwise, increase the iteration number by $g = g+1$ and go back to Step 2.

6. Simulation Experiment and Analysis

In order to verify the feasibility and effectiveness of IQAFSA, four typical function simulations are proposed compared with QAFSA, BAFSA, and GAFSA in the literature [20]:

E.g. 1 [20]:

$$\min f_1(x) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10] \quad (10)$$

$$\text{s.t.} \quad -10 < x_i < 10$$

E.g. 2 [20]:

$$\min f_2(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos \left(\frac{x_i}{\sqrt{i}} \right) + 1 \quad (11)$$

$$\text{s.t.} \quad -100 < x_i < 100$$

TABLE 2: (a) Results of simulation with BAFSA and GAFSA. (b) Results of simulation with QAFSA and IQAFSA.

| (a) | | | | | | | | |
|-------|-----|---------|------------|------------|----------|------------|------------|----------|
| fun | dim | max T | BAFSA | | | GAFSA | | |
| | | | A_{\min} | G_{\min} | σ | A_{\min} | G_{\min} | σ |
| f_1 | 3 | 1000 | 1.49 | $1.303e-4$ | 0.62 | 1.49 | $6.526e-6$ | 2.01 |
| | 5 | 1000 | 9.391 | 5.434 | 2.20 | 2.9351 | 0.995 | 1.81 |
| | 10 | 1500 | 50.88 | 36.7075 | 7.82 | 20.844 | 2.985 | 12.1 |
| f_2 | 3 | 1000 | 0.007 | $8.301e-4$ | $4e-3$ | $8.5e-3$ | $2.99e-10$ | $7e-3$ |
| | 5 | 1000 | 0.0332 | $2.24e-2$ | $4e-3$ | $2.5e-2$ | $1.088e-8$ | 0.02 |
| | 10 | 1500 | 0.1584 | 0.1162 | 0.03 | 3.2433 | 2.3018 | 0.73 |
| f_3 | 3 | 1000 | 0.07 | $1.7e-4$ | 0.31 | 0.06 | $2.92e-7$ | 0.27 |
| | 5 | 1000 | 0.267 | $8.91e-4$ | 0.60 | 0.93 | $6.9e-6$ | 0.98 |
| | 10 | 1500 | 2.6388 | 2.1013 | 0.21 | 1.64 | 0.0011 | 0.93 |
| f_4 | 2 | 1000 | 0.0053 | $2.4e-4$ | 0.01 | 0.0039 | $6.3e-15$ | 0.05 |

| (b) | | | | | | | | |
|-------|-----|---------|------------|------------|----------|------------|------------|----------|
| fun | dim | max T | QAFSA | | | IQAFSA | | |
| | | | A_{\min} | G_{\min} | σ | A_{\min} | G_{\min} | σ |
| f_1 | 3 | 1000 | $5.013e-6$ | $5.413e-8$ | 0 | $5.413e-8$ | $5.413-8$ | 0 |
| | 5 | 1000 | 0.3786 | $9.022e-8$ | 1.35 | 0.3786 | $9.022e-8$ | 1.35 |
| | 10 | 1500 | 10.154 | 2.701 | 8 | 9.4 | $1.804e-7$ | 5.7 |
| f_2 | 3 | 1000 | $8.34e-11$ | $8.3e-11$ | 0 | $8.34e-11$ | $8.3e-11$ | 0 |
| | 5 | 1000 | $6.074e-3$ | $1.0e-10$ | 0.016 | $6.074e-3$ | $1.0e-10$ | 0.016 |
| | 10 | 1500 | 3.211 | 2.121 | 0.21 | 0.1374 | $1.3e-14$ | 0.096 |
| f_3 | 3 | 1000 | $3.72e-8$ | $3.7e-8$ | 0 | $3.72e-8$ | $3.7e-8$ | 0 |
| | 5 | 1000 | $3.72e-8$ | $3.7e-8$ | 0 | $3.72e-8$ | $3.7e-8$ | 0 |
| | 10 | 1500 | 0.356 | 0.0128 | 0.53 | 0.0226 | $3.7e-8$ | 0.0278 |
| f_4 | 2 | 1000 | $0.9e-10$ | $1.6e-16$ | 0 | $1.6e-16$ | $1.6e-16$ | 0 |

E.g. 3 [25]:

$$\begin{aligned}
\min \quad & f_3(x) \\
& = -20 \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}\right) \\
& \quad - \exp\left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)\right) + 20 + e \\
\text{s.t.} \quad & -32 < x_i < 32
\end{aligned} \tag{12}$$

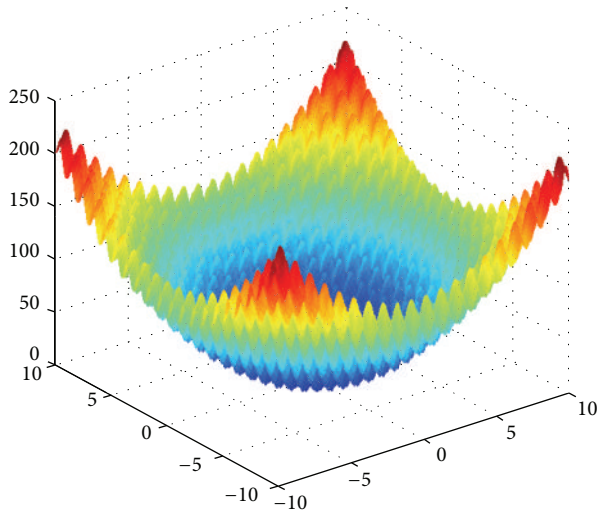
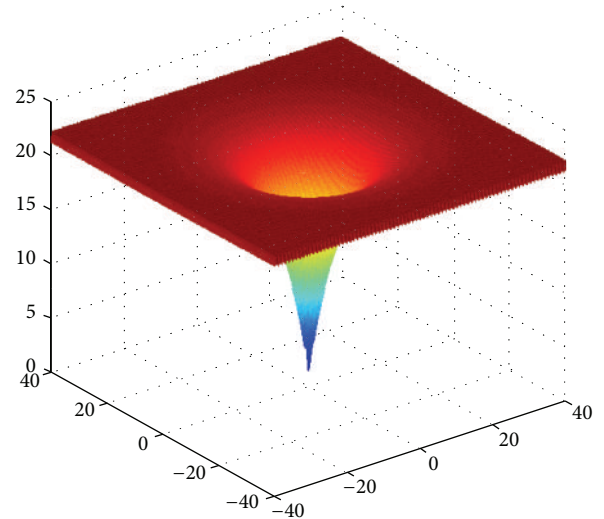
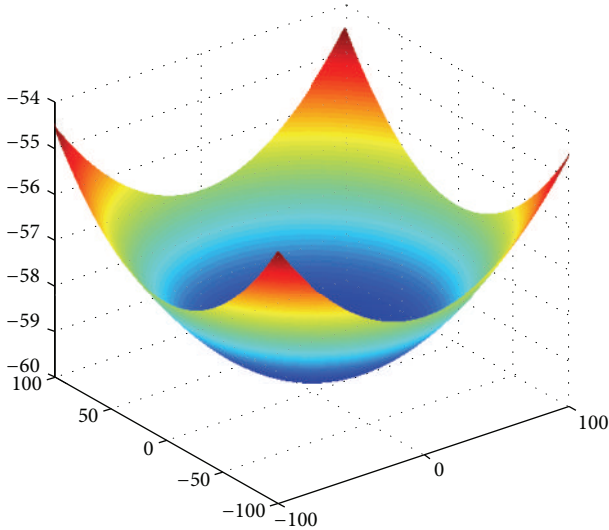
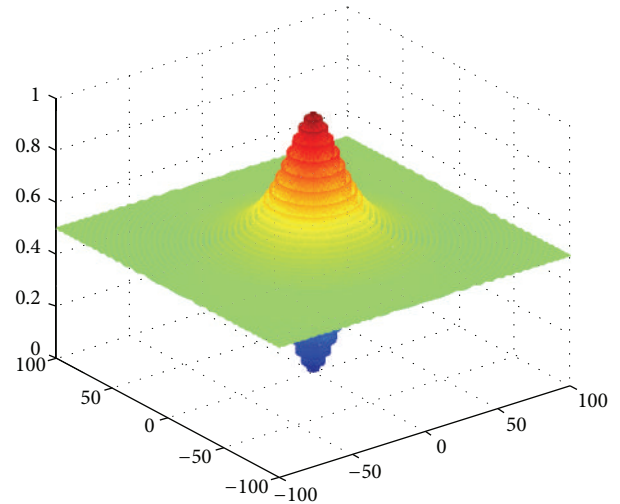
E.g. 4 [20]:

$$\begin{aligned}
\min \quad & f_4(x_1, x_2) = 0.5 + \frac{\sin^2 \sqrt{x_1^2 + x_2^2} - 0.5}{[1 + 0.001(x_1^2 + x_2^2)]^2} \\
\text{s.t.} \quad & -100 < x_1, x_2 < 100.
\end{aligned} \tag{13}$$

We can find the optimal value to be 0 through the analysis for function expression and the function image shown in Figures 1, 2, 3, and 4. The theoretical minimum value of these four multidimensional functions are 0 actually, where $f_4(x)$ is a two-dimensional function; other functions

are all multidimensional functions. Use BAFSA, GAFSA, QAFSA, and IQAFSA to get minimum value of four different functions, respectively. Parameter settings are set as follows: fishnum = 40, try-number = 9, visual = 1, $\delta = 0.618$, $\text{step}_{\min} = 0.0002$, $s = 2$, $\text{step} = 0.5$, and $\text{visual}_{\min} = 0.001$. The maximum iteration is set to be 1000 times and 1500 times. All algorithms run 50 times from different random initial population, respectively. The global average minimum value, the global minimum value, and standard deviation are to be as a measurement of the performance of algorithm, and the results are shown from Tables 2(a) and 2(b), where fun denotes function and dim denotes dimension; max T is maximum evolution generation, A_{\min} is average minimum, G_{\min} is global minimum value, and δ is standard deviation, respectively.

This paper uses BAFSA, GAFSA, QAFSA, and IQAFSA for 50 times and obtains the average of the evolutionary curve and draws Figures 5–14 of different dimension functions f_1 , f_2 , f_3 , and f_4 . In Table 2, the average minimum value, the global minimum value, and the global minimum of the four functions have been obtained under different dimensions and number of iterations by using BAFSA, GAFSA, QAFSA, and IQAFSA. In each figure, the ordinate is expressed by natural logarithm of optimal average of the function (i.e., $\ln(f)$), and the abscissa is expressed by evolution generation; four different kinds of lines show the changing tendency of

FIGURE 1: Image of function $f_1(x)$.FIGURE 3: Image of function $f_3(x)$.FIGURE 2: Image of function $f_2(x)$.FIGURE 4: Image of function $f_4(x)$.

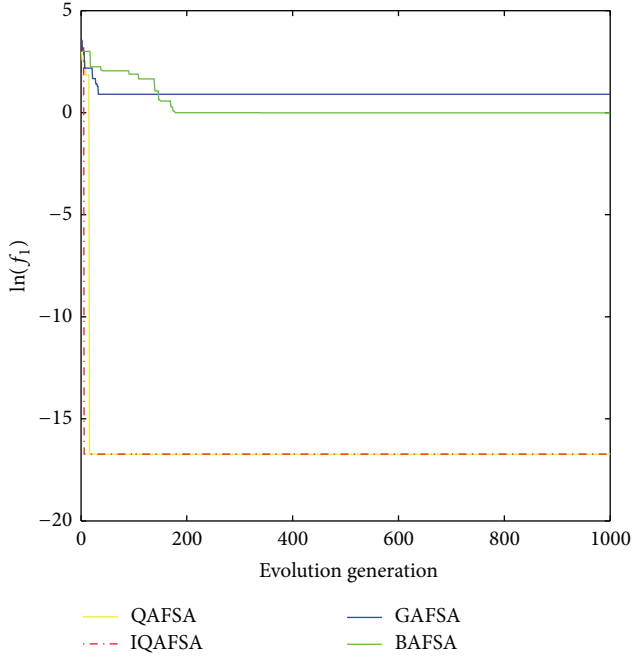
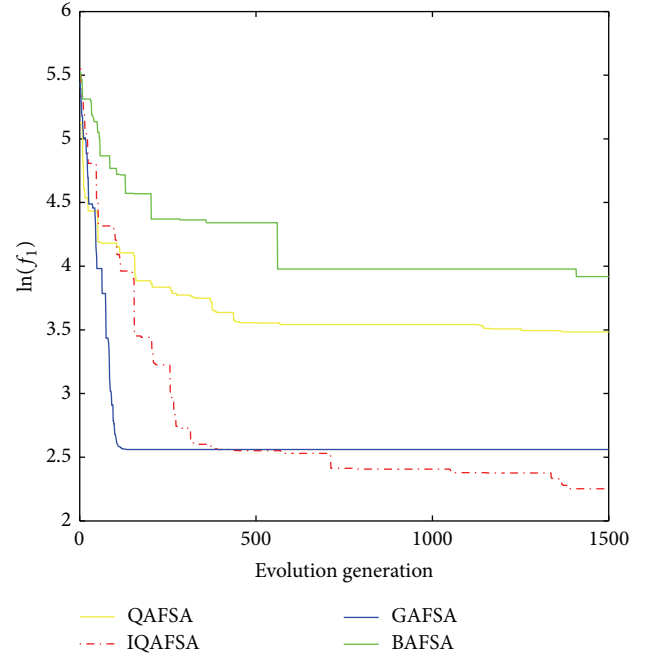
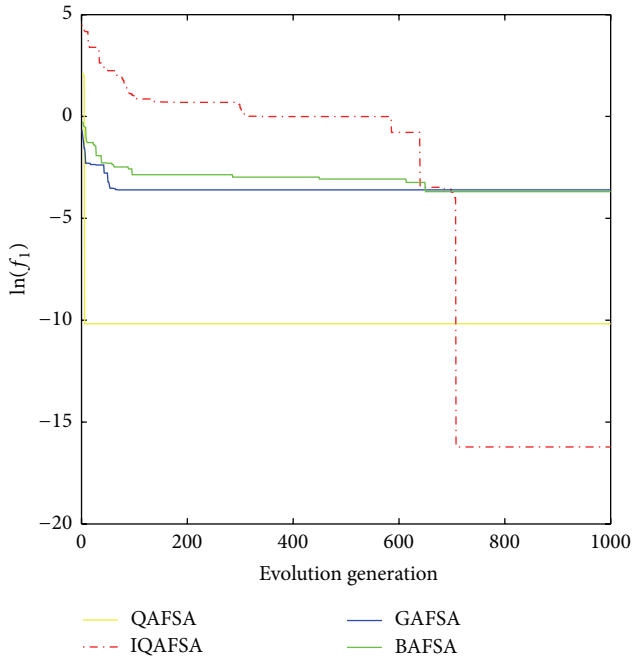
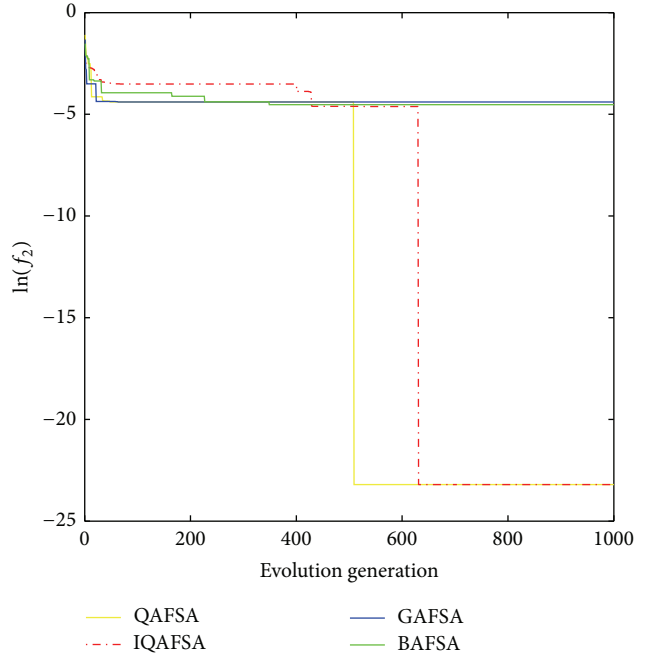
minimum value obtained by BAFSA, GAFSA, QAFSA, and IQAFSA with the increase of the number of iterations. From Table 2, it is clearly seen that the average optimization results and the optimal results of IQAFSA are much better than those of GAFSA, BAFSA, and QAFSA in different dimensions and number of iterations of each function. Meanwhile, the precision of the optimization results is greater than GAFSA, BAFSA, and QAFSA. Furthermore, most of standard deviations of IQAFSA are almost 0. In addition, from each figure, it is quite evident that stability of IQAFSA is better than those of GAFSA, BAFSA, and QAFSA by comparing the standard deviation. With the increase of the dimension, the effect becomes more obvious compared with other algorithms. For example, the standard deviation of function f_1 with 3D is 0 obtained by IQAFSA, but the others are usually bigger than 0. At the same time, in all figures, each optimal value is in the form of a logarithm in order to broaden the differences

between different algorithms. From each figure, it is easily seen that convergence speed and optimization precision of IQAFSA are obviously better than those of GAFSA, BAFSA, and QAFSA. For example, Figure 8 shows that the optimal values obtained in the quickest way by IQAFSA are better than any other optimal values by other algorithms.

7. Distributed Network Considering DG

7.1. Formulation of Objective Function. The optimal configuration of the distribution networks with DG units is a nonlinear optimization problem. In this paper, the objective function of the nonlinear model is formulated with the aim of finding the minimum power loss. Thus, on the condition that the constraints below are considered, the objective function of the reconstruction is defined as

$$\min f = \sum_{i=1}^n \frac{P_i^2 + Q_i^2}{U_i^2} \cdot R_i, \quad (14)$$


 FIGURE 5: Average min evolution curve 3D function of $f_1(x)$.

 FIGURE 7: Average min evolution curve 10D function of $f_1(x)$.

 FIGURE 6: Average min evolution curve 5D function of $f_1(x)$.

 FIGURE 8: Average min evolution curve 3D function of $f_2(x)$.

where P_i is real power load and Q_i is reactive power load of the node i , respectively. U_i is voltage of node i , and R_i is resistance of branch i , whilst n is the number of branches.

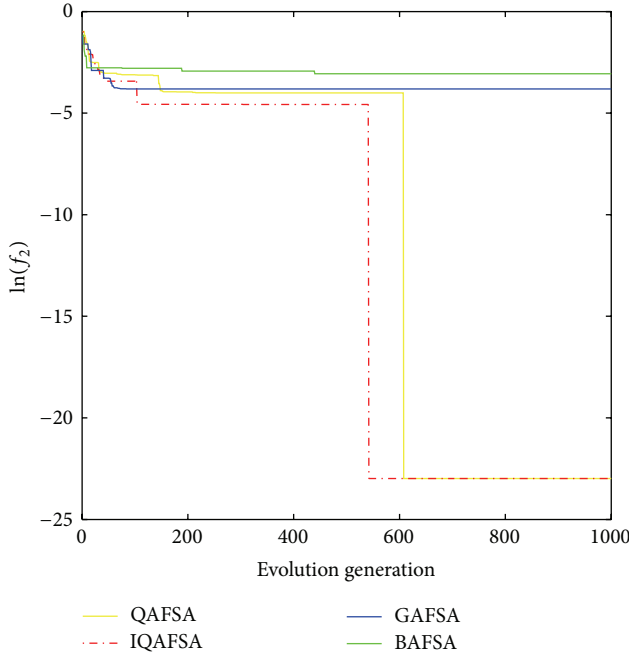
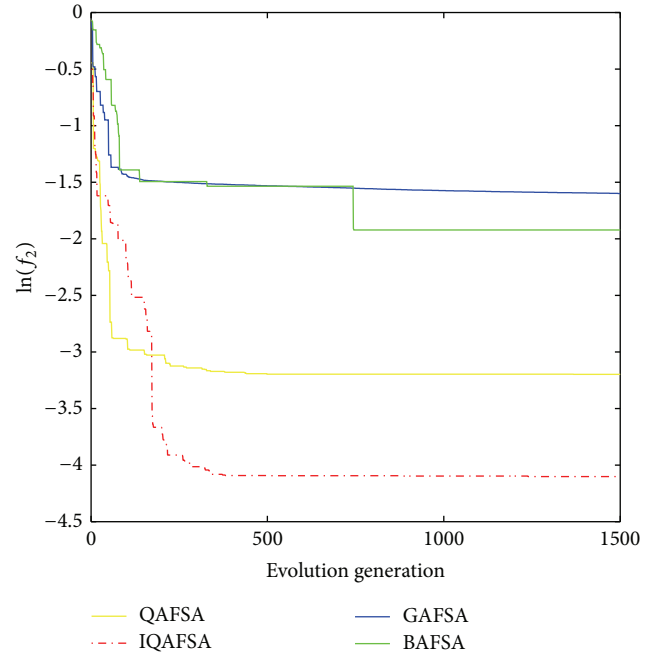
7.2. Constraints

(a) Voltage amplitude constraints are

$$U_{\min} \leq U_i \leq U_{\max}. \quad (15)$$

(b) Active power and reactive power constraints of the DG units are

$$\begin{aligned} P_{DG_i}^{\min} &\leq P_{DG_i} \leq P_{DG_i}^{\max}, \\ Q_{DG_i}^{\min} &\leq Q_{DG_i} \leq Q_{DG_i}^{\max}. \end{aligned} \quad (16)$$

FIGURE 9: Average min evolution curve 5D function of $f_2(x)$.FIGURE 10: Average min evolution curve 10D function of $f_2(x)$.

(c) Current constraints are

$$I_i^{\min} \leq I_i \leq I_i^{\max}. \quad (17)$$

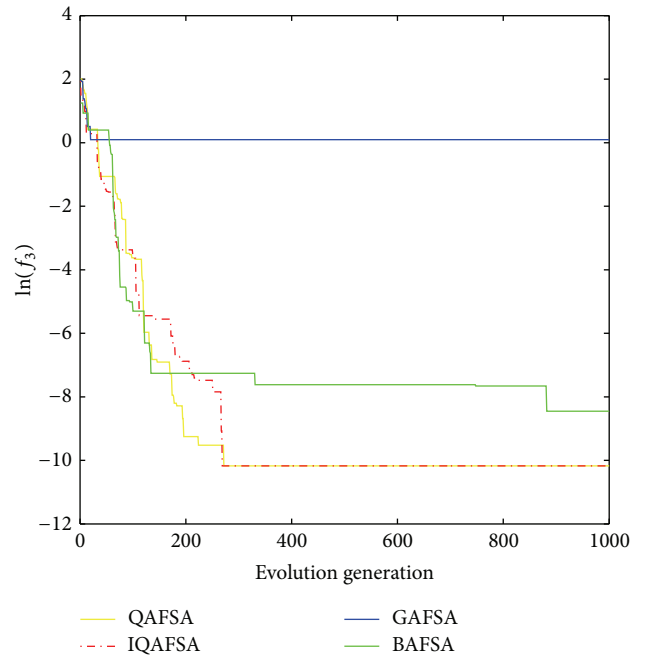
(d) Power flow equation constraints are

$$P_i = V_i \sum_{j=1}^n V_j (G_{ij} \cos \delta_{ij} + B_{ij} \sin \delta_{ij}), \quad (18)$$

$$Q_i = V_i \sum_{j=1}^n V_j (G_{ij} \sin \delta_{ij} - B_{ij} \cos \delta_{ij}),$$

where U_i is voltage of node i , U_{\min} is the minimum, and U_{\max} is the maximum of the node voltage, respectively; P_{DG_i} is active power and Q_{DG_i} is reactive power of DG_i , respectively; $P_{DG_i}^{\min}$ is the minimum and $P_{DG_i}^{\max}$ is the maximum of the active power of DG_i , whilst $Q_{DG_i}^{\min}$ is minimum of the reactive power of DG_i and $Q_{DG_i}^{\max}$ is the maximum. I_i is the current of the node i , I_{\min} is the minimum, and I_{\max} is the maximum of node current separately. G_{ij} is conductance and B_{ij} is susceptance between node i and node j . δ_{ij} is the voltage phase angle difference.

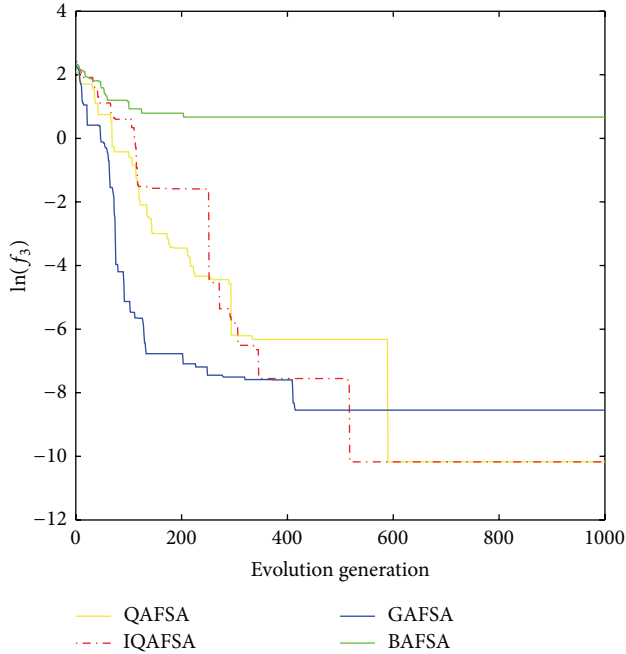
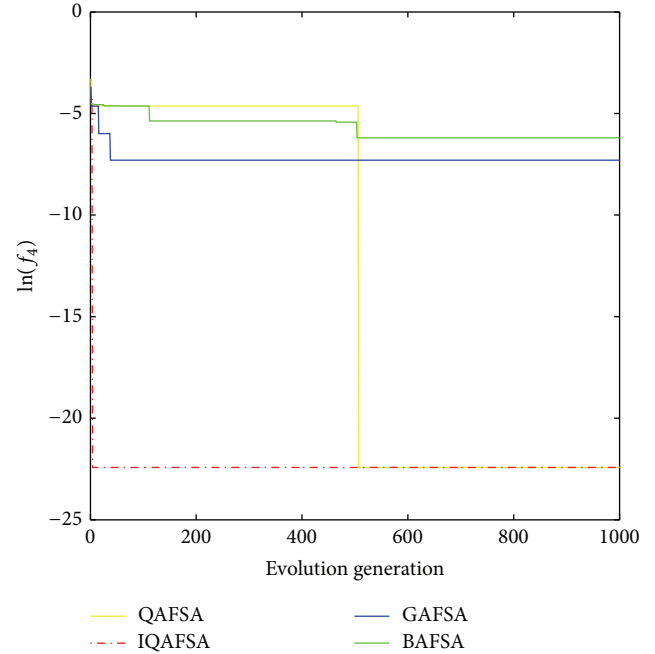
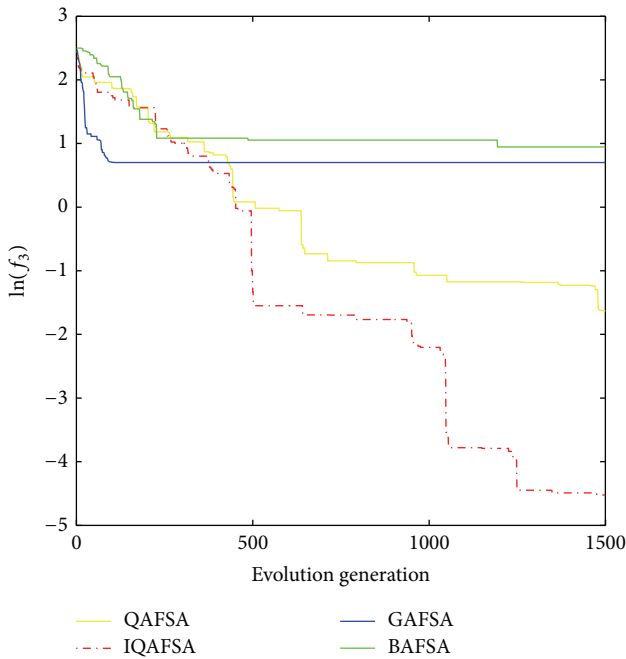
7.3. Handling of the DG in Calculating Load Flow. DG installed on the radial power distribution network can be simplified to a PV node, PQ node, or PI node. In this paper, DG is considered as a PQ node which has constant power factor. The characteristic of DG makes it an optional and alternative power supply normally so that it is usually installed on the location of the load. Assume that placements of all DG units are in the neighborhood of the load instead of line in the present paper. So, the basic configuration of

FIGURE 11: Average min evolution curve 3D function of $f_3(x)$.

the distribution network after installing DG is shown in Figure 15.

According to the comparison of active powers from the load and DG, there are three kinds of situations between load and distributed network when it comes to active power flow.

- (1) If P_{DG_i} is greater than P_{L_i} , the load of the node i can be considered as power supply outputting $P_{DG_i} - P_{L_i}$


 FIGURE 12: Average min evolution curve 5D function of $f_3(x)$.

 FIGURE 14: Average min evolution curve 3D function of $f_4(x)$.

 FIGURE 13: Average min evolution curve 10D function of $f_3(x)$.

to distributed network, namely, to form reverse power flow.

- (2) If P_{DG_i} is equal to P_{L_i} , there is no active flow between distributed network and load.
- (3) If P_{DG_i} is less than P_{L_i} , the distributed network provides load with the active power of $P_{L_i} - P_{DG_i}$.

The route is shown in Figure 16.

TABLE 3: Suitable size for BAFSA.

| Location of DG | 16 | 17 | 32 |
|-----------------------|-------|-------|-------|
| Active power of DG/kW | 416.6 | 211.9 | 965.1 |
| Size of DG/kW | 500 | 300 | 1000 |

7.4. Simulation. IEEE33 nodes distributed network system [26] is adopted to take a simulation test in our work as shown in Figure 17. The rated voltage, total active power, and reactive power of system are 12.26 kV, 3715 kW, and 2300 kvar [27], respectively. Concurrently the active power loss is 201.715 kW and reactive power loss is 148.225 kW without DG units. The voltage amplitude of the system is shown in Figure 18. To verify feasibility and stability of IQAFSA and compare with BAFSA, GAFSA, and QAFSA, it is assumed that locations of DG units are fixed due to certain constraints such as power resources stability; buses 16, 17, and 32 whose voltage qualities are the lower are selected as DG location, and all of them are operated in a constant power factor mode, also known as the PQ node. It is equal to negative load which is installed in the neighborhood of the load.

Considering that decrease in total power loss depends on size and location of DG in this paper, four kinds of algorithms were used to determine the optimal size of DG on the condition that power loss of system is minimum. Finally, results were compared with each other. Some parameters are set as follows: fishnum = 40, try-number = 9, visual = 1, $\delta = 0.618$, step = 0.5, $\text{visual}_{\min} = 0.001$, $\text{step}_{\min} = 0.0002$, $s = 2$. The max evolutionary generations were set to 500 times and results were gotten in Tables 3–7.

Tables 3, 4, 5, and 6 provide simulation results carried out on the test network given in Figure 17, using BAFSA, GAFSA, QAFSA, and IQAFSA algorithms, respectively. Simulation

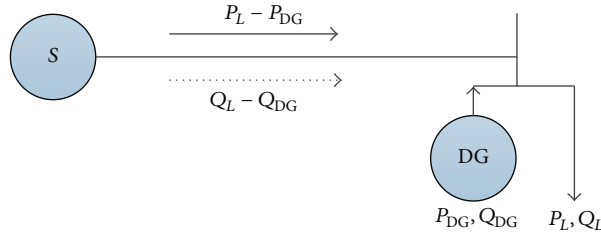


FIGURE 15: Basic configuration of distribution network after installing DG.

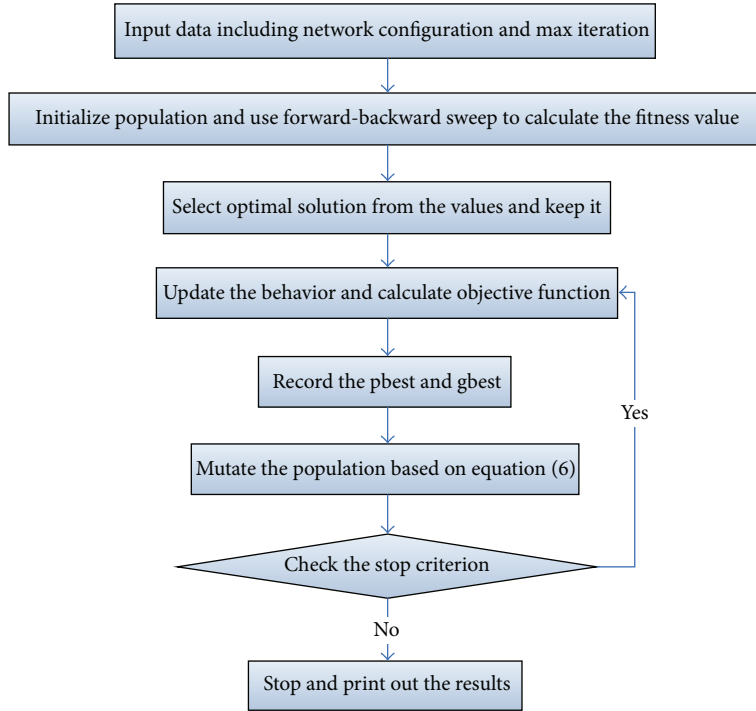


FIGURE 16: Flowchart of the proposed IQAFSA considering DG.

TABLE 4: Suitable size for GAFSA.

| Location of DG | 16 | 17 | 32 |
|-----------------------|-------|------|-------|
| Active power of DG/kW | 574.1 | 93.1 | 938.5 |
| Size of DG/kW | 400 | 300 | 1000 |

TABLE 5: Suitable size for QAFSA.

| Location of DG | 16 | 17 | 32 |
|-----------------------|-------|-------|--------|
| Active power of DG/kW | 187.5 | 500.0 | 1000.0 |
| Size of DG/kW | 300 | 500 | 1000 |

TABLE 6: Suitable size for IQAFSA.

| Location of DG | 16 | 17 | 32 |
|-----------------------|-------|-------|------|
| Active power of DG/kW | 441.1 | 250.0 | 1000 |
| Size of DG/kW | 500 | 300 | 1000 |

TABLE 7: Power losses and processing time.

| | BAFSA | GAFSA | QAFSA | IQAFSA |
|-------------------|-------|-------|-------|--------|
| Power loss/kW | 46.0 | 45.4 | 45.9 | 45.0 |
| Processing time/s | 3.10 | 1.91 | 1.20 | 0.02 |

Tables 3–6 that the sizes of DG units are maintained within permitted range of tolerance. Table 7 gives the system losses and processing time with four kinds of algorithms, and the minimum obtained by IQAFSA after comparing results with each other. Therefore, the minimum system loss and processing time are 45.0 kW and 0.02 s corresponding to the fact that outputs of buses 16, 17, and 32 are 441.4 kW, 250 kW, 1000 kW, respectively. It is shown that the optimal placement of DG units in system caused a larger reduction in power losses and less time with IQAFSA. In consideration of realistic factors, the sizes of buses 16, 17, and 32 were set to 500 kW, 300 kW, and 1000 kW.

results are obtained from MATLAB and include the best solutions, as provided in Table 7. It can be observed from

Convergence curves for four algorithms can also be used to show the capability and the speed of algorithm in obtaining

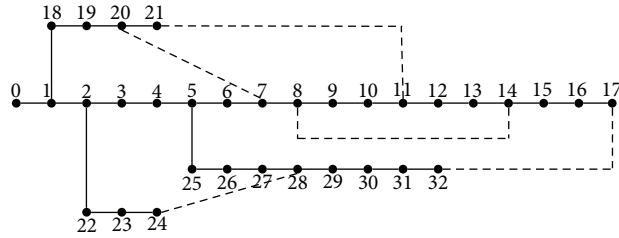


FIGURE 17: A 33-bus radial distribution network system.

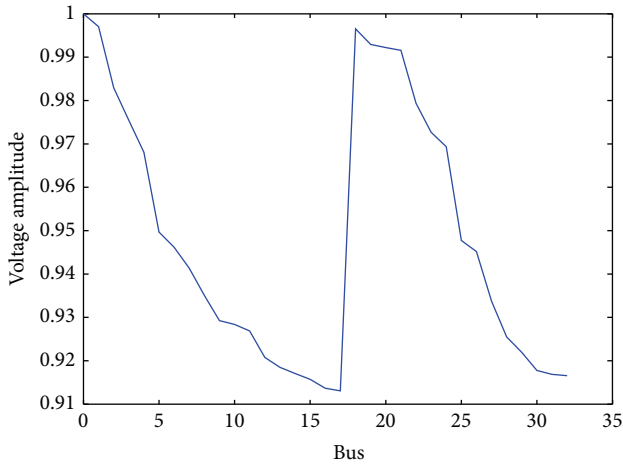


FIGURE 18: Voltage amplitude of IEEE33 nodes system.

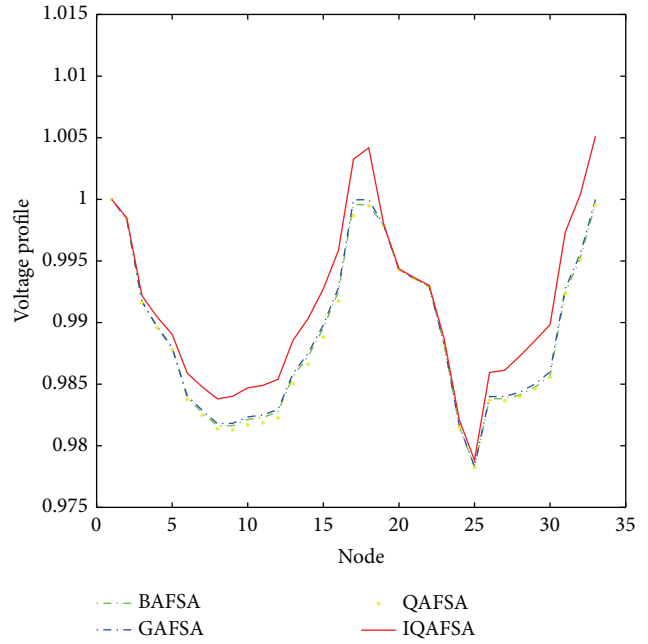


FIGURE 20: Voltage level comparison on IEEE33 nodes system.

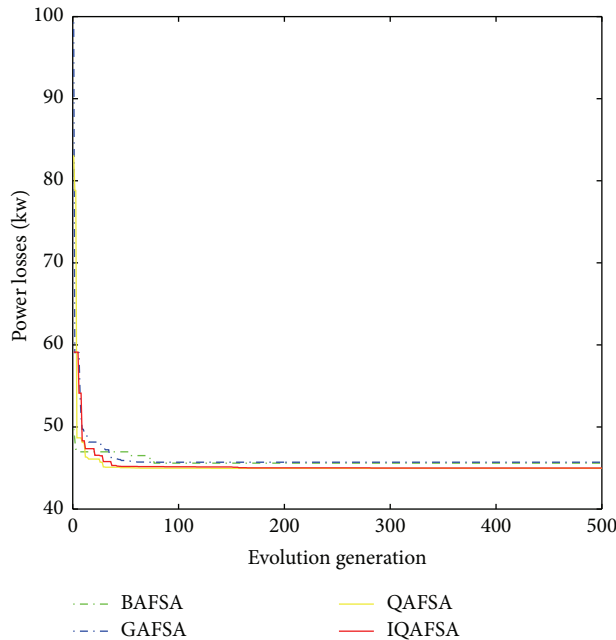


FIGURE 19: Convergence curves for four algorithms.

the optimal results. By taking the best results that are given by four algorithms and plotting their best fitness value in the population for each iteration as shown in Figure 19, IQAFSA gives the faster results in reaching the global optimal solution and stayed there till the end of iterations. Furthermore, other

algorithms failed to provide the global minimum and only reach the local minimum. Hence, it can be concluded that the proposed method in this paper results in a superior performance as compared to the existing methods such as BAFSA and QAFSA and also numerically converges more quickly than the traditional methods.

Figure 20 shows the improvement in voltage profile under load models. As shown in Figure 20, the voltage at all buses after inserting DG units in system is higher than before, especially in the place of the bus where DG is located.

8. Conclusion

In this paper, quantum computing is introduced on basis of artificial fish algorithm, and the improved quantum artificial fish algorithm is put forward. In the early stage of the artificial fish optimization, following behavior and preying behavior of artificial fish are added in the late stage, which improve convergence speed and optimization precision and enhance the diversity of population by implementation of the mutation after each iteration. Simulation results show that compared with GAFSA, BAFSA, and QAFSA, IQAFSA is more suitable for solving high dimensional and complex

nonconvex programming. So the algorithm is more convergent. Finally, IQAFSA is applied to optimal configuration of DG in distributed network system and simulation results indicate that IQAFSA has great validity and feasibility. The theory research of IQAFSA is still in its infancy. In the future, using more test functions with higher dimension to verify performance of the proposed method is necessary. Convergence and stability of the algorithm remains to be proven so that more detailed work and further research from the theory are needed to expand.

Conflict of Interests

The authors declare no conflict of interests.

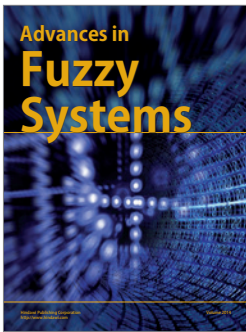
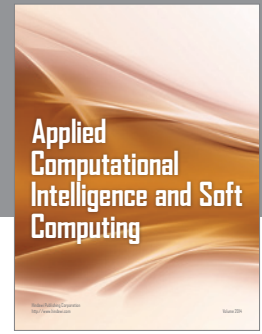
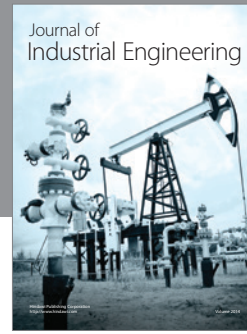
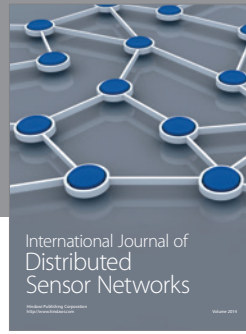
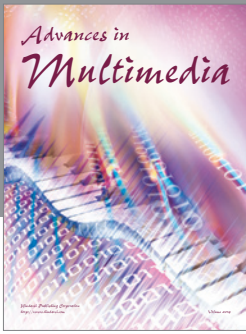
Acknowledgments

This work was supported by the National Natural Science foundation of China under Grant 61374028, Hubei Province Key Laboratory of Systems Science in Metallurgical Process of China under Grant Z201402, and the Natural Science Foundation of Hubei Province, China, under Grant 2013CFA131. The authors sincerely appreciate the reviewers for their careful reading and significant comments which have resulted in the present improved version of the original paper. At the same time, we also thank the editors for their valuable suggestions which were vital to improve the presentation of this paper.

References

- [1] P. Alemi and G. B. Gharehpetian, "DG allocation using an analytical method to minimize losses and to improve voltage security," in *Proceedings of the IEEE 2nd International Power and Energy Conference (PECon '08)*, pp. 1575–1580, Johor Bahru, Malaysia, December 2008.
- [2] A. Swarnkar, N. Gupta, and K. R. Niazi, "Adapted ant colony optimization for efficient reconfiguration of balanced and unbalanced distribution systems for loss minimization," *Swarm and Evolutionary Computation*, vol. 1, no. 3, pp. 129–137, 2011.
- [3] J. Olamaei, T. Niknam, and G. Gharehpetian, "Application of particle swarm optimization for distribution feeder reconfiguration considering distributed generators," *Applied Mathematics and Computation*, vol. 201, no. 1-2, pp. 575–586, 2008.
- [4] T. Niknam, B. B. Firouzi, and A. Ostadi, "A new fuzzy adaptive particle swarm optimization for daily Volt/Var control in distribution networks considering distributed generators," *Applied Energy*, vol. 87, no. 6, pp. 1919–1928, 2010.
- [5] W. Freitas, J. C. M. Vieira, A. Morelato, and W. Xu, "Influence of excitation system control modes on the allowable penetration level of distributed synchronous generators," *IEEE Transactions on Energy Conversion*, vol. 20, no. 2, pp. 474–480, 2005.
- [6] Y. Hu, T. S. Du, J. H. Wu, D. Y. Li, and W. W. Li, "Solving high dimensional and complex non-convex programming based on improved quantum artificial fish algorithm," *Open Automation and Control Systems*, vol. 6, pp. 129–127, 2014.
- [7] A. A. Abou El-Ela, S. M. Allam, and M. M. Shatla, "Maximal optimal benefits of distributed generation using genetic algorithms," *Electric Power Systems Research*, vol. 80, no. 7, pp. 869–877, 2010.
- [8] R. K. Singh and S. K. Goswami, "Optimum siting and sizing of distributed generations in radial and networked systems," *Electric Power Components and Systems*, vol. 37, no. 2, pp. 127–145, 2009.
- [9] T. Chen and R. Xiao, "Modeling design iteration in product design and development and its solution by a novel artificial bee colony algorithm," *Computational Intelligence and Neuroscience*, vol. 2014, Article ID 240828, 13 pages, 2014.
- [10] B. Sookananta, W. Kuanprab, and S. Hanak, "Determination of the optimal location and sizing of Distributed Generation using Particle Swarm Optimization," in *Proceedings of the International Conference on Electrical Engineering/Electronics Computer Telecommunications and Information Technology (ECTI-CON '10)*, pp. 818–822, Chaing Mai, Thailand, May 2010.
- [11] T.-S. Du, P.-S. Fei, and Y.-J. Shen, "A modified niche genetic algorithm based on evolution gradient and its simulation analysis," in *Proceedings of the 3rd IEEE International Conference on Natural Computation (ICNC '07)*, vol. 4, pp. 35–39, IEEE, Haikou, China, August 2007.
- [12] H.-C. Tsai and Y.-H. Lin, "Modification of the fish swarm algorithm with particle swarm optimization formulation and communication behavior," *Applied Soft Computing Journal*, vol. 11, no. 8, pp. 5367–5374, 2011.
- [13] A. M. Rocha, T. F. Martins, and E. M. Fernandes, "An augmented Lagrangian fish swarm based method for global optimization," *Journal of Computational and Applied Mathematics*, vol. 235, no. 16, pp. 4611–4620, 2011.
- [14] X. L. Li, Z. J. Shao, and J. X. Qian, "An optimization model based on animal commune: fish swarm algorithm," *System Engineering Theory and Practice*, vol. 22, pp. 32–38, 2002.
- [15] L. Gyongyosi and S. Imre, "Algorithmic superactivation of asymptotic quantum capacity of zero-capacity quantum channels," *Information Sciences*, vol. 222, pp. 737–753, 2013.
- [16] J. Sun, W. Chen, W. Fang, X. Wun, and W. Xu, "Gene expression data analysis with the clustering method based on an improved quantum-behaved Particle Swarm Optimization," *Engineering Applications of Artificial Intelligence*, vol. 25, no. 2, pp. 376–391, 2012.
- [17] J. X. V. Neto, D. L. A. Bernert, and L. S. Coelho, "Improved quantum-inspired evolutionary algorithm with diversity information applied to economic dispatch problem with prohibited operating zones," *Energy Conversion and Management*, vol. 52, pp. 8–14, 2011.
- [18] E. Rieffel and W. Polak, "An introduction to quantum computing for non-physicists," *ACM Computing Surveys*, vol. 32, no. 3, pp. 300–335, 2000.
- [19] B. Qiao and H. E. Ruda, "Nonlinear Liouville equation, project Green function and stabilizing quantum computing," *Physica A: Statistical Mechanics and Its Applications*, vol. 333, pp. 197–224, 2004.
- [20] L. G. Wang, Y. Hong, and Q. H. Shi, "The global edition artificial fish swarm algorithm," *Journal of System Simulation*, vol. 21, pp. 7483–7502, 2009.
- [21] P. C. Li and S. Y. Li, "Learning algorithm and application of quantum BP neural networks based on universal quantum gates," *Journal of Systems Engineering and Electronics*, vol. 19, no. 1, pp. 167–174, 2008.
- [22] D. Bhowmik and S. Bandyopadhyay, "Gate control of the spin-splitting energy in a quantum dot: application in single qubit rotation," *Physica E: Low-Dimensional Systems and Nanostructures*, vol. 41, no. 4, pp. 587–592, 2009.

- [23] T. Radtke and S. Fritzsche, "Simulation of n -qubit quantum systems. I. Quantum registers and quantum gates," *Computer Physics Communications*, vol. 173, no. 1-2, pp. 91–113, 2005.
- [24] M. J. Qi and P. Y. Wang, "Quantum artificial fish algorithm," *Anhui Agricultural Sciences, China*, vol. 40, pp. 4469–4472, 2012.
- [25] Y. Tan and G. Z. Tan, "A chaotic local search strategy of differential evolution algorithm for global optimization," *Computer Engineering and Its Application*, vol. 45, no. 14, pp. 15–17, 2009.
- [26] E. B. Mesut and F. W. Felix, "Network reconfiguration in distribution systems for loss reduction and load balancing," *IEEE Transactions on Power Delivery*, vol. 4, no. 2, pp. 1401–1407, 1989.
- [27] D. Ye, Z. He, and T. Zang, "Siting and sizing of distributed generation planning based on adaptive mutation particle swarm optimization algorithm," *Power System Technology*, vol. 35, no. 6, pp. 155–160, 2011.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

