Research Article

The Improvement of Quantum Genetic Algorithm and Its Application on Function Optimization

Huaixiao Wang, Jianyong Liu, Jun Zhi, and Chengqun Fu

1 College of Field Engineering, PLA University of Science and Technology, Nanjing 210007, China
2 Zhenjiang Watercraft College, Zhenjiang 212003, China

Correspondence should be addressed to Huaixiao Wang; 753048237@qq.com

Received 22 January 2013; Revised 29 March 2013; Accepted 29 March 2013

1. Introduction

Quantum genetic algorithm (QGA) is the product of the combination of quantum computation and genetic algorithms, and it is a new evolutionary algorithm of probability [1]. In 1996, quantum genetic algorithm is first proposed by Naryana and Moore, and it is successfully used to solve the TSP problem [2]. QGA is essentially a kind of genetic algorithm and can be applied in the field that the conventional genetic algorithm can be applied. The efficiency of QGA is significantly better than the conventional genetic algorithm. The QGA is with a small value of the population, a fast speed of convergence, a great capability of global optimization, and a good robustness. The quantum state vector is introduced in the Genetic Algorithm to express genetic code, and quantum logic gates are used to realize the chromosome evolution. By these means, better results are achieved. However, there are still some problems in conventional QGA [3, 4]. For example, determining the direction of the rotating angle relies on a lookup table. It leads to a program with multiple judgment conditions. Moreover, the fixed rotating angle has a negative impact on the quickly searching and the lately convergence [5]. Aiming to resolve the above problem, the QGA is improved with an adaptive evolution process in this paper. Quantum mutation and quantum disaster operations are also introduced to make the algorithm with a better performance.

Quantum computation is contrary to classical computation. It uses the superposition, coherence, and the entanglement of different qubits of quantum state to realize quantum computation [6]. Quantum computation is the product of quantum mechanics applying in the field of algorithm. The ability of parallelism is the essential difference between quantum computation and classical computation. In the probability calculation, the system is not in an invariable state. Conversely, it has a certain probability, and the state probability vector is corresponding to different possible states. Quantum computation is similar to it, the probability amplitudes of quantum states is used in quantum computation, and probability amplitudes of quantum states are squared normalized, so the speed of calculation of quantum computation is \( \sqrt{N} \) times faster than the speed of classic calculation. Quantum transformation is realized by quantum rotating gates. There are some special properties of quantum computation relative to classical computation [7]. Some mechanisms of these properties can be introduced to the optimization algorithms for improving traditional optimization algorithm.

(1) Superposition of quantum computation: the basic unit for information storage is qubit in the quantum computer [8]. An \( n \)-bits quantum register can be in the state \( |\phi\rangle \) which is
the coherent superposition state of $2^n$ ground states. An act on
an $n$-bits quantum register is equal to the act on $2^n$ numbers.
The relationship between superposition state and ground states is

$$|\phi\rangle = \sum_i c_i |\phi_i\rangle,$$

(1)

where $c_i$ represents the probability amplitude of the state $|\phi_i\rangle$; $|c_i|^2$ represents the probability to collapse to the ground state $|\phi_i\rangle$ when the state $|\phi\rangle$ is measured; $c_i$ should satisfy the condition [9]:

$$\sum_i |c_i|^2 = 1.$$  

(2)

(2) Coherence of quantum computation: different from classical computation, coherence is another important property of quantum computation. The relative phase of the respective ground states changes with the interference that occurs to each ground state by the action of quantum rotating gates. For example, a quantum system of a single qubit is $|\phi\rangle = c_1 |0\rangle + c_2 |1\rangle$ and $|c_1|^2 + |c_2|^2 = 1$. After the role of some quantum gate, the values of $c_1$ and $c_2$ change correspondingly, and the new $c_1$ and $c_2$ still satisfy $|c_1|^2 + |c_2|^2 = 1$. So the probability amplitude of the ground state $|1\rangle$ will be reduced if the probability amplitude of the ground state $|0\rangle$ increases. When the quantum system is measured, coherence will disappear and collapse to some ground state, while the probability is determined by $|c_i|^2$.

(3) Entanglement of the quantum state: the quantum state which cannot be broken down into the form of the direct product of two subsystems is called entanglement state [10]. Both the operated state of the qubit and the state of the other qubit entangled are changed once operating on one or more specific qubits.

(4) Parallelism of the quantum computation: parallelism of quantum computation is completed in the same quantum circuit rather than being implemented by multiple hardwares calculated simultaneously. Parallelism of quantum computation makes use of the superimposed ability of different states which the quantum computer lies in; therefore, using a single $f(x)$ circuit can calculate multiple values of $x$.

2. Conventional Quantum Genetic Algorithm

Quantum genetic algorithm is proposed based on the concept of quantum bits and quantum superposition state [11]. In a quantum computer, a double quantum system called qubit acts as the physical media to store the information units. Qubit is the smallest unit of information in quantum computers. A qubit can be in any state $|0\rangle$, $|1\rangle$, and superposition state between $|0\rangle$ and $|1\rangle$ [12]. A state of a qubit is described as

$$|\phi\rangle = \alpha |0\rangle + \beta |1\rangle,$$

(3)

where $\alpha$ and $\beta$ are the complex numbers which are called the probability amplitude of corresponding state of qubit and satisfy the condition $|\alpha|^2 + |\beta|^2 = 1$. The states $|0\rangle$ and $|1\rangle$, respectively, denote the spin-down state and spin-up state [13]. A qubit can contain the information of both the state $|0\rangle$ and the state $|1\rangle$.

2.1. Qubit Encoding. Binary code is used to encode qubit on the polymorphic problem. A qubit can be defined by its probability amplitude as $|\alpha\beta|$, and the multiqubits encoding which has $m$ parameters is defined as follows:

$$q_j = \left[ \begin{array}{cccc} \alpha'_{11} & \alpha'_{12} & \cdots & \alpha'_{1k} \\ \beta'_{11} & \beta'_{12} & \cdots & \beta'_{1k} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha'_{m1} & \alpha'_{m2} & \cdots & \alpha'_{mk} \\ \beta'_{m1} & \beta'_{m2} & \cdots & \beta'_{mk} \end{array} \right],$$

(4)

where $q_j$ represents the $j$th individual chromosome of the $t$th generation; $k$ represents the number of qubit encoding of each gene; $m$ represents the number of genes in the chromosome [14].

Initialize the quantum encoding $(\alpha, \beta)$ of each individual in the population with $(1/\sqrt{2}, 1/\sqrt{2})$, which indicates that when $t = 0$, the possibility of each state expressed by a chromosomal is equal [15].

2.2. Quantum Rotating Gates. Compared with the conventional genetic algorithm, quantum genetic algorithm applies the probability amplitude of qubits to encode chromosome and uses quantum rotating gates to realize chromosomal updated operation. Since the chromosomes are in superposition state or entanglement state, the generation of offspring is not determined by the parent group when the Quantum rotating gates is used to realize the genetic operation. It is jointly determined by the optimal individual of the parent group and probability amplitude of each state. In other words, the genetic manipulation of quantum genetic algorithm is mainly through acting on the superposition state or entanglement state by the Quantum rotating gates to change the probability amplitude. Therefore, the construction of Quantum rotating gates is the key issue of quantum genetic algorithm [16], and it directly affects the performance of the algorithm.

Quantum rotating gates can be designed according to the practical problems and usually can be defined as

$$U(\theta) = \left[ \begin{array}{cc} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{array} \right].$$

(5)

The updated process is

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

(6)
Table 1: Adjustment strategy of rotating angle.

<table>
<thead>
<tr>
<th>$x_i$</th>
<th>$best_i$</th>
<th>$f(x) &gt; f(best)$</th>
<th>$\Delta \theta_i$</th>
<th>$s(\alpha_i, \beta_i)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>FALSE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>TRUE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>FALSE</td>
<td>$\Delta \theta_i$</td>
<td>+1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>TRUE</td>
<td>$\Delta \theta_i$</td>
<td>-1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>FALSE</td>
<td>$\Delta \theta_i$</td>
<td>-1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>TRUE</td>
<td>$\Delta \theta_i$</td>
<td>+1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>FALSE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>TRUE</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

where $(\alpha_i, \beta_i)^T$ and $(\alpha'_i, \beta'_i)^T$ are the probability amplitudes of the $i$th qubit in chromosome before and after the quantum rotating gates updating, respectively; $\theta_i$ is the rotating angle; the value and the sign of $\theta_i$ are determined by the adjustment strategy [17], which are shown in Table 1. Moreover, $x_i$ is the $i$th bit of the current chromosome; $best_i$ is the $i$th bit of the current optimal chromosome; $f(x)$ is the fitness function; $s(\alpha_i, \beta_i)$ is the direction of the rotating angle; $\Delta \theta_i$ is the value of the rotating angle. The value of $\Delta \theta_i$ is determined by certain adjustment strategy, in conventional quantum genetic algorithm and the value of $\Delta \theta_i$ is generally a constant value is around $0.01\pi$. The adjustment strategy is, comparing the fitness of the currently measured value $f(x)$ of the individual $q_i^f$ with the fitness of the current optimal individual $f(best_i)$, if $f(x) > f(best)$, then adjust the corresponding qubits of $q_i^f$, making the probability amplitude $(\alpha_i, \beta_i)$ evolves toward the direction that is propitious to the emergence of $x_i$. Conversely, if $f(x) < f(best)$, then adjust the corresponding qubits of $q_i^f$, making the probability amplitude $(\alpha_i, \beta_i)$ evolves toward the direction that is propitious to the emergence of best.

2.3. Analysis on the Characteristic of Quantum Genetic Algorithm. Quantum genetic algorithm is a more wonderful optimization process than the conventional genetic algorithm, and its encoding mode is more complex, and each generation of the evolution can cover a wider area [18]. Quantum genetic algorithm combines some characteristics of quantum computation with the genetic algorithm whose individuals are chosen by the natural selection. The fine species better adapt to environment of quantum genetic algorithm are produced by the operation of quantum rotating gates with the goal of the best individual of current generation [19]. The main features of quantum genetic algorithm are as follows.

(1) QGA is self-organizing, self-adaptive, and self-learning [20]. (2) The object handling by QGA is not the parameter, but the chromosome strings encoded according to parameter variable [21]. The encoding operation enables the quantum genetic algorithm to operate directly on the structure of the object. (3) QGA searches a group of points rather than a single point in the solution space at the same time. QGA samples in different areas of space simultaneously [22]. (4) In QGA, the knowledge of the searching space or other ancillary information is not needed. Only the fitness function is used to evaluate the individuals [23].

The flowchart of conventional quantum genetic algorithm is shown in Figure 1.

3. Improvements of Quantum Genetic Algorithm

There are some shortages in conventional quantum genetic algorithm [24]. (1) The premature convergence problem: conventional quantum genetic algorithm simply uses fitness to evaluate the solution, causing the genes of some individuals whose fitness is large to spread rapidly in the population, making the population loses diversity prematurely. The solution often falls into the local optimal solution; (2) local search ability: conventional quantum genetic algorithm is good at global search performance, but its local search performance is inadequate, resulting in slower convergence in the later
stage of evolution and cannot converge to the global optimal solution.

3.1. Determination on Rotating Direction. In the optimization process, the value and direction of the quantum rotating gates are determined firstly. The current method to determine the direction of the rotating angle is based on a lookup table. As the lookup table involves judgment of multiple conditions, the efficiency of the algorithm is affected [25]. Therefore, a new method is introduced to simplify the determination of the direction of the rotating angle in this paper.

Theorem 1. Construct a determinant $A = \begin{bmatrix} \alpha_a & \alpha_i \\ \beta_a & \beta_i \end{bmatrix}$, where, $(\alpha_a, \beta_a)$ is the probability amplitude of some qubit of the optimal solution which has been searched. $(\alpha_i, \beta_i)$ is the probability amplitude of corresponding state of qubit in the current solution. When $x_i \neq \text{best}$, if $A \neq 0$, then the direction of the rotating angle is $\text{sgn}(A)$. If $A = 0$, the direction of the rotating angle can be either positive or negative.

Proof. The angles of the qubits $[\alpha_a, \beta_a]^T$ and $[\alpha_i, \beta_i]^T$ in the unit circle are separately $\theta_a$ and $\theta_i$; then,

$$A = \begin{bmatrix} \cos \theta_a & \cos \theta_i \\ \sin \theta_a & \sin \theta_i \end{bmatrix} = \sin (\theta_i - \theta_a). \quad (7)$$

If $A \neq 0$, when $0 < |\theta_i - \theta_a| < \pi$, $\text{sgn}(\Delta \theta) = -\text{sgn}(\theta_i - \theta_a) = -\text{sgn}(\sin(\theta_i - \theta_a)) = -\text{sgn} A$ and when $\pi < |\theta_i - \theta_a| < 2\pi$, $\text{sgn}(\Delta \theta) = \text{sgn}(\theta_i - \theta_a) = -\text{sgn}(\sin(\theta_i - \theta_a)) = -\text{sgn} A$. If $A = 0$, then $|\theta_i - \theta_a| = 0$ or $\pi$. Positive and negative rotating effect is the same at this time, so the direction of the rotating angle $\theta$ can be either positive or negative (proof is over).

3.2. Self-Adaptive Rotating Angle Strategy. The conventional quantum genetic algorithm adopts a rotating strategy with fixed angle. In this paper, a rotating strategy with self-adaptive angle is adopted. It can dynamically adjust the value of the rotating angle of the Quantum rotating gates based on the evolutionary process [26]. A large rotating angle is set early in the evolutionary process to quickly facilitate the entire interval and find the region with the optimal values. To accurately find the optimal value, we reduce the value of rotating angle, while the evolution increases. Measure the individual $q^j_i$ and compare its fitness $f(x_i)$ with the fitness $f(\text{best})$ of the best individual. Adjust the corresponding qubit in $q^j_i$ based on the result of the above comparison and make $(\alpha, \beta)$ evolve to the direction of the optimal solution.

The self-adaptive quantum genetic algorithm (AQGA) is proposed in this paper. AQGA adjusts the value of the rotating angle according to the situation of evolution of the population at any time.

Adjust the value of the rotating angle according to the number of generation; the regulator is as the following formula:

$$\theta_i = \theta_{max} - \left( \frac{\theta_{max} - \theta_{min}}{\text{iter}} \right) \times \text{iter}, \quad (8)$$

3.3. Add Quantum Mutation Operation. Add quantum mutation operation in conventional quantum genetic algorithm. Quantum mutation enables some individuals to deviate slightly from the current evolutionary direction and prevent the evolution of individual into a local optimal solution [27]. Quantum mutation can completely reverse the individual’s evolutionary direction by swapping the value of probability amplitude of qubits $(\alpha, \beta)$.

Quantum mutation is an assistant operation with the purpose to enhance the local searching ability of quantum genetic algorithm and avoid the loss of important information in the population. Quantum NOT gates is adopted to realize chromosomal variation. Quantum NOT gate is applied to transform the selected number of qubits according to mutation probability randomly to transform corresponding probability amplitude of the qubits. Quantum mutation operation helps to increase the diversity of the population and reduce the probability of premature convergence.

Pseudocode program of quantum mutation process is described as in Algorithm 1.

3.4. Add Quantum Disaster Operation. Add quantum disaster operation in the conventional quantum genetic algorithm. The algorithm may fall into local optimal solution, while the algorithm has performed several generations and the best individual is in a stable state. Thus, we need to take the quantum disaster operation to get out of the local optimal solution [28]. The method is to apply a large disturbance to some individuals in the population and regenerate some other new random individuals.

Quantum disaster process pseudocode program is described as in Algorithm 2.

It needs to be noted that the intercross operation is not added into the improved quantum genetic algorithm. The crossover operator is able to change the linear superposition...
Begin
If (disaster-condition)
Begin
If (The chromosome is not the best chromosome)
Initialize the chromosome;
End
End

Algorithm 2

Start
Initialize the population $Q(t_0)$ and every parameter
Test every individual of $Q(t_0)$
evaluate the fitness of every individual of $Q(t_0)$
Set the best individual as the evolutional goal of next generation
End

Yes
Meet final condition
Test every individual of $Q(t)$
Test every individual of $Q(t)$
Quantum rotating door update population, get $Q(t + 1)$
Quantum mutation
Memorize the best individual and its fitness
Whether or not disaster
Yes
Colony disaster
$t = t + 1$

No
Test every individual of $Q(t)$

Figure 2: Flowchart of improved quantum genetic algorithm.

4. Simulation Test

The simulation test in this paper is based on the AMD CPU 1.8 GHz, 1.5 GB RAM; the algorithm is written and compiled in MATLAB.

4.1. Problem Description. Find the optimal solution in the complex binary function:

$$\max f(x, y) = x \sin(4\pi x) + y \sin(20\pi y),$$

$$-3.0 \leq x_1 \leq 12.1,$$

$$4.1 \leq x_2 \leq 5.9.$$  \hfill (9)

The figure of the function is shown in Figure 3. The parameters are set as follows. Both the conventional quantum genetic algorithm and improved quantum genetic algorithm are encoded by the binary; the evolution generation is 200; the size of population is 40; the length of each binary variable is 20; fitness function is the objective function. Take 10 times simulation test using conventional quantum genetic algorithm and its improved method, respectively.

4.2. The Results of Conventional Quantum Genetic Algorithm. The results of the conventional quantum genetic algorithm are shown in Table 2. Make a census of the data in Table 2, the statistical results are shown in Table 3. The result of 200 generations’ evolutionary process of a certain time of conventional quantum genetic algorithm is shown in Figure 4. The conventional quantum genetic algorithm obtains the optimal solution around 130 generation. Evolutionary process is slower between 50 and 110 generations. The $x$-axis represents the evolutionary generations; $y$-axis represents the best fitness of every generation. (The meaning of the $x$-axis and the $y$-axis of the following figure is the same as Figure 4).

4.3. Result of Self-Adaptive Quantum Genetic Algorithm. The rotating angle adaptively changes with the evolution generation based on the conventional quantum genetic algorithm. Four cases are tested in this paper; the maximum rotating angle and the minimum rotating angle of the four cases are, respectively, $(0.008\pi, 0.1\pi)$, $(0.01\pi, 0.1\pi)$, $(0.01\pi, 0.05\pi)$, and $(0.005\pi, 0.05\pi)$. Take 10 rounds of the experiments for the four cases; the experimental results were shown in Tables 4, 5, 6, 7, and 8, respectively.

Make a census of the experimental data in Tables 4, 5, 6, and 7, the statistical results are shown in Table 8. According to the data in Table 8, the rotating strategy is selected $(0.005\pi, 0.05\pi)$ in this paper.
6 Mathematical Problems in Engineering

Table 2: The results of the conventional quantum genetic algorithm.

<table>
<thead>
<tr>
<th>Index</th>
<th>Times</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>(x_2)</td>
<td></td>
<td>5.72504</td>
<td>5.32505</td>
<td>5.72509</td>
<td>5.72505</td>
<td>5.72529</td>
<td>5.72504</td>
<td>5.72529</td>
<td>5.72496</td>
<td>5.72505</td>
<td>5.72504</td>
</tr>
<tr>
<td>Optimal value</td>
<td></td>
<td>17.3503</td>
<td>16.9503</td>
<td>17.3441</td>
<td>17.3503</td>
<td>17.3496</td>
<td>17.3503</td>
<td>17.3496</td>
<td>17.3502</td>
<td>17.3503</td>
<td>17.3441</td>
</tr>
<tr>
<td>Evolution generations</td>
<td></td>
<td>130</td>
<td>110</td>
<td>80</td>
<td>100</td>
<td>80</td>
<td>90</td>
<td>110</td>
<td>180</td>
<td>110</td>
<td>60</td>
</tr>
</tbody>
</table>

Table 3: The statistical results of conventional quantum genetic algorithm.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>The optimal result</th>
<th>The worst result</th>
<th>The biggest evolution generation</th>
<th>The smallest evolution generation</th>
<th>The average evolution generation</th>
<th>Times of convergence</th>
</tr>
</thead>
<tbody>
<tr>
<td>CQGA</td>
<td>(11.6255, 5.72504, 17.3503)</td>
<td>(11.6254, 5.32505, 16.9503)</td>
<td>180</td>
<td>60</td>
<td>105</td>
<td>5</td>
</tr>
</tbody>
</table>

Table 4: The result of self-adaptive quantum genetic algorithm with the rotating strategy selected \((0.008\pi, 0.1\pi)\).

<table>
<thead>
<tr>
<th>Index</th>
<th>Times</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>(x_2)</td>
<td></td>
<td>5.72505</td>
<td>5.72504</td>
<td>5.42813</td>
<td>5.72363</td>
<td>5.72504</td>
<td>5.42813</td>
<td>5.3252</td>
<td>5.72503</td>
<td>5.72504</td>
<td>5.72503</td>
</tr>
<tr>
<td>Evolution generations</td>
<td></td>
<td>20</td>
<td>20</td>
<td>10</td>
<td>70</td>
<td>20</td>
<td>30</td>
<td>20</td>
<td>40</td>
<td>25</td>
<td>10</td>
</tr>
</tbody>
</table>

Figure 4: Evolutionary process of conventional quantum genetic algorithm.

Figure 5: Evolutionary process of self-adaptive quantum genetic algorithm.

4.4. The Result of Self-Adaptive Quantum Genetic Algorithm in which Mutation Operation Is Added. The mutation operation is added based on the self-adaptive quantum genetic algorithm, and the probability of mutation is 0.001. Table 9 shows the statistical results of the experiment by performing the algorithm 10 times. We can get the conclusion that the speed of evolution is improved again.

One certain time of the evolution process is shown in Figure 6. Through the evolutionary process in Figure 6, we can get that the optimal solution is obtained around the 30th
Table 5: The result of self-adaptive quantum genetic algorithm with the rotating strategy selected (0.01π, 0.1π).

<table>
<thead>
<tr>
<th>Index</th>
<th>Times</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td>$x_2$</td>
<td>5.72504</td>
</tr>
<tr>
<td>Evolution generations</td>
<td>20</td>
</tr>
</tbody>
</table>

Table 6: The result of self-adaptive quantum genetic algorithm with the rotating strategy selected (0.01π, 0.05π).

<table>
<thead>
<tr>
<th>Index</th>
<th>Times</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td>$x_2$</td>
<td>5.72509</td>
</tr>
<tr>
<td>Optimal value</td>
<td>15.8503</td>
</tr>
<tr>
<td>Evolution generations</td>
<td>30</td>
</tr>
</tbody>
</table>

Figure 6: Evolutionary process of self-adaptive quantum genetic algorithm in which mutation operation is added.

Figure 7: Evolutionary process of self-adaptive mutation quantum genetic algorithm added disaster operation.

4.5. The Result of Self-Adaptive Mutation Quantum Genetic Algorithm Added Disaster Operation. The disaster operation is added based on the self-adaptive mutation quantum genetic algorithm. Table 10 shows the statistical results of the experiment by performing the algorithm 10 times. We can get the conclusion that the times of convergence of self-adaptive mutation quantum genetic algorithm adding disaster operation have been improved, and the biggest evolution generation is largely more than the algorithm without disaster operation; the reason is that usually the disaster operation happens late in the evolution. The disaster operation makes the times of convergence improved, but the biggest evolution generation unavoidably becomes more.

An evolutionary process is shown in Figure 7.

4.6. Analysis on Evolutionary Process of Self-Adaptive Quantum Genetic Algorithm Added Intercross Operation. As mentioned previously, the improved quantum genetic algorithm cannot add the intercross operation [30]. Adding the intercross operation will decline the efficiency of the algorithm. The evolution process of self-adaptive quantum genetic algorithm with the adding intercross operation is shown in Figures 8 and 9. Compare Figure 5 with Figure 8, the optimal
Table 7: The result of self-adaptive quantum genetic algorithm with the rotating strategy selected $(0.005\pi, 0.05\pi)$.

<table>
<thead>
<tr>
<th>Index</th>
<th>Times</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_1$</td>
<td>11.6255</td>
</tr>
<tr>
<td>$x_2$</td>
<td>5.32504</td>
</tr>
<tr>
<td>Optimal value</td>
<td>16.9503</td>
</tr>
<tr>
<td>Evolution generations</td>
<td>40</td>
</tr>
</tbody>
</table>

Table 8: The statistical results of self-adaptive quantum genetic algorithm.

<table>
<thead>
<tr>
<th>Rotating strategy</th>
<th>The optimal result</th>
<th>The worst result</th>
<th>The biggest evolution generation</th>
<th>The smallest evolution generation</th>
<th>The average evolution generation</th>
<th>Times of convergence</th>
</tr>
</thead>
<tbody>
<tr>
<td>$(0.008\pi, 0.1\pi)$</td>
<td>$(11.6255, 5.72503, 17.3503)$</td>
<td>$(10.1024, 5.72504, 15.8483)$</td>
<td>70</td>
<td>10</td>
<td>26.5</td>
<td>3</td>
</tr>
<tr>
<td>$(0.01\pi, 0.1\pi)$</td>
<td>$(11.6255, 5.72504, 17.3503)$</td>
<td>$(10.1256, 5.32505, 15.4503)$</td>
<td>70</td>
<td>10</td>
<td>28.3</td>
<td>2</td>
</tr>
<tr>
<td>$(0.01\pi, 0.05\pi)$</td>
<td>$(11.6255, 5.72504, 17.3503)$</td>
<td>$(10.1256, 5.72509, 15.8503)$</td>
<td>40</td>
<td>20</td>
<td>29</td>
<td>4</td>
</tr>
<tr>
<td>$(0.005\pi, 0.05\pi)$</td>
<td>$(11.6255, 5.72502, 17.3503)$</td>
<td>$(11.6255, 4.92502, 16.5503)$</td>
<td>70</td>
<td>20</td>
<td>42.5</td>
<td>5</td>
</tr>
</tbody>
</table>

Figure 8: Evolutionary process of self-adaptive quantum genetic algorithm adding the intercross operation.

Figure 9: Evolutionary process of self-adaptive quantum genetic algorithm adding the intercross and mutation operation.

4.7 Comparison of Time between the Two Methods to Determine Rotating Angle Direction. Take 8 times of experiments separately using the conventional method and the method of this paper to determine the rotating angle direction; get rid of the largest and smallest group, respectively, and the time of the rest 6 groups is shown in Table II.

From Table II, it can be concluded that the average running time of the improved method to determine the rotating angle direction is significantly shorter than the time of conventional method. The efficiency of algorithm is improved.
5. Conclusions

In this paper, six rounds of experiments were presented in this paper. We performed the experiments in each round for 10 tests. The experiments show that the conventional quantum genetic algorithm has been improved by adding many improvements. The simulation test shows that to improve the evolution efficiency of quantum genetic algorithm, some proper operations can be added in the evolutionary process; however, not all improvement strategies which can be applied to traditional genetic algorithms can be used to improve the conventional quantum genetic algorithm; it is because of the specificity of the conventional quantum genetic algorithm. The improvements made in this paper improve the algorithm structure, and efficiency of evolution of the conventional quantum genetic algorithm is also improved.

References


