

Research Article

Improved Quantum Genetic Algorithm in Application of Scheduling Engineering Personnel

Huaxiao Wang, Ling Li, Jianyong Liu, Yong Wang, and Chengqun Fu

College of Field Engineering, PLA University of Science and Technology, Nanjing 210007, China

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

Received 22 January 2014; Revised 5 June 2014; Accepted 13 July 2014; Published 23 July 2014

Academic Editor: Mohammad T. Darvishi

Copyright © 2014 Huaxiao Wang 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.

To verify the availability of the improved quantum genetic algorithm in solving the scheduling engineering personnel problem, the following work has been carried out: the characteristics of the scheduling engineering personnel problem are analyzed, the quantum encoding method is proposed, and an improved quantum genetic algorithm is applied to address the issue. Taking the low efficiency and the bad performance of the conventional quantum genetic algorithm into account, a universal improved quantum genetic algorithm is introduced to solve the scheduling engineering personnel problem. Finally, the examples are applied to verify the effectiveness and superiority of the improved quantum genetic algorithm and the rationality of the encoding method.

1. Introduction

In engineering work, the number of participants is large and the qualities of the personnel are uneven, so the time of each group carrying out an engineering work is different; the processes of engineering work are so many that the scheduling of the engineering personnel is complex [1]. There is parallelism in some processes of engineering work. Any work group can be selected for any process of engineering work and the operating times of different work groups are different. These factors increase the complexity and flexibility of scheduling. Different scheduling schemes cause different total operation time. Proper scheduling scheme can improve efficiency and save time [2]. So, solving the personnel scheduling problem of engineering work is one of the key aspects to improve the efficiency of engineering work.

Scheduling engineering personnel is a complex NP-hard problem [3]. Currently the main methods to solve such problems contain branch-bound method, heuristic algorithms, and so on, but these algorithms can only solve the scheduling engineering personnel problem with a small scale, and there are imperfections in coding method of these algorithms, which may lead to illegal solutions [4]. The improved quantum genetic algorithm has better robustness,

versatility, excellent computing power, implicit parallelism, and global search capability and can be successful in solving the scheduling engineering operating personnel problem.

2. Description of Scheduling Engineering Operating Personnel Problem

Scheduling engineering operating personnel problem is the extension of standard scheduling personnel problem.

2.1. Standard Scheduling Personnel Problem. The standard personnel scheduling problem can be described as follows: suppose there are m tasks which need to be finished, all of which can be divided into two stages: A and B . The two stages must be finished in order, as in Figure 1. Every stage of the task is finished by only one work group. The object is to find an order to make the time from the beginning of the first task to the end of the last task shortest.

The rule of sorting of the two-stage engineering work has been put forward by Johnson in 1954, which is as follows:

$$\min(a_i, b_j) \leq \min(a_j, b_i). \quad (1)$$

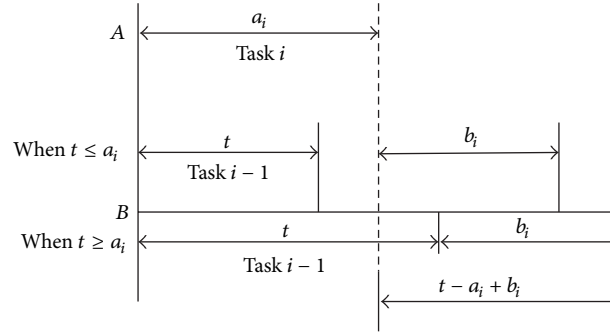


FIGURE 1: Diagram of standard scheduling personnel problem.

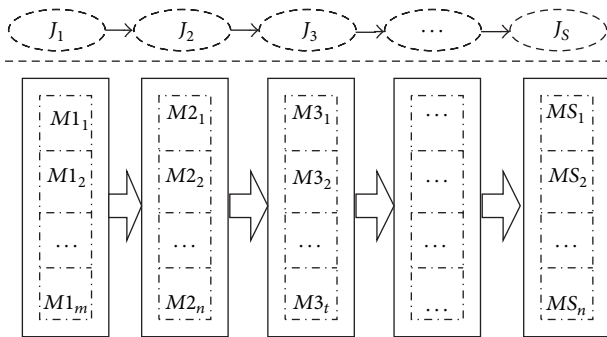


FIGURE 2: Distribution of engineering personnel.

Equation (1) is the condition that task i is ordered before the task j , where a_i is the time needed by the first stage of task i and b_j is the time needed by the second stage of task j .

2.2. Extension of Scheduling Engineering Personnel Problem. Extension of scheduling engineering personnel problem is based on the standard personnel scheduling problem. The stages of the task needed to be done are more than two, the work groups for finishing every stage of the task are more than one, and the time in which different work teams finish the task is different. There is not an authentic rule of sorting of the extension of scheduling engineering personnel problem.

Extension of scheduling engineering operating personnel problem can be described as follows. There are a number of engineering works which can be decomposed into a certain number of steps, and at least one of all steps can be completed in parallel by multiple work groups. The problem is to determine the distribution of work groups in parallel. That is, there are n work groups and N engineering works $\{W_1, W_2, W_3, \dots, W_N\}$. Decompose these engineering works into S processes uniformly $\{J_1, J_2, J_3, \dots, J_S\}$. Each process J_i has Z_i parallel work group ($1 \leq i \leq S, 1 \leq Z_i \leq n, \sum Z_i > S$). The figure of scheduling engineering operating personnel problem is shown in Figure 2. The target is to minimize the maximum flow time through conducting an optimal allocation [5].

Each process of each engineering work can be completed by any parallel team of corresponding process, but the work

times are different [6]. The specific example is shown in Table 1. J_i stands for the i th process, M_{ij} stands for the j th parallel work group of the i th process, and the data of M_{ij} stands for the work time that the j th parallel work group needs to complete the i th process. Table 1 is an operational timetable which includes four engineering works (W_1, W_2, W_3, W_4), three processes (J_1, J_2, J_3), and nine engineering work groups.

The engineering personnel scheduling problem in this paper satisfies the following constraints: (1) different engineering works have the same priority; (2) there is no order constraint between different engineering works, but the processes of the same engineering work are successive [7]; (3) any engineering work can start at the beginning; (4) one work group only can carry out one engineering work at the same time; (5) the same process of the same engineering work can only be completed by one work group. (6) Once started, each process of the engineering work cannot be interrupted. (7) The only performance is the maximum completion time T_{\max} ; that is, the purpose of scheduling is to minimize the maximum completion time:

$$\min T_{\max} = \min (\max T_k), \quad 1 \leq k \leq m, \quad (2)$$

where T_k is the completion time of the work group M_{sp} , $1 \leq p \leq Z_s$ (Z_s is the total number of work groups of each process), and M_{sp} is the p th parallel work group of the final process.

3. Calculation of Engineering Work Time

We need to follow certain principles to calculate the work time of the determined scheme to minimize the work time when the scheduling scheme of engineering work groups is determined. During the whole engineering work, the consuming time is divided into two parts: the working time and the waiting time [8].

Suppose the o_k th work group of the work groups of the first process $M_{(o_k)}^{(1)}$ needs cost $X_{(o_k)k}^{(2)}$ to complete the first process of the k th engineering work, the p_k th work group of the work groups of the second process $M_{(p_k)}^{(2)}$ needs cost $X_{(p_k)k}^{(2)}$ to complete the second process, and the q_k th work group of

TABLE 1: Example of scheduling engineering personnel problem.

Engineering works	Processes								
	J_1			J_2			J_3		
	$M1_1$	$M1_2$	$M1_3$	$M1_4$	$M2_1$	$M2_2$	$M3_1$	$M3_2$	$M3_3$
W_1	3	2	4	3	3	4	5	3	4
W_2	4	5	3	2	6	5	5	6	3
W_3	4	3	5	6	4	2	3	4	5
W_4	2	4	1	3	5	3	4	6	2

the work groups of the third process $M_{(q_k)}^{(3)}$ needs cost $X_{(p_k)k}^{(2)}$ to complete the third process. The total time the first process of the k th engineering work costs is $M_{(o_k)k}^{(1)}$; the total time the p_{k-1} th work group $M_{(p_{k-1})k}^{(2)}$ costs on the second process of the k th engineering work is $M_{(p_{k-1})k-1}^{(2)}$, ($k > 1$). Calculation of $M_{(p_k)k}^{(2)}$ is divided into two situations:

- (1) when $M_{(o_k)k}^{(1)} \geq M_{(p_{k-1})k-1}^{(2)}$, $M_{(p_k)k}^{(2)} = M_{(o_k)k}^{(1)} + X_{(p_k)k}^{(2)}$;
- (2) when $M_{(o_k)k}^{(1)} < M_{(p_{k-1})k-1}^{(2)}$, $M_{(p_k)k}^{(2)} = M_{(p_{k-1})k-1}^{(2)} + X_{(p_k)k}^{(2)}$.

Take the two situations together, the formula of calculating the total time $M_{(p_k)k}^{(2)}$ of the p_k th work group $M_{(p_k)k}^{(2)}$ completing the second process of the k th engineering work is

$$M_{(p_k)k}^{(2)} = \max \left(M_{(o_k)k}^{(1)}, M_{(p_{k-1})k-1}^{(2)} \right) + X_{(p_k)k}^{(2)}. \quad (3)$$

Similarly, similar formula of the third step can be obtained. The specific formulas are in Table 2. The formulas in Table 2 indicate the total time M_j^i of the k th engineering work cost from the beginning to the i th process.

4. The Improved Quantum Genetic Algorithm Used to Solve the Scheduling Engineering Personnel Problem

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 [9]. It is successfully used to solve the TSP problem [10]. 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 [11]. Genetic algorithm has been widely applied in scheduling engineering personnel [12–14]. The improved quantum genetic algorithm is applied to solve the scheduling engineering personnel problem in order to achieve the optimization of scheduling engineering personnel.

4.1. The Qubit Encoding of Scheduling Engineering Personnel Problem. Ensuring that the encoding method is the most important for applying improved quantum genetic algorithm

to solve the scheduling engineering personnel problem. The qubit encoding needs to combine the optimization problem with the improved quantum genetic algorithm. The engineering works which can be divided into the same number of processes need to be completed, and there is at least one process that can be completed by multiple parallel work groups. It needs to ensure the distribution of parallel work groups in order to minimize the maximum completion time.

The features of scheduling engineering personnel problem are as follows: for example, there are N engineering works that need to be completed; every work must be completed through S processes. The number of parallel work groups of every process is Z_i ($1 \leq i \leq S$). There is at least one process that can be completed by multiple parallel work groups; that is, there is at least one Z_j which is greater than one. Construct a $N \times S$ dimensional matrix:

$$A_{N \times S} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1S} \\ a_{21} & a_{22} & \cdots & a_{2S} \\ \vdots & \vdots & & \vdots \\ a_{1S} & a_{2S} & \cdots & a_{NS} \end{bmatrix}, \quad (4)$$

where the element a_{ij} of matrix is a random real number between the ranges $[1, Z_j + 1)$, which means that the j th process of the i th engineering work is completed by the $\text{int } a_{ij}$ th work group (int means getting the greatest integer which is not greater than a_{ij}).

If $\text{int } a_{jk_1} = \text{int } a_{jk_2}$ ($k_1 \neq k_2$), it indicates that the same processes of multiple engineering works are completed by the same work group. Use variable n_i ($i = 1, \dots, \sum Z_j$) to record the number of engineering works every work group completes.

Accord the structure of (2) to design the chromosome of quantum genetic algorithm:

$$\text{chromosome}_k = \left[N, S, \begin{pmatrix} J_1 & J_2 & \cdots & J_s \\ Z_1 & Z_2 & \cdots & Z_s \end{pmatrix}, \right. \\ \left. a_{11}, a_{12}, \dots, a_{1S}, \dots, a_{N1}, \dots, a_{NS} \right]. \quad (5)$$

TABLE 2: Total time of completion of every process.

Work	Process				
	Process 1	Process 2	Process 3	...	Process i
1	$0 + X_{(o_1)1}^{(1)}$	$M_{(o_1)1}^{(1)} + X_{(p_1)1}^{(2)}$	$M_{(p_1)1}^{(2)} + X_{(q_1)1}^{(3)}$...	$M_{(y_1)1}^{(i-1)} + X_{(z_1)1}^{(i)}$
2	$M_{(o_1)1}^{(1)} + X_{(o_2)2}^{(1)}$	$\max(M_{(o_2)2}^{(1)}, M_{(p_1)1}^{(2)}) + X_{(p_2)2}^{(2)}$	$\max(M_{(p_2)2}^{(2)}, M_{(q_1)1}^{(3)}) + X_{(q_2)2}^{(3)}$...	$\max(M_{(y_2)2}^{(i-1)}, M_{(z_1)1}^{(i)}) + X_{(z_2)2}^{(i)}$
3	$M_{(o_2)2}^{(1)} + X_{(o_3)3}^{(1)}$	$\max(M_{(o_3)3}^{(1)}, M_{(p_2)2}^{(2)}) + X_{(p_2)3}^{(2)}$	$\max(M_{(p_3)3}^{(2)}, M_{(q_2)2}^{(3)}) + X_{(q_3)3}^{(3)}$...	$\max(M_{(y_3)3}^{(i-1)}, M_{(z_2)2}^{(i)}) + X_{(z_3)3}^{(i)}$
⋮	⋮	⋮	⋮	⋮	⋮
k	$M_{(o_{k-1})k-1}^{(1)} + X_{(o_k)k}^{(1)}$	$\max(M_{(o_k)k}^{(1)}, M_{(p_{k-1})k-1}^{(2)}) + X_{(p_k)k}^{(2)}$	$\max(M_{(p_k)k}^{(2)}, M_{(q_{k-1})k-1}^{(3)}) + X_{(q_k)k}^{(3)}$...	$\max(M_{(y_k)k}^{(i-1)}, M_{(z_{k-1})k-1}^{(i)}) + X_{(z_k)k}^{(i)}$

The mathematical model is $\min = \max(M_{(p_k)k}^{(2)}, M_{(q_{k-1})k-1}^{(3)}) + X_{(q_k)k}^{(3)}$, where k is the number of engineering works.

The improved quantum genetic algorithm adopted in this paper is the algorithm based on the algorithm in document

[11]. The multiqubits encoding which has m parameters is defined as follows:

$$q_j^t = \left[\begin{array}{c|c|c|c|c|c|c|c|c|c|c|c|c|c|c|c} \alpha_{11}^t & \alpha_{12}^t & \dots & \alpha_{1k}^t & \alpha_{21}^t & \alpha_{22}^t & \dots & \alpha_{2k}^t & \alpha_{m1}^t & \alpha_{m2}^t & \dots & \alpha_{mk}^t & \beta_{11}^t & \beta_{12}^t & \dots & \beta_{1k}^t & \beta_{21}^t & \beta_{22}^t & \dots & \beta_{2k}^t & \beta_{m1}^t & \beta_{m2}^t & \dots & \beta_{mt}^t \end{array} \right], \quad (6)$$

where q_j^t 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

[15]. Initialize the quantum encoding (α, β) 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 chromosome is equal [16].

The allocation coding matrix is encoded by qubit as

$$Q = \left[\begin{array}{c|c|c|c|c|c|c|c|c|c|c|c|c|c|c|c} \alpha_{111} & \alpha_{112} & \dots & \alpha_{11Q_1} & \alpha_{121} & \alpha_{122} & \dots & \alpha_{12Q_2} & \dots & \alpha_{1S1} & \alpha_{1S2} & \dots & \alpha_{1SQ_S} & \beta_{111} & \beta_{112} & \dots & \beta_{11Q_1} & \beta_{121} & \beta_{122} & \dots & \beta_{12Q_2} & \dots & \beta_{1S1} & \beta_{1S2} & \dots & \beta_{1SQ_S} \\ \alpha_{211} & \alpha_{212} & \dots & \alpha_{21Q_1} & \alpha_{221} & \alpha_{222} & \dots & \alpha_{22Q_2} & \dots & \alpha_{2S1} & \alpha_{2S2} & \dots & \alpha_{2SQ_S} & \beta_{211} & \beta_{212} & \dots & \beta_{21Q_1} & \beta_{221} & \beta_{222} & \dots & \beta_{22Q_2} & \dots & \beta_{2S1} & \beta_{2S2} & \dots & \beta_{2SQ_S} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \alpha_{N11} & \alpha_{N12} & \dots & \alpha_{N1Q_1} & \alpha_{N21} & \alpha_{N22} & \dots & \alpha_{N2Q_2} & \dots & \alpha_{NS1} & \alpha_{NS2} & \dots & \alpha_{NSQ_S} & \beta_{N11} & \beta_{N12} & \dots & \beta_{N1Q_1} & \beta_{N21} & \beta_{N22} & \dots & \beta_{N2Q_2} & \dots & \beta_{NS1} & \beta_{NS2} & \dots & \beta_{NSQ_S} \end{array} \right]. \quad (7)$$

The element in (7) is $\begin{array}{c|c|c|c} \alpha_{ij1} & \alpha_{ij2} & \dots & \alpha_{ijQ_j} \\ \beta_{ij1} & \beta_{ij2} & \dots & \beta_{ijQ_j} \end{array}$ which represents the qubit encoding of the ordinal number of work team, where i ($i = 1, 2, \dots, N$) represents the ordinal number of engineering work, j ($j = 1, 2, \dots, S$) represents the ordinal number of process, and Q_k ($k = 1, \dots, S$) represents the length of the qubit encoding; usually the more work groups there are for the process k , the longer the length is. The number a_{ij} generated by qubit encoding which represents the j th process of the i th work is finished by the a_{ij} th work group.

4.2. The Solution Based on Improved Quantum Genetic Algorithm. The solution should be solved based on the qubit coding.

4.2.1. Quantum Rotating Gates. Quantum genetic algorithm applies the probability amplitude of qubits to encode chromosome and uses quantum rotating gates to realize chromosomal updated operation.

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

$$U(\theta_i) = \begin{bmatrix} \cos \theta_i & -\sin \theta_i \\ \sin \theta_i & \cos \theta_i \end{bmatrix}. \quad (8)$$

The updated process is

$$\begin{bmatrix} \alpha'_i \\ \beta'_i \end{bmatrix} = U(\theta_i) \begin{bmatrix} \alpha_i \\ \beta_i \end{bmatrix} = \begin{bmatrix} \cos \theta_i & -\sin \theta_i \\ \sin \theta_i & \cos \theta_i \end{bmatrix} \begin{bmatrix} \alpha_i \\ \beta_i \end{bmatrix}, \quad (9)$$

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; θ_i is the rotating angle, and the value and the sign of θ_i are determined by the adjustment strategy [17, 18]. θ_i is static in conventional quantum genetic algorithm. The value of θ_i can be dynamically adjusted based on the evolutionary process in the improved quantum genetic algorithm. Adjust the value of the rotating angle according to the number of generation; the regulator is as follows:

$$\theta_i = \theta_{\max} - \left(\frac{\theta_{\max} - \theta_{\min}}{it \max} \right) * \text{iter}, \quad (10)$$

where θ_i is the value of the rotating angle of the i th generation, θ_{\max} is the maximal rotating angle, θ_{\min} is the minimal rotating angle, $it \max$ is the maximal generation, and $iter$ is the current generation. $\theta_{\max} = 0.01\pi$, $\theta_{\min} = 0.001\pi$.

The adjustment strategy is as follows: comparing the fitness of the currently measured value $f(x)$ of the individual q_j^t with the fitness of the current optimal individual $f(\text{best}_i)$, if $f(x) > f(\text{best}_i)$, then adjust the corresponding qubits of q_j^t , making the probability amplitude (α_i, β_i) evolve toward the direction that is propitious to the emergence of x_i . Conversely, if $f(x) < f(\text{best}_i)$, then adjust the corresponding qubits of q_j^t , making the probability amplitude (α_i, β_i) evolve toward the direction that is propitious to the emergence of best_i .

Construct a determinant $A = \begin{vmatrix} \alpha_b & \alpha_i \\ \beta_b & \beta_i \end{vmatrix}$, where (α_b, β_b) is the probability amplitude of some qubit of the optimal solution which has been searched. (α_i, β_i) is probability amplitude of corresponding state of qubit in the current solution. When $x_i \neq \text{best}_i$, 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.

4.2.2. Quantum Mutation. Add quantum mutation operation in conventional quantum genetic algorithm. Quantum mutation enables some individual to deviate slightly from the current evolutionary direction and prevent the evolution of individual into a local optimal solution [19]. Quantum mutation can completely reverse the individual's evolutionary direction by swapping the value of probability amplitude of qubits (α, β) . Quantum NOT gates are adopted to realize chromosomal variation.

TABLE 3: One result of the whole interference crossover.

Chromosome number	New chromosome							
1	A(1)	D(2)	C(3)	B(4)	A(5)	D(6)	C(7)	B(8)
2	B(1)	A(2)	D(3)	C(4)	B(5)	A(6)	D(7)	C(8)
3	C(1)	B(2)	A(3)	D(4)	C(5)	B(6)	A(7)	D(8)
4	D(1)	C(2)	B(3)	A(4)	D(5)	C(6)	B(7)	A(8)

4.2.3. Quantum Disaster. Add quantum disaster operation in the conventional quantum genetic algorithm. The method is to apply a large disturbance to some individuals in the population and regenerate some other new random individuals [20].

4.2.4. Whole Interference Crossover. Whole interference crossover operator is utilized to realize evolution instead of the traditional crossover operator. All chromosomes are involved in the whole interference crossover. Set the number of population to be m , and length of chromosome is n ($m < n$). A_{ij} is the j th bit of the i th individual. Then, the chromosome of the i th individual of the new family through the whole interference crossover is $A_{(i \bmod m)1} - A_{(i+1 \bmod m)2} - \dots - A_{(i+k \bmod m)(k+1)} - \dots - A_{(i+n-1 \bmod m)n}$. The new whole interference crossover operator is mainly used when the algorithm converges to local optimal solution to generate new individuals to overcome the prematurity in later evolution. Suppose that there are 4 populations and the lengths of them are all 8. One result of the whole interference crossover is as in Table 3.

It is important to note that quantum rotating gates, quantum mutation, and quantum disaster all act on quantum chromosome. However, whole interference crossover acts on the chromosome generated by quantum chromosome. The evolution performance will not be introduced if the whole interference crossover acts on quantum chromosome.

The improved strategy of quantum genetic algorithm mainly contains self-adaptive rotating angle strategy, the improvement of program structure of quantum evolution, added quantum mutation operation, quantum disaster operation, and whole interference crossover operation. The flowchart of improved quantum genetic algorithm is shown in Figure 3.

4.3. The Determining Flow of Scheduling Engineering Personnel. The steps to determine the scheme of scheduling engineering personnel are as follows.

Step 1 (initialize distribution scheme). Initialize allocation scheme to form the initial population according to the needs of the evolution of quantum genetic algorithm and the assignment matrix.

Step 2 (time calculation). Calculate the time of all the allocated schemes in the population.

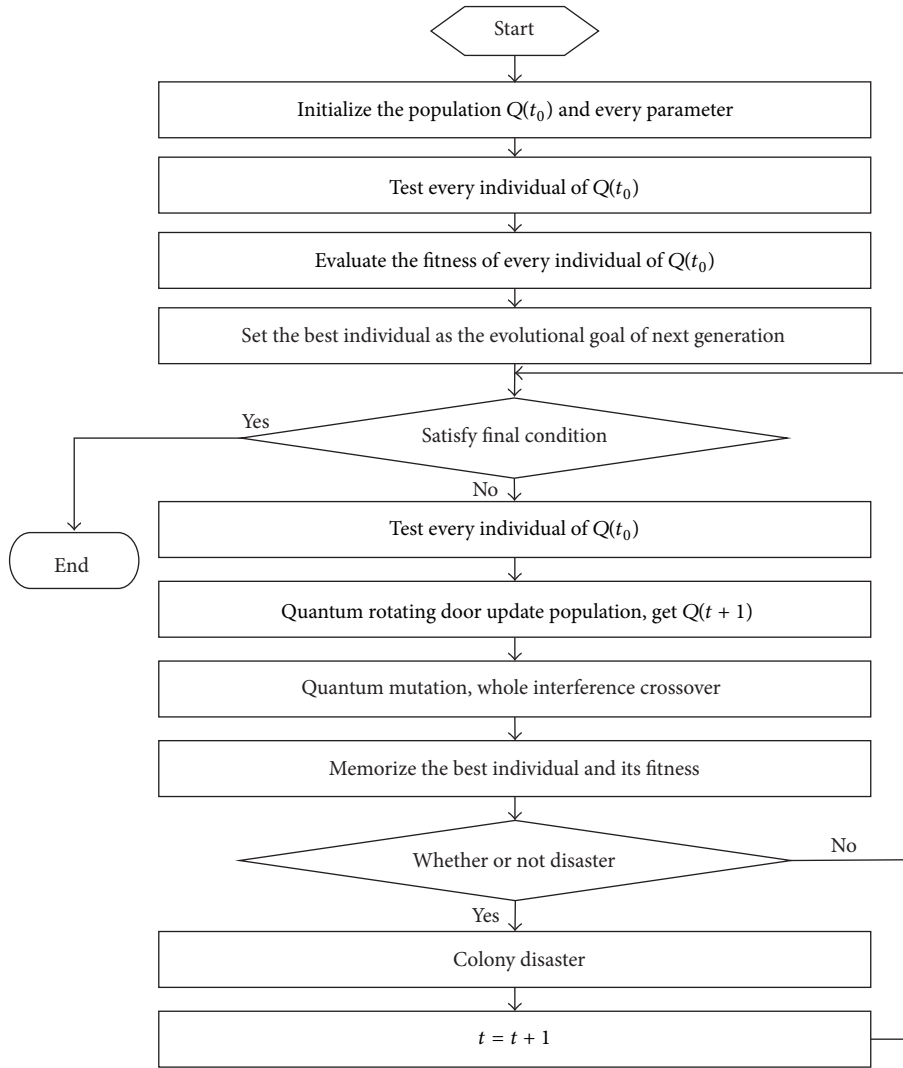


FIGURE 3: Flowchart of improved quantum genetic algorithm.

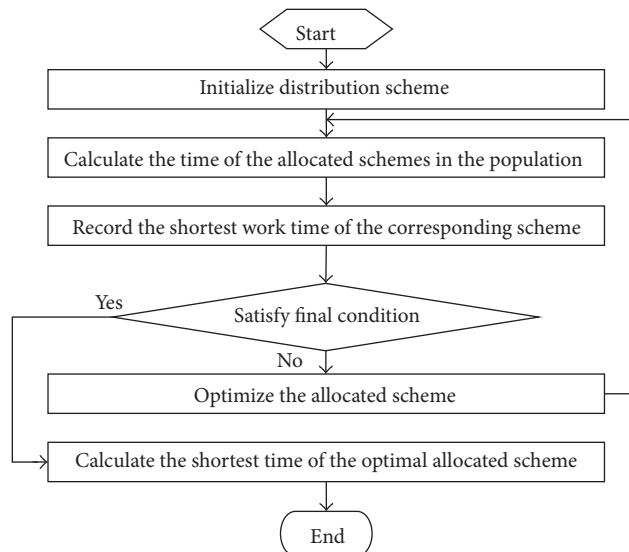


FIGURE 4: Flowchart of acquiring the optimal allocated scheme and the shortest work time.

Step 3 (optimize the allocation scheme). Apply the improved quantum genetic algorithm to optimize the allocation scheme in order to get the optimal allocation scheme. Go to *Step 2*, according to the termination condition, to do the judgment; if the termination condition is satisfied, then go to the End or else do the loop to *Step 3*.

Step 4 (determination of the optimal allocation scheme and the shortest work time). Choose the optimal allocation scheme from the population satisfied the termination condition. Calculate the work time of the optimal allocation scheme.

The flowchart is shown in Figure 4.

5. Simulation Analyses

Take some road building project as an example. There are 12 roads which need to be built. The building work can be divided into three procedures: prospecting, drawing and location designing, and construction. There are 3 prospecting groups, 2 drawing and location designing groups, and 3 construction groups. The capacities of the groups are different. The specific work times are shown in Table 4.

The specific steps of acquiring the optimal allocated scheme and the shortest work time are as follows.

(1) Construct 12×3 dimensional matrix:

$$A_{12 \times 3} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ \vdots & \vdots & \vdots \\ a_{121} & a_{122} & a_{123} \end{bmatrix}, \tag{11}$$

where $1 \leq a_{11}, a_{21}, a_{31}, \dots, a_{121} < 4$, $1 \leq a_{12}, a_{22}, a_{32}, \dots, a_{122} < 3$, $1 \leq a_{13}, a_{23}, a_{33}, \dots, a_{123} < 4$. The corresponding qubit chromosome is

$$Q_{12 \times 3} = \begin{bmatrix} \alpha_{111} & \alpha_{112} & \alpha_{121} & \alpha_{131} & \alpha_{132} \\ \beta_{111} & \beta_{112} & \beta_{121} & \beta_{131} & \beta_{132} \\ \alpha_{211} & \alpha_{212} & \alpha_{221} & \alpha_{231} & \alpha_{232} \\ \beta_{211} & \beta_{212} & \beta_{221} & \beta_{231} & \beta_{232} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \alpha_{1211} & \alpha_{1212} & \alpha_{1221} & \alpha_{1231} & \alpha_{1232} \\ \beta_{1211} & \beta_{1212} & \beta_{1221} & \beta_{1231} & \beta_{1232} \end{bmatrix}. \tag{12}$$

Suppose the generated matrix is

$$A_{12 \times 3} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ \vdots & \vdots & \vdots \\ a_{121} & a_{122} & a_{123} \end{bmatrix} = \begin{bmatrix} 1.2 & 2.3 & 1 & 3.6 & 2.1 & 1.8 & 1.3 & 3.2 & 2.5 & 2.7 & 3.4 & 1.7 \\ 2.9 & 1.2 & 1.6 & 2.3 & 1.1 & 2.5 & 2.7 & 1.4 & 2.2 & 1.3 & 1.6 & 2.8 \\ 3.7 & 1.5 & 2.4 & 1.6 & 3.3 & 2.8 & 2.5 & 3.6 & 1.9 & 2.1 & 1.9 & 2.2 \end{bmatrix}^T \tag{13}$$

$$A'_{12 \times 3} = \begin{bmatrix} 1 & 2 & 1 & 3 & 2 & 1 & 1 & 3 & 2 & 2 & 3 & 1 \\ 2 & 1 & 1 & 2 & 1 & 2 & 2 & 1 & 2 & 1 & 1 & 2 \\ 3 & 1 & 2 & 1 & 3 & 2 & 2 & 3 & 1 & 2 & 1 & 2 \end{bmatrix}^T.$$

Then, the encoding section of the assigned chromosome is

$$[1, 2, 1, 3, 2, 1, 1, 3, 2, 2, 3, 1, 2, 1, 1, 2, 1, 2, 2, 1, 2, 1, 1, 2, 3, 1, 2, 1, 2]. \tag{14}$$

(2) Seek the values of time $b_{k|a_{ki}}$ corresponding with $W_k M i_{a_{ki}}$ (where $1 \leq k \leq 12$, $1 \leq i \leq 3$, i, j, k are integers) in Table 2. Then, the work timetable of every engineering work is gotten. In Table 5, the figure denotes the work time, and the subscript denotes the work group.

(3) Apply the method of Section 3 to calculate the work times of every work group according to the order determined by Step (2) and the time determined by Step (3). Then, find the maximum value as the total engineering work time.

Apply GA (genetic algorithm), QGA (quantum genetic algorithm), and IQGA (improved quantum genetic algorithm) to solve the problem of scheduling engineering personnel. The three kinds of algorithm are written and compiled in MATLAB. The simulation test in this paper is based on the AMD CPU 1.8 GHz, 1.5 GB RAM. The parameters of GA are set as follows: GA is encoded by real number, the total evolution generation is 80, the size of population is 40, and the probability of mutation is 0.01. The parameters of QGA are set as follows: QGA is encoded by the binary, the total evolution generation is 80, the size of population is 40, the length of the variable expressing 3 work groups is 2, and the length of the variable expressing 2 work groups is 1. The parameters of IQGA are set as follows: IQGA is encoded by the binary, the total evolution generation is 80, the size of population is 40, the length of the variable expressing 3 work groups is 2, the length of the variable expressing 2 work groups is 1, and the probability of mutation is 0.01. The fitness function is the objective function. The fitness function tends to the optimal

TABLE 4: Work time of every work group.

	$M1_1$	J_1 $M1_2$	$M1_3$	$M2_1$	J_2 $M2_2$	$M3_1$	J_3 $M3_2$	$M3_3$
W_1	3	2	4	3	4	5	3	4
W_2	4	5	3	6	5	5	6	3
W_3	4	3	5	4	2	3	4	5
W_4	2	5	3	5	4	1	5	4
W_5	4	2	6	3	1	3	4	6
W_6	1	5	3	2	3	5	2	7
W_7	7	6	8	4	6	6	7	6
W_8	6	4	5	7	5	4	8	5
W_9	2	5	4	1	2	3	6	3
W_{10}	3	6	4	3	4	4	3	9
W_{11}	5	2	3	3	5	8	5	4
W_{12}	4	3	4	6	5	4	7	5

TABLE 5: Work time of every engineering work and corresponding work group.

Engineering work	Process		
	J_1	J_2	J_3
W_1	3_1	4_2	4_3
W_2	5_2	6_1	5_1
W_3	4_1	4_1	4_2
W_4	3_3	4_2	1_1
W_5	2_2	3_1	6_3
W_6	1_1	3_2	2_2
W_7	7_1	6_2	7_2
W_8	5_3	7_1	5_3
W_9	5_2	2_2	3_1
W_{10}	6_2	3_1	3_2
W_{11}	3_3	3_1	8_1
W_{12}	4_1	5_2	7_2

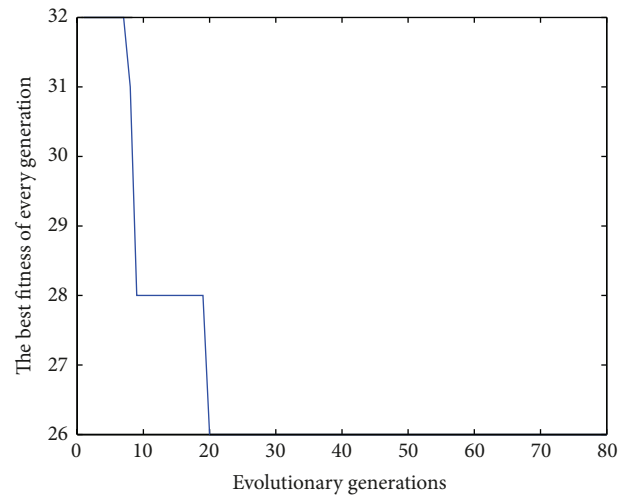


FIGURE 5: The changing curve of work time of the best individual in the population.

value while the evolution increases. Simulate the process of evolution. Ten rounds of experiments were presented in this paper. The simulation results of the 3 kinds of algorithm are in Tables 6, 7, and 8. The result of some time of the IQGA is shown in Figures 5, 6, and 7.

The statistical results of the experiment are shown in Table 9.

Table 9 shows that compared with GA, QGA can get its smallest value in a short time; however, QGA can hardly get the optimal result. Though the evolution time of the IQGA is longer and the evolution generation of the IQGA is more, the IQGA can get the optimal result in a bigger probability.

In Figure 5, x -axis represents the evolution generations, and y -axis represents the work time. The work time gradually becomes shorter with the increase of the generation.

In Figure 6, x -axis represents the evolution algebra and y -axis represents the work time. The average work time of all individuals has a decreasing trend, but there is a gap between the optimal value and the average work time.

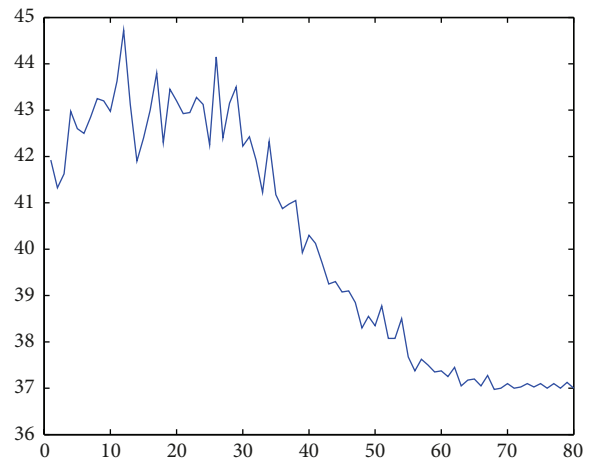


FIGURE 6: The changing curve of average work time of all individuals.

TABLE 6: The simulation results of ten rounds of experiments of GA.

Round	1	2	3	4	5	6	7	8	9	10
Simulation result	27	26	26	28	27	29	28	28	28	28
Simulation time (second)	3.007	2.905	2.894	3.102	2.732	2.940	2.946	2.846	3.468	2.937
Evolution generation	22	20	23	28	19	19	18	29	18	29
Evolution time (second)	0.8269	0.7263	0.8320	1.0857	0.6488	0.6983	0.6629	1.0317	0.7803	1.0647

TABLE 7: The simulation results of ten rounds of experiments of QGA.

Round	1	2	3	4	5	6	7	8	9	10
Simulation result	30	27	29	30	28	29	31	30	29	28
Simulation time (second)	5.679	5.337	5.547	5.576	5.766	6.150	5.813	5.828	6.024	5.802
Evolution generation	12	13	10	4	4	15	3	4	8	21
Evolution time (second)	0.8519	0.8673	0.6934	0.2788	0.2883	1.1531	0.2180	0.2914	0.6024	1.5230

TABLE 8: The simulation results of ten rounds of experiments of IQGA.

Round	1	2	3	4	5	6	7	8	9	10
Simulation result	27	28	26	27	28	27	27	26	28	26
Simulation time (second)	8.847	8.580	8.527	8.459	8.505	8.364	8.422	8.712	8.342	8.524
Evolution generation	18	10	12	12	11	9	13	22	21	16
Evolution time (second)	1.9906	1.0725	1.2791	1.2689	1.1694	0.9410	1.3686	2.3958	2.1898	1.7048

TABLE 9: The statistical results of the experiment.

Algorithm	The optimal result	The worst result	The average result	The fastest evolution time (s)	The slowest evolution time (s)	The average evolution time (s)	Times of convergence
GA	26	29	27.5	0.6488	1.0857	0.8358	2
QGA	27	31	29.1	0.2788	1.5230	0.6768	0
IQGA	26	28	27	0.9410	2.3958	1.5381	3

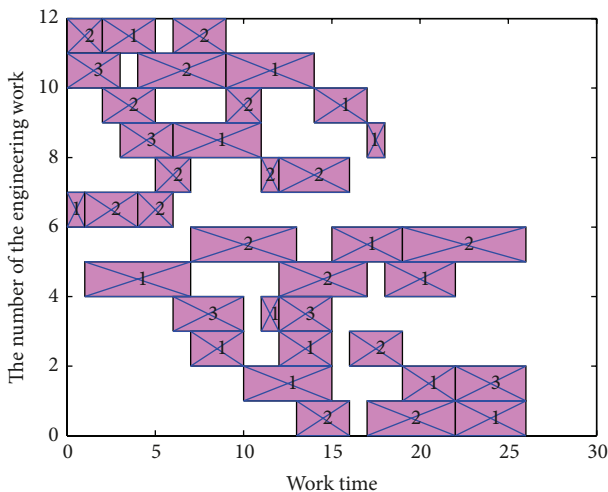


FIGURE 7: Gantt chart of the optimal scheme.

In Figure 7, x -axis represents time; y -axis represents the number of the engineering work. The length of the rectangles represents the work time of the engineering work; the number in the rectangle represents the number of the work group of the corresponding process.

Engineering personnel can be scheduled based on the Gantt chart of the optimal scheme. The role of the Gantt chart is to aid in allocation decisions of scheduling engineering personnel.

6. Conclusions

Compared with genetic algorithm and standard quantum evolutionary algorithm, the advantages of improved quantum genetic algorithm are as follows: (1) IQGA adopts directed evolution which can find the optimal solution more quickly than undirected evolution; (2) the self-adaptive quantum rotation angle with larger rotation angle at the early stage of the evolution and with smaller rotation angle in the late evolution is advantageous for exploration and exploitation; (3) quantum mutation is good to generate new quantum chromosome to increase the diversity of population to extend the search space; (4) quantum disaster can make evolution jump out of local optimal solution to increase the search probability of the global optimal solution; (5) the whole interference crossover takes effect on the level of chromosome coding generated by quantum chromosome to generate new chromosome coding. The search space is effectively expanded and the probability to find optimal solution is increased.

So, IQGA can search the optimal solution faster and more efficiently based on the above-mentioned advantages.

Application of improved quantum genetic algorithm and appropriate mathematical model can solve the complex scheduling engineering personnel problem. Through this method, we can obtain the optimal allocation scheme, and the work time of the allocation scheme is also determined. The Gantt chart of the result clearly describes the work time and work order of each engineering work. The method in this paper can provide powerful support for the scheduling engineering personnel. The proposed method in this paper not only can be applied in the scheduling engineering personnel but also can be used for similar types of problem with multistage, multiple projects, and multiple subjects and the abilities of subjects are different, such as other scheduling problems and flexible flow shop scheduling problem.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

- [1] P. Brucker, R. Qu, and E. Burke, "Personnel scheduling: models and complexity," *European Journal of Operational Research*, vol. 210, no. 3, pp. 467–473, 2011.
- [2] T. Lapègue, O. Bellenguez-Morineau, and D. Prot, "A constraint-based approach for the shift design personnel task scheduling problem with equity," *Computers and Operations Research*, vol. 40, no. 10, pp. 2450–2465, 2013.
- [3] S.-W. Lin and K.-C. Ying, "Minimizing shifts for personnel task scheduling problems: a three-phase algorithm," *European Journal of Operational Research*, vol. 237, no. 1, pp. 323–334, 2014.
- [4] M. Krishnamoorthy, A. T. Ernst, and D. Baatar, "Algorithms for large scale shift minimisation personnel task scheduling problems," *European Journal of Operational Research*, vol. 219, no. 1, pp. 34–48, 2012.
- [5] G. DuCote and E. M. Malstrom, "Design of personnel scheduling software for manufacturing," *Computers and Industrial Engineering*, vol. 37, no. 1, pp. 473–476, 1999.
- [6] P. M. Koeleman, S. Bhulai, and M. van Meersbergen, "Optimal patient and personnel scheduling policies for care-at-home service facilities," *European Journal of Operational Research*, vol. 219, no. 3, pp. 557–563, 2012.
- [7] J. van den Bergh, J. Beliën, P. de Bruecker, E. Demeulemeester, and L. de Boeck, "Personnel scheduling: a literature review," *European Journal of Operational Research*, vol. 226, no. 3, pp. 367–385, 2013.
- [8] M. Sabar, B. Montreuil, and J.-M. Frayret, "A multi-agent-based approach for personnel scheduling in assembly centers," *Engineering Applications of Artificial Intelligence*, vol. 22, no. 7, pp. 1080–1088, 2009.
- [9] K.-H. Han and J.-H. Kim, "Genetic quantum algorithm and its application to combinatorial optimization problem," in *Proceedings of the Congress on Evolutionary Computation*, pp. 1354–1360, Piscataway, NJ, USA, July 2000.
- [10] A. Narayanan and M. Moore, "Quantum-inspired genetic algorithms," in *Proceedings of the IEEE International Conference on Evolutionary Computation (ICEC '96)*, pp. 61–66, Nagoya, Japan, May 1996.
- [11] H. Wang, J. Liu, J. Zhi, and C. Fu, "The improvement of quantum genetic algorithm and its application on function optimization," *Mathematical Problems in Engineering*, vol. 2013, Article ID 730749, 10 pages, 2013.
- [12] J. Tang, G. Zhang, B. Lin, and B. Zhang, "Hybrid genetic algorithm for flow shop scheduling problem," in *Proceedings of the International Conference on Intelligent Computation Technology and Automation (ICICTA '10)*, vol. 2, pp. 449–452, Changsha, China, May 2010.
- [13] R. Zhang, W. Wei, Z. Jiang, and X. Ma, "Research and simulation on flow-shop scheduling problem based on improved genetic algorithm," in *Proceedings of the 7th International Conference on Computer Science and Education (ICCSE '12)*, pp. 916–919, July 2012.
- [14] F. Jin and Y. Fu, "Master-slave genetic algorithm for flow shop scheduling with resource flexibility," in *Proceedings of the IEEE International Conference on Advanced Management Science (ICAMS '10)*, vol. 1, pp. 341–346, Chengdu, China, July 2010.
- [15] G. Zhang and H. Rong, "Quantum-inspired genetic algorithm based time-frequency atom decomposition," in *Proceedings of the 7th international conference on Computational Science (ICCS '07)*, 2007.
- [16] J.-A. Yang and Z.-Q. Zhuang, "Actuality of research on quantum genetic algorithm," *Journal of Computer Science & Technology*, vol. 30, no. 11, pp. 13–15, 2003.
- [17] W. Shu and B. He, "A quantum genetic simulated annealing algorithm for task scheduling," in *Advances in Computation and Intelligence*, L. Kang, Y. Liu, and S. Zeng, Eds., vol. 4683 of *Lecture Notes in Computer Science*, pp. 169–176, Springer, Berlin, Germany, 2007.
- [18] Y. Li, Y. Zhang, Y. Cheng, and X. Jiang, "A novel immune quantum-inspired genetic algorithm," in *Advances in Natural Computation*, vol. 3612 of *Lecture Notes in Computer Science*, pp. 215–218, 2005.
- [19] K. Han and J. Kim, "Genetic quantum algorithm and its application to combinatorial optimization problem," in *Proceedings of the Congress on Evolutionary Computation*, pp. 1354–1360, San Diego, Calif, USA, July 2000.
- [20] H. Miao, H. Wang, and Z. Deng, "Quantum genetic algorithm and its application in power system reactive power optimization," in *Proceedings of the International Conference on Computational Intelligence and Security (CIS '09)*, vol. 1, pp. 107–111, Beijing, China, 2009.