

Research Article

A New Improved Quantum Evolution Algorithm with Local Search Procedure for Capacitated Vehicle Routing Problem

Ligang Cui, Lin Wang, Jie Deng, and Jinlong Zhang

School of Management, Huazhong University of Science and Technology, Wuhan, Hubei 430074, China

Correspondence should be addressed to Lin Wang; wanglin982@gmail.com

Received 26 June 2013; Accepted 11 October 2013

Academic Editor: John Gunnar Carlsson

Copyright © 2013 Ligang Cui 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.

The capacitated vehicle routing problem (CVRP) is the most classical vehicle routing problem (VRP); many solution techniques are proposed to find its better answer. In this paper, a new improved quantum evolution algorithm (IQEA) with a mixed local search procedure is proposed for solving CVRPs. First, an IQEA with a double chain quantum chromosome, new quantum rotation schemes, and self-adaptive quantum Not gate is constructed to initialize and generate feasible solutions. Then, to further strengthen IQEA's searching ability, three local search procedures 1-1 exchange, 1-0 exchange, and 2-OPT, are adopted. Experiments on a small case have been conducted to analyze the sensitivity of main parameters and compare the performances of the IQEA with different local search strategies. Together with results from the testing of CVRP benchmarks, the superiorities of the proposed algorithm over the PSO, SR-1, and SR-2 have been demonstrated. At last, a profound analysis of the experimental results is presented and some suggestions on future researches are given.

1. Introduction

Vehicle routing problem (VRP) has been presented more than fifty years by Dantzig and Ramser [1], and it is still drawing many researchers' attention [2, 3]. In a VRP, a vertex usually denotes the distribution center and other vertexes are used to denote the clients on demand; the distance between the vertexes is usually considered as the traveling distance or traveling cost; a classical VRP [4, 5] may include two principles: each client needs to be visited only once and the starting point and terminal point are both at the distribution center. So the objective of a VRP is to minimize the total distances.

It is well known that VRP is an NP-hard problem [6]. While CVRP is the most basic problem of VRPs and studying CVRP has a fundamental meaning to learn other advanced problems in this field and develop its solution methodologies [7]. Laporte [8] reviewed the VRP algorithms development from 1959 to 2009; these algorithms were classified as exact algorithm, classical heuristic algorithm, and metaheuristic algorithm. As it was stated in his paper, solving an NP-hard

CVRP will require more computational time but the outcome may not be improved a lot.

In recent years, the solving methods of VRPs are mainly on finding a proper intelligent or heuristic algorithm with problem-oriented solving mechanisms to find near optimal solutions [9]. Vera et al. [10] used the variable neighborhood search (VNS) heuristic algorithm solving the periodic VRP (PVRP) without time window. Barker and Ayechev [11] proposed a genetic algorithm for VRP and Rego and Alidaee [12] reviewed some of the most important tabu search heuristics for the VRPs. Marinakis et al. [13] proposed a hybrid particle swarm optimization algorithm for the VRP, while Szeto et al. [14] used a bee colony algorithm for the CVRP. These algorithms have been proven successfully in solving VRPs. They are widely accepted as being more efficient than some exact algorithms [4] to some extent.

In 1996, new evolutionary algorithm called quantum-inspired algorithm was proposed [15], which combined the quantum mechanism and the evolution computing characters together, so Quantum-inspired algorithm is also called the quantum evolution algorithm (QEA). QEA has been

widely used in a variety of fields, such as SVM parameter optimization [16] and nuclear reactor reload optimization problem [17].

QEA was born to solve the TSP; there have been only a few satisfactory results in dealing with the VRP, such as Zhang et al. [18] and Zhang et al. [19]. Current QEA-based methods for solving VRPs have three main shortages. (1) The first one is the complex encoding process. Classical QEA encoding procession need to be converted among quantum encoding-binary encoding-real number encoding-integer encoding, which increases the computational complexity. Nevertheless, this conversion has lost half of quantum encoding carrying information. (2) Another disadvantage of classical QEA is its initialization largely based on the encoding method, which has separated the relations between visiting customer and candidate vehicles. (3) The updating operation is stylized as it relies on the lookup table. The non problem-oriented process also impacts QEA searching efficiency.

Therefore, an improved quantum evolution algorithm (IQEA) with a mixed local search procedure called IQEA+ is proposed for solving CVRPs, in which a new double chain based encoding method to deal with the relations between vehicles and clients, a revised quantum rotation mechanism to simplify the updating processions, and an adaptive mutation operator are adopted. To further strengthen the searching abilities of IQEA, three local search procedures, 1-1 exchange, 1-0 exchange, and 2-OPT are also adopted.

The remaining contents are organized as follows. In Section 2, we review the fundamental knowledge and the improvements of QEA. Section 3 mainly discusses the encoding mechanism, the revised updating method, and mutation operations. Moreover, the framework of the IQEA is shown in this section. While the numerical experiments and results are presented in Section 4. In Section 5, conclusions and future research directions are provided.

2. Literature Review of QEA

2.1. Basic Knowledge of QEA. Quantum-inspired algorithm was born to solve the TSP [15]. QEA is also a probability algorithm, which includes some similar features of the evolutionary algorithm (EA), such as the representation of the individuals, the fitness functions, the updating processes, and the mutations [27]. However, it is not quite the same as a classical EA for it is mainly constructed with the concept of quantum computation [28] that reflects microparticle behaviors based on quantum mechanics. In QEA, an individual of a chromosome is usually expressed by a certain number of quantum bits (Q-bits), which is represented as a linear superposition mode. The state of a Q-bit can be expressed as follow:

$$|\Psi\rangle = \alpha |0\rangle + \beta |1\rangle, \quad (1)$$

where α and β are complex numbers. In QEA, they are called probability amplitudes of $|0\rangle$ and $|1\rangle$, respectively. The smallest unit of information stored in a Q-bit may be in $|0\rangle$, $|1\rangle$, or any superposition of the two states, while $|\alpha|^2$ and $|\beta|^2$

denote the probabilities of the Q-bit in “0” state and in “1” state and they satisfy the normalization equation:

$$|\alpha|^2 + |\beta|^2 = 1. \quad (2)$$

Then, we can construct a population of Q-bit individuals in the t th generation like that $Q(t) = \{q_1^t, q_2^t, \dots, q_n^t\}$, where n is the size of the population and q_j^t is the j th chromosome of a certain population in the t th generation. Therefore, a string of n Q-bits can be expressed as follows:

$$q_j^t = \left[\begin{array}{c|c|c|c} \alpha_1^t & \alpha_2^t & \dots & \alpha_n^t \\ \beta_1^t & \beta_2^t & \dots & \beta_n^t \end{array} \right], \quad (3)$$

where t is the t th generation, m is the length of q_j^t , j is the j th chromosome, and $j = 1, 2, \dots, n$. The expression can also be described as another form:

$$q_j^t = \left[\begin{array}{c|c|c|c} \sin(\theta)_1^t & \sin(\theta)_2^t & \dots & \sin(\theta)_n^t \\ \cos(\theta)_1^t & \cos(\theta)_2^t & \dots & \cos(\theta)_n^t \end{array} \right], \quad (4)$$

where $\theta \in [0, 2\pi]$.

A basic QEA procedure is as follows.

(1) Set the maximum generation and the population size. Initialize the population $Q(t)$ where $t = 0$. A general method is as follows: let all $2 \times m \times n$ probability amplitudes be equal to $1/\sqrt{2}$, which means that all the possible linear superposition states are endowed with the same probability $1/\sqrt{2^m}$; that is,

$$|\Psi_{t=0}^{q_j}\rangle = \sum_{k=1}^{2^m} \frac{1}{\sqrt{2^m}} |s_k\rangle, \quad (5)$$

where S_k is a binary string (x_1, x_2, \dots, x_m) denotes the k th state of the quantum chromosome, and $x_i = 0, 1$; $i = 1, 2, \dots, m$.

(2) Observe the $Q(t)$ to obtain its binary solution set $P(t) = (x_1^t, x_2^t, \dots, x_m^t)$. Each x_j^t ($j = 1, 2, \dots, n$) is a binary string ($X_1^t, X_2^t, \dots, X_m^t$) with a length of m and $X_i^t = 0$ or 1 ($i = 1, 2, \dots, m$) according to certain conditions: randomly generate a number p which lies in $[0, 1]$ if p is larger than the square of probability amplitude of the i th Q-bit, then we take “1” in the binary state; otherwise, “0” is adopted in the binary state.

(3) Convert the binary solutions to real number solutions corresponding to specific questions. Use a fitness function to evaluate every individual and store the best solutions; if the object reached, stop the algorithm; otherwise, turn to step 4.

(4) Update and execute mutation operations. Different from recombination and mutation operations of conventional EA, quantum-gate (Q-gate) is adopted in QEA to update the current solutions. There are several kinds of Q-gates, such as the NOT gate, controlled-Not gate, Hadamard gate, R & N_e gate and Phase gate [29], but all of them must meet requirement of the Schrödinger equation; that is,

$$UU^T = I, \quad (6)$$

TABLE 1: The lookup table of rotation angle.

x_i	b_i	$f(x_i) \geq f(b_i)$	$\Delta\theta_i$	$s(\alpha_i, \beta_i)$			
				$\alpha_i\beta_i > 0$	$\alpha_i\beta_i < 0$	$\alpha_i = 0$	$\beta_i = 0$
0	0	False	0	0	0	0	0
0	0	True	0	0	0	0	0
0	1	False	0	0	0	0	0
0	1	True	0.05π	-1	+1	± 1	0
1	0	False	0.01π	-1	+1	± 1	0
1	0	True	0.025π	+1	-1	0	± 1
1	1	False	0.005π	+1	-1	0	± 1
1	1	True	0.025π	+1	-1	0	± 1

where U is a Q-gate and I is a unit matrix. Because QEA's updating and mutation operations are mainly constructed based on a single Q-bit, the single quantum bit gate—quantum rotation gate—has been widely used. A rotation gate $U(\theta_i)$ as an updating operator is employed in updating a Q-bit $[\alpha_i, \beta_i]^T$:

$$\begin{bmatrix} \alpha'_i \\ \beta'_i \end{bmatrix} = U(\theta_i) \begin{bmatrix} \alpha_i \\ \beta_i \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} \alpha_i \\ \beta_i \end{bmatrix}, \quad (7)$$

where θ_i is the rotation angle and its magnitude is obtained by the equation $\theta_i = s(\alpha_i, \beta_i)\Delta\theta_i$. $s(\alpha_i, \beta_i)$ is the sign of the rotation angle whose lookup table is shown in Table 1. In Table 1, b_i is the best solution B of i th Q-bit and x_i is the binary solution X of the i th bit, respectively.

While in the mutation operation, the quantum NOT gate is commonly used. The essence of the NOT gate is to swap the positions of a Q-bit. A NOT gate U is expressed as follows:

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

The mutation procedure is as follows: firstly, set a mutation probability p_1 and generate a p_2 in $[0, 1]$; secondly, compare p_1 with p_2 ; if p_1 is larger than p_2 , select a certain chromosome, from which a number of Q-bits are chosen to carry out the mutation operation; otherwise, do nothing.

(5) Check whether the maximum generation is reached. If not, $t = t + 1$, go back to step 2; otherwise, stop the iteration.

The flow chart of the original QEA is shown in Figure 1.

2.2. The Development of QEA. Compared with other evolutionary algorithms, better performance of QEA has been showed in balancing the exploration and exploitation [22]. But, for all that, it shows great potential in solving some specific problems, especially for some intractable problems [30], such as in the quantum turing machine and quantum automata, quantum circuits, adiabatic quantum computation and topological quantum computation.

2.2.1. Hybrid Quantum Evolution Algorithms (HQEA). Combining quantum mechanism with other EA algorithms is widely accepted by researchers; here we can just call them the hybrid quantum evolution algorithms (HQEA). Zheng

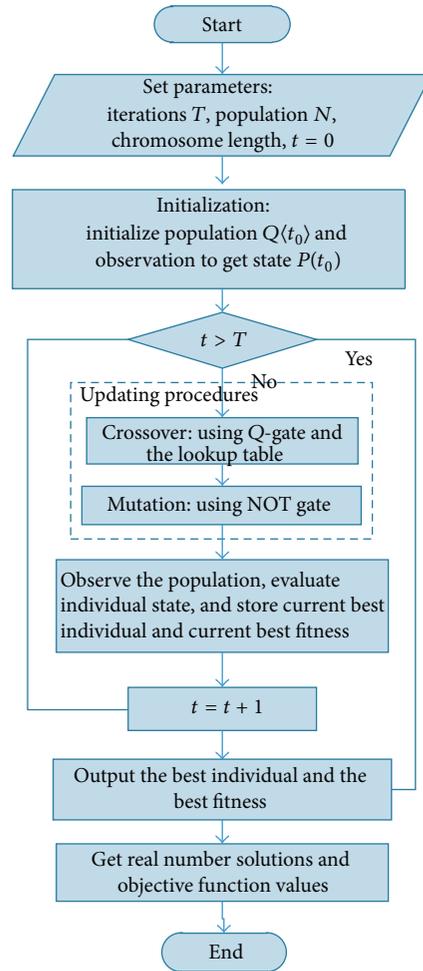


FIGURE 1: Flow chart of original QEA.

and Yamashiro [31] proposed a novel quantum differential evolutionary algorithm (QDEA) with a strategy of largest rotating angle value rule to determine job sequence based on job's quantum information. Su et al. [32] used a quantum-inspired differential evolution (DE) algorithm to discover the classification rules. Similarly, Su and Yang [33] proposed DE/QDE to learn the Takagi-Sugeno (T-S) fuzzy model.

Typically speaking, the quantum mechanism is the most favored to be integrated with the swarm algorithm. For

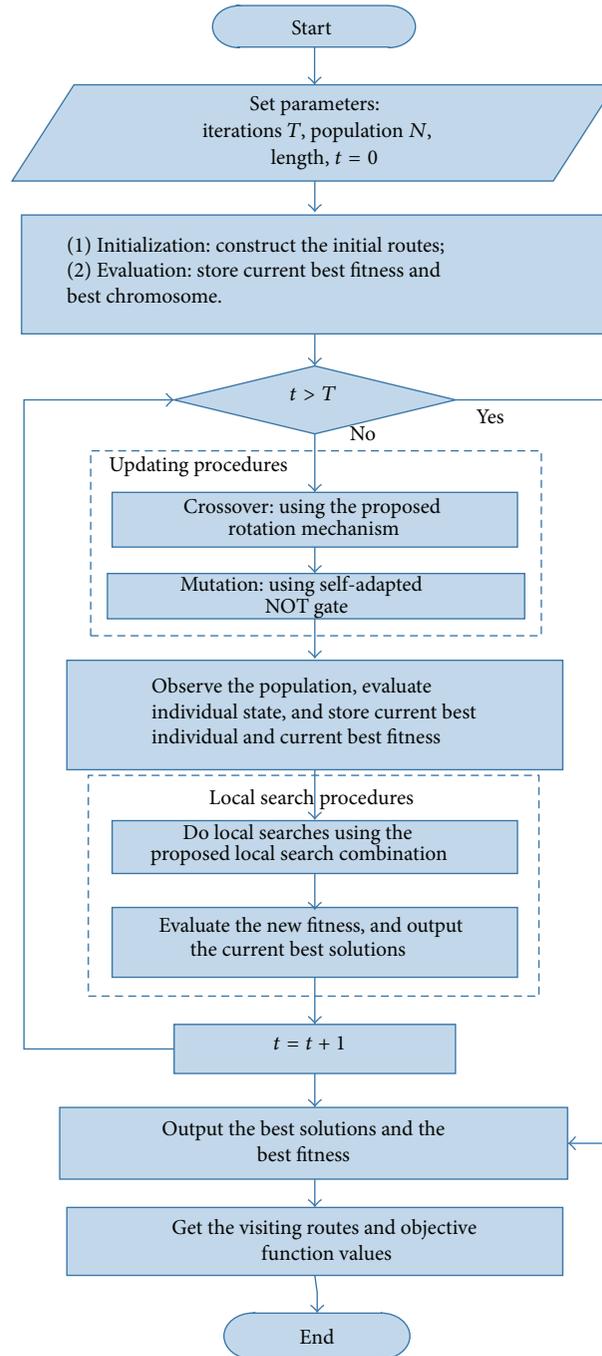
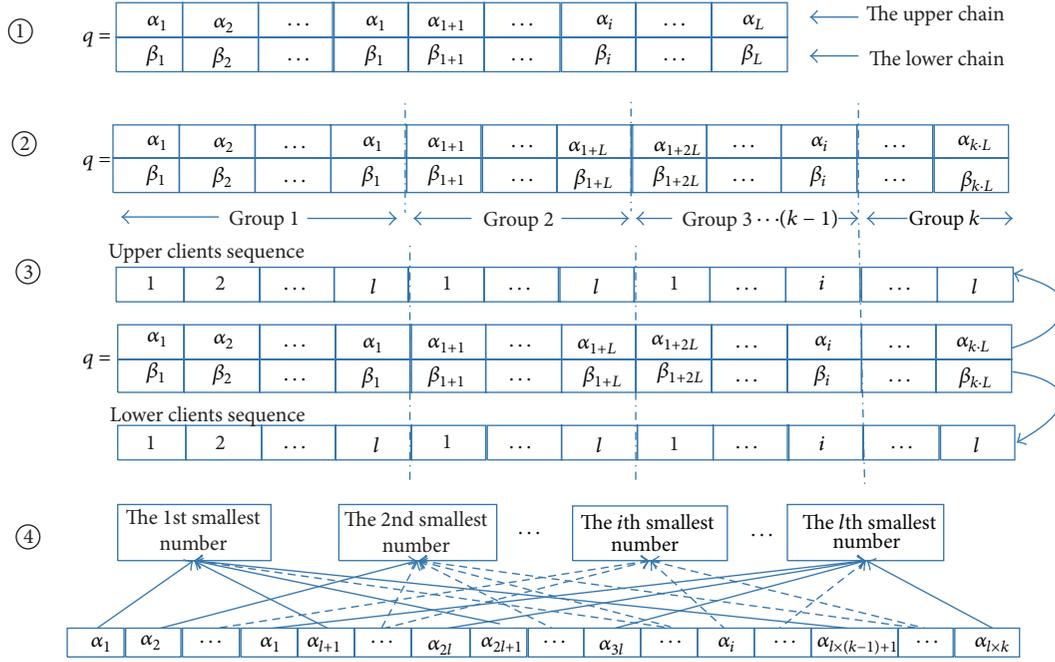


FIGURE 2: Flow chart of IQEA+.

instance, Sun et al. [34] proposed quantum delta potential well based particle swarm optimization algorithm on the hypothesis that the PSO system is a quantum system, each particle of which is of quantum state formulated by the wave function. To deal with the problem of prematurity of particle swarm optimization in global searching, a global convergence guaranteed method called quantum behaved PSO (QPSO) was developed and a vector evaluated QPSO was also presented [35].

2.2.2. Improved QEA (IQEA). Other researchers have paid attention to the improvement of the QEA itself performances, such as redesigning the encoding mechanism, optimizing the rotation angle, and setting properly rotate directions; we can just call these algorithms the improved QEA (IQEA).

P. L. Li and S. Y. Li [21] presented a double chain quantum evolution algorithm (DCQEA), in which each quantum chromosome contained two gene chains and each gene chain represented a feasible solution. While in their another paper,



Where $i = 1, 2, \dots, L$. The dash line arrow means possible relations

FIGURE 3: Solution representations.

a QEA algorithm under the Bloch coordinates of the Q -bits (BQEA) encoding is proposed [36]. Gao and Wang [23] adopted a Q -bit representation to solve the multiobjective optimization problems (MOOPs) and a chaos-based approach is suggested to initialize the population. Based on paralleled quantum chromosomes, Gu et al. [37] used a novel parallel quantum genetic algorithm for stochastic job shop scheduling. In order to improve the efficiency of genetic algorithm in solving the above question, a novel parallel quantum genetic algorithm was presented by Gu et al. [38].

QEA is also a probabilistic algorithm; its solution spaces update is both guided by the best individual and the quantum rotation gate mechanism. Researchers have found that quantum rotation gate has significant impact on its convergence speed, fitness value, and robustness [22]. Zhang and Gao [20] have introduced two types of rotation angles, which can provide a good balance between “fine search” and “coarse search.” By referring to the gradient of fitness function, P. L. Li and S. Y. Li [21] presented a new quantum rotation gate. While Gao and Wang [23] proposed a chaos-based rotation gate which depended on the rank of the parent cell in the population and a chaos equation. Lee et al. [25] constructed a self-adaptive operator which would adjust the size of the grid and was very simple to implement. Some of the above improvement measures are listed in Table 2.

In the encoding stage, the quantum encoding is often transferred into a binary encoding of a certain evolutionary algorithm, for example, the encoding method of the genetic algorithm (GA). The fact is that there are a few papers trying to solve the VRPs or VRPs variants with QEA. However, their

methods of the evolutionary process are mainly on the basic QEA. So, in Section 3, a new improved QEA is introduced to deal with CVRPs with new rotation mechanisms.

3. The Proposed IQEA with Local Search Procedure for CVRP

CVRP is usually formulated as follows: given there are K ($k = 1, 2, \dots, K$) candidate vehicles providing the services for L ($i = 1, 2, \dots, L$) customers and $i = 0$ represented the distribution center. The capacity of each vehicle is b_k ($k = 1, 2, \dots, K$), while the demands of each customer are d_i ($i = 1, 2, \dots, L$). Transport distance/cost from customer i to customer j is $c_{i,j}$. The CVRP model is

$$\min z = \sum_{k=1}^K \sum_{i=0}^L \sum_{j=0}^L c_{ij} x_{i,j,k}, \quad (9)$$

$$x_{i,j,k} = \begin{cases} 1 & \text{vehicle } k \text{ visiting from } i \text{ to } j, \\ 0 & \text{other,} \end{cases} \quad \forall i, j, k, \quad (10)$$

$$y_{i,k} = \begin{cases} 1 & \text{customer } i \text{ is served by } k, \\ 0 & \text{other,} \end{cases} \quad \forall i, k, \quad (11)$$

$$\sum_{i=1}^L d_i y_{i,k} \leq b_k, \quad \forall k, \quad (12)$$

$$\sum_{k=1}^K y_{i,k} = 1, \quad \forall i, \quad (13)$$

TABLE 2: Different quantum rotation angles of QEA.

No.	Different quantum rotation angles	Authors
1	Type I: $\Delta\theta_{j,i}^t = \text{sgn}(\alpha_{j,i}^t \beta_{j,i}^t) \theta_0 \exp(\beta_{j,i}^t / \alpha_{j,i}^t + \gamma)$, where θ_0 is an initial rotation angle and γ is a scale parameter; type II: $\alpha_{j,i}^{t+1} = \alpha_{j,i}^{t+1}/(\text{fix}(k/5) + 1)$; $\beta_{j,i}^{t+1} = \sqrt{1 - (\alpha_{j,i}^{t+1})^2}$, where $\text{fix}(\cdot)$ is a round function and k denotes the number of continuously invalid evolution alleles	Zhang and Gao [20]
2	$\theta = -\text{sgn}(A) \times \theta_0 \times (\ \nabla \text{fixmax} - \nabla \text{fixmin}\ / \ \nabla \text{fix}(X) - \nabla \text{fixmin}\)$, where A is $\sin(\theta_1 - \theta_0)$, θ_0 and θ_1 are the initial and the current best rotation angle, $\ \cdot\ $ is a certain norm, and $\nabla \text{fix}(\cdot)$ is the gradient function of fitness at X .	P. L. Li and S. Y. Li [21]
3	$\theta_i = x \Delta\theta_i(1 + x\lambda_i)$; $\Delta\theta_i = \sqrt{\sum_{n=1}^K (\omega_{bi}^n - \omega_i)/K}$; $x = \text{sgn}(\alpha_i \beta_i(b_i - 0.5))$; $\lambda_0 = f(X_i)$; $\lambda_{t+1} = \mu\lambda_t(1 - \lambda_t)$, where ω_i represents the i th rotation angle and ω_{bi}^n the n th best individual's i th rotation angle; the last two equations are the logistic equations	Hu and Wu [22]
4	$\theta_i = \lambda \cdot \Phi_k^j \cdot \Phi_k^j$; $\Phi_k^j = 8\Phi_{k-1}^j(1 - \Phi_{k-1}^j) - 1$; $\lambda = \gamma\lambda_0((1 - K)/K)$, where the second equation is a modified chaos logistic equation, λ_0 is a control parameter, γ is a temporary parameter, and generally $\gamma = 1$.	Gao and Wang [23]
5	$\Delta\theta = \min \Delta\theta + (\max \Delta\theta - \min \Delta\theta)(\omega_1 \cdot (\ x - b\ /n) + \omega_2 \cdot (f(x) - f(b) /\max\{f(x), f(b)\}))$, where $\max \Delta\theta$ and $\min \Delta\theta$ represent the maximum and minimum rotation angle separately, ω_1 and ω_2 are two predetermined constant numbers, and n is the length of its binary chromosome.	Chen et al. [24]
6	$\theta = k \cdot f(\alpha_i, \beta_i)$; $k = \pi \cdot \exp(-t/\text{iter}_{\max})$, where t is the current iteration of the algorithm and iter_{\max} is the max iteration of the algorithm.	Lee et al. [25]

$$\sum_{i=1}^L x_{i,j,k} = y_{j,k}, \quad \forall j, k, \quad (14)$$

$$\sum_{j=1}^L x_{i,j,k} = y_{i,k}, \quad \forall i, k, \quad (15)$$

$$\sum_{i,j \in S \times S} x_{i,j,k} \leq |S| - 1, \quad S \subset \{1, 2, \dots, L\}, \quad S \neq \Phi, \quad \forall k, \quad (16)$$

where (9) is the objective function; here we want to obtain the minimized total routing distances; $x_{i,j,k}$ in (10) and $y_{i,k}$ in (11) are both the $\{0, 1\}$ decision variables and $x_{i,j,k}$ means that, when the vehicle k visits the customer j from i , $x_{i,j,k}$ is equal to 1; otherwise, $x_{i,j,k}$ is 0 and $y_{i,k}$ means that the vehicle k will visit customer i ; (12) is capacity constraints, which means that the maximum loading of vehicle k can not bypass b_k ; (13) is to guarantee that there is only has one vehicle visiting customer i ; (14) and (15) are used to guarantee that one vehicle will serve only one customer and (16) is used to eliminate subcircle or subroute. The essence of the CVRP is to minimize the total distribution distances under the constraints of capacity.

3.1. The Encoding Mechanism and Solution Presentation. In the encoding stage of classical QEA, quantum codes need to be translated into binary codes, and then the binary codes are continued to be translated into real numbers, which result in encoding procedure that is too complex and trivial to operate. Therefore, a new encoding mechanism is proposed to skip the intermediate encoding steps and take full advantage of quantum information rather than only care about the Q-bit with a bigger probability amplitude. Further, a double chain quantum chromosome is also adopted to improve the searching efficiency. Suppose there are l clients to be served by one distribution center and k vehicles are asked to serve them. Obviously, there are k routes but we do not know which client will be served by which vehicle.

3.1.1. Solution Representations. In this part, in order to construct routes, k groups were set up, and each of them was assigned a string of l numbers which lie in $(0, 1)$. Definitely, the vehicle with bigger number will get more chances to serve the client. The encoding steps are listed below.

- (1) According to the number of vehicles and clients, a quantum chromosome (Q-chrome) length L is randomly generated, where $L = l \times k$, l is the number of clients, and k is the number of vehicles.
- (2) Divide the Q-chrome into k groups; each group has l members, which just correspond to the clients that need to be served.
- (3) Match the given number (quantum amplitude) in each group to the potential client. Make sure the client number is in ascending order (or in descending order).
- (4) Select a number (quantum amplitude) every l length from one chain of Q-chrome, and store the smallest number and its position. The procedure will iterate l times till all l biggest numbers are selected out and all the clients are allocated to the corresponding vehicles. The whole solution representation procedures are shown in Figure 3.

The essence of the four-step assignment is that, if the position of the biggest number is fixed, the represented vehicle will render the service and the client will win the chance to receive the service accordingly. Still, we should remember that there are 2 chains in a Q-chrome, so all the groups and all processing procedures in step 4 will be doubled.

3.1.2. Construct the Vehicle Routes. As the visiting clients in each route have been determined, visiting orders of the clients in each route will be arranged. By far, the smallest numbers we have got in the former part have been all divided into k

TABLE 3: Schemes of the local search operators.

Scheme	Description
(1)	Positive sequence: 1-1 exchange → 1-0 exchange → 2-OPT
(2)	Positive sequence: 1-0 exchange → 1-1 exchange → 2-OPT
(3)	Mixed sequence of 1-0 exchange, 1-1 exchange, and 2-OPT

groups. In one group, the numbers are sorted in descending order. Because the clients' sequence is related to the value of Q-bits in each group, the service (visiting) orders are sorted according to its corresponding value.

To illustrate the encoding and solution presentation clearly, an example with 6 clients 2 vehicles is given below, so a chromosome length 12 (6×2) is randomly generated. The whole process that generates the initial routes and visiting sequences is shown in Figure 4. There are 6 steps to finish the whole routing construction process.

From Figure 4, we could get the sequence of clients assigned to two vehicles. If we define the distribution center as 0, then the complete routes are [0-6-2-0] and [0-1-5-3-4-0]. And, if we consider those operations in the lower chain at the same time, obviously, the searching efficiency of the IQEA will be doubled.

3.2. The Proposed Fitness Function. The minimized objective function is adopted as the fitness function, and a big punishment number is used to deal with unfeasible routes; the fitness function equation can be described as follows:

$$f = \begin{cases} z & \text{feasible solutions,} \\ z + M & \text{otherwise,} \end{cases} \quad (17)$$

where f is the fitness function, z is the objective function of the CVRP model, see (9), and M is a big positive number (10^3 – 10^6). Equation (17) means that, if the solutions we obtained are all feasible, the fitness function is equal to the objective function; if there are unfeasible solutions, a big number will be added to the objective function to eliminate unfeasible routes.

3.3. The Updating Mechanism of IQEA. There are many QEA updating methods; see Table 2. Zhang and Gao's [20] rotation angle is not simple enough, and Lee et al. [25] presented a method that is easy to be understood and implemented, but it is not a problem oriented method. Therefore, a new quantum rotation gate mechanism is used here. As we all know, there are two parts of a whole updating procedure: the magnitudes of quantum rotation angle and the rotation directions which need to be considered in chromosome updating. So these two part procedures are provided below.

3.3.1. Rotation Angle of Q-Bit. In order to let current Q-bits search towards the best Q-bits, the magnitude of current quantum amplitudes and the best quantum amplitudes should be considered together. Because the biggest number is

acquired from the comparisons between different groups in a single Q-chrome, the magnitude of rotation angle is derived from its own chromosome. So the rotation angle calculation method is different from the method in the work of Hu and Wu [22], whose rotation angle mainly depends upon the current and the current best rotation angle. The proposed rotation angle is formulated as follows:

$$\bar{\theta}_i = \frac{1}{k} \sum_{j=0}^{k-1} \theta_{i+j \times L}, \quad (18a)$$

$$\Delta\theta_i = \frac{1}{k} \sqrt{\sum_{j=0}^{k-1} (\theta_i - \bar{\theta}_i)^2}, \quad (18b)$$

where $i = 1, 2, \dots, L$ and $\theta_{i+j \times L}$ is the $(i + j \times L)$ th rotation angle, $\bar{\theta}_i$ is the mean value of each group, and $\Delta\theta_i$ is the variance of numbers in each group. It is quite different from traditional strategies with fixed value in Table 1 that each group has a different deviate value $\Delta\theta_i$.

3.3.2. Rotation Direction of Q-Bit. The rotate direction is as important as the magnitude of rotation angle to QEAs efficiency, which is used to control the current rotation angle to rotate towards the current best rotation angle. If α_1 and β_1 are the probability amplitudes of current optimal solutions, α_2 and β_2 are the probability amplitudes of current solutions, $E_1 = \tan(\beta_1/\alpha_1)$, $\epsilon_1 = |E_1|$, $\theta_1 = \tan^{-1}(\beta_1/\alpha_1)$, $E_2 = \tan(\beta_2/\alpha_2)$, $\epsilon_2 = |E_2|$, and $\theta_2 = \tan^{-1}(\beta_2/\alpha_2)$, then one has the following.

Theorem 1. *Rotation directions selection strategies,*

- (1) when $\epsilon_1 \geq \epsilon_2$ and if $E_2 > 0$, the symbol (of θ_i) is positive, or else, if $E_2 < 0$, the symbol is negative;
- (2) when $\epsilon_1 < \epsilon_2$ and if $E_2 > 0$, the symbol (of θ_i) is negative, or else, if $E_2 < 0$, the symbol is positive.

For the proof of Theorem 1, please refer to Appendix A.

3.4. Mutation Operation. Even though mutation process can increase the diversity of the searching population, effective strategies of mutation operation for QEA have not drawn too much attention. So, to some extent, the quantum Not gate may have played a role in local searches, but it still depends on the mutation rate. If the mutation probability is too small, only a small part of Q-bits is exchanged and the local search ability is not fully fulfilled. However, too large mutation rate will result in unstable searches. So an adaptive mutation strategy is adopted: in the initial stages, the mutation rate is set large to guide the algorithm searching in the whole space and keep solutions diversity; in the last stages, a small mutation rate is adopted to keep the searching process stable and efficient.

The pseudo codes of Algorithm 1 of the adaptive mutation operation are given.

In the pseudo codes, the *maxgen* means the maximum iterations, *gen* is the current iterated generation, N is the

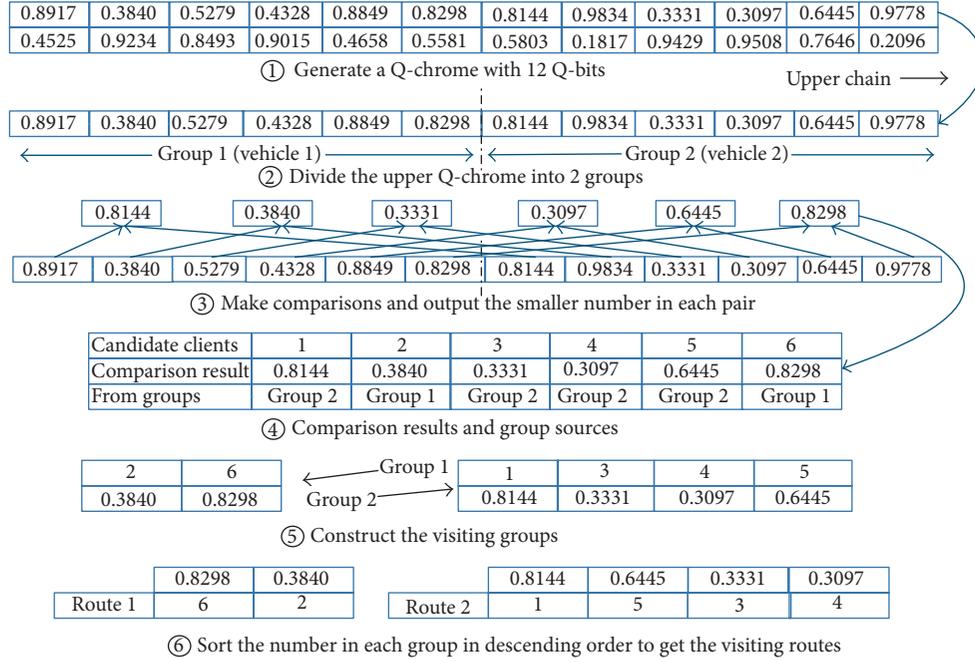


FIGURE 4: Vehicle routes construct process.

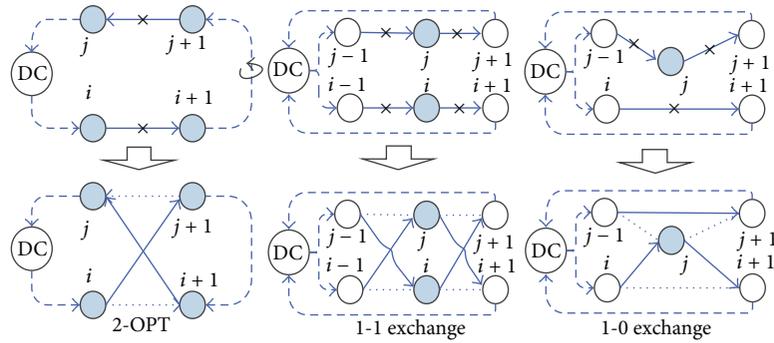


FIGURE 5: Three local search procedures.

- (1) Settings: $P_m = 1 - gen/maxgen$, $N = \text{ceil}(P_m \times \text{popsize})$;
- (2) **if** $N > \text{popsize}$ **then**
- (3) $N = \text{popsize}$;
- (4) **end if**
- (5) Randomly generate the positions N_{pos} of N Q-chroms, $N_{\text{pos}} \in [1, \text{popsize}]$;
- (6) Randomly generate K different intergers in each group, $K \in \text{Mutpo}$;
- (7) **if** $N \neq 0$ **then**
- (8) **for** $i = 1$ to N **do**
- (9) **for** $k = 1$ to K **do**
- (10) Use the quantum Not-gate mutating the specific Q-bit;
- (11) **end for**
- (12) **end for**
- (13) **end if**

ALGORITHM 1: Pseudo codes of adaptive mutation operation.

TABLE 4: Distance matrix and the demand quantities of each client.

Client	0	1	2	3	4	5	6	7	8	Demand
0	0	4	6	7.5	9	20	10	16	8	—
1	4	0	6.5	4	10	5	7.5	11	10	1
2	6	6.5	0	7.5	10	10	7.5	7.5	7.5	2
3	7.5	4	7.5	0	10	5	9	9	15	1
4	9	10	10	10	0	10	7.5	7.5	10	2
5	20	5	10	5	10	0	7	9	7.5	1
6	10	7.5	7.5	9	7.5	7	0	7	10	4
7	16	11	7.5	9	7.5	9	7	0	10	2
8	8	10	7.5	15	10	7.5	10	10	0	2

number mutated, and N_{pos} is the position of N Q-chrom. K is the number of vehicles, which means each group will select one Q-bit mutated. So, as iteration goes on, the P_m is getting smaller and smaller, but the mutation position will not be fixed, which can help keep the searching towards the best solution in one aspect and maintain population diversities.

3.5. Local Search Algorithm. Mutation operation can provide diversities to the quantum population, but we are quite not sure that the solutions, have been found are the global optimized solutions. That is to say, the proposed IQEA still has a risk of being trapped in local minimum solutions. Therefore, some local search operators should be employed to increase algorithm convergence and qualities of solutions. To improve the local search abilities and jump out the local minimum points, many local search procedures and operators are proposed in kinds of evolution algorithms, such as k -exchange [39] and λ -exchange neighborhood [40] and ICROSS, IRP, IOPT, and O-OPT [41]. In order to improve the local optimal both for a single route and multiple routes, Ai and Kachitvichyanukul [7] used three local search procedures: 2-OPT, exchange 1-1, and 1-0 exchange. But they were just operated separately because the combination mechanism of the three procedures was not set up. So a new local search mechanism that combines procedures of 2-OPT, exchange 1-1, and 1-0 exchange together is proposed.

3.5.1. Three Local Search Procedures. As depicted in the work of Ai and Kachitvichyanukul [7], 2-OPT is used to improve a single route that is determined by IQEA procedures by exchanging two different clients' visiting sequence in that route: if distance of the new route is smaller than the former route, the new one is accepted. Continue the procedure until all the clients are all checked and a local optimal is obtained (see Figure 5: 2-OPT).

The next two procedures 1-1 exchange and 1-0 exchange are adopted to deal with two adjacent candidate routes that have not reached their local optimum. 1-1 exchange is tried to interchange one client from the first route with another client in the other route (see Figure 5: 1-1 exchange) when the distance of the two clients is lower than a predetermined number according to a certain condition. If the total distance of the modified routes is better than that of the former ones, the bad routes will be replaced by new ones.

While in 1-0 exchange, a client is removed from the first route and inserted into the second route. If the total distance of new routes is lower than that of the old routes, the routes will be updated till all the clients in the first route are checked. The basic principal of 1-0 exchange is shown in Figure 5: 1-0 exchange.

More detailed information on the three local search procedures is depicted in Appendix B.

3.5.2. Design Local Search Algorithm. Before the design of the scheme of local searches, we should learn that there is a basic principle to follow, which is also our aim to do local searches: to convert an unfeasible route into a feasible route, and improve the feasible route to a local optimal route. Hence, we presented three local search schemes using the three above operators (there still are other schemes, the explanations on why we did not adopt them will be given later), and they are listed in Table 3.

There are some characters about three local search operators: (1) 1-1 exchange procedure deals with two client vertexes corresponding to two routes, without impacting the sequence of the two routes; (2) 1-0 exchange procedure is to extract one vertex from one route and insert it into another route, while the relative positions of remaining vertexes in two candidate routes are without changes (which still need to be visited by corresponding vehicles); (3) 2-OPT procedure only arranges the vertexes sequence of one candidate route and has nothing to do with other routes. Moreover, if we execute the 2-OPT firstly and then do the other two procedures, we will face the problem that, once the local search operation finished, the improved routes we got may still not be a local optimum; it still needs another 2-OPT operation to find its local optimal. Therefore, the 2-OPT procedure must be put in the end of the execution sequence.

How to arrange 1-0 exchange and 1-1 exchange procedures is another problem. From a problem-oriented point of view, the two procedures are all used to deal with two routes. On their processing objects, there exist 3 cases: both of the two routes are unfeasible, one is feasible and the other one is unfeasible, and both of the two routes are feasible. According to the local search principle stated above, an unfeasible route is improved to a feasible route. In dealing the routes are all unfeasible ones, the 1-0 exchange operator which is not completed for a vertex that moved from one

TABLE 5: Comparison of different algorithms.

Orders	DGA	PSO-SA	IQEA	IQEA + 2-OPT	IQEA + 1-1 exchange	IQEA + 1-0 exchange	IQEA+
1	69.0	67.5	67.5	69	67.5	69.5	67.5
2	70.5	67.5	69.0	69	67.5	69	67.5
3	67.5	67.5	67.5	67.5	69	69	67.5
4	70.5	67.5	67.5	67.5	67.5	67.5	67.5
5	69.0	67.5	70.5	69	69.5	67.5	67.5
6	71.0	67.5	69.0	67.5	69	67.5	67.5
7	69.5	67.5	69.5	69	67.5	69.5	67.5
8	70.0	67.5	67.5	67.5	69.5	67.5	67.5
9	67.5	67.5	67.5	67.5	69	67.5	67.5
10	70.0	67.5	67.5	69	67.5	67.5	67.5
AVG	69.45	67.5	68.3	68.25	68.35	68.20	67.5

route to another may not get two feasible routes, while the 1-1 exchange operator has more chances to improve the situation. In the case with of feasible route and one unfeasible route, 1-0 exchange can generate two feasible routes by removing a vertex from the unfeasible route and inserting it into the other route. In the third situation, both of the operators can be used to get better routes.

The pseudo codes of local search Algorithm 2 are as follows.

The local search procedure should be performed considering the coordination of the searching efficiency and searching effectiveness, so Algorithm 2 is conducted after the QEA searching has been finished. In order to separate the former IQEA, we named the IQEA with local search scheme 3 the IQEA+ (Figure 2).

4. Experiments and Discussions

The experiment is firstly conducted with a simple small case. For large-scale VRPs, parts of the standard benchmark are based on the work of Christofides et al. [42] and part of data sets are obtained from the website: <http://branchandcut.org/VRP/data/>. All the experiments were carried out in Windows system using Matlab 7.1 and performed on an Intel Pentium Dual-Core at 3.2 GHz and 2 GB RAM computer.

4.1. A Small CVRP Case and Parameters Analysis

4.1.1. A Contrastive Experiment: Small Case of CVRP. To illustrate the algorithm clearly, a small case of CVRP was introduced from Zhang et al. [18] and Chen et al. [43]. The content of the small case is as follows: 8 clients are served by 2 vehicles, the load capacity of which is 8 ton and the depot is node 0, and the optimal value of objective function is 67.5. The distance of every pair of clients and the demand of each client are shown in Table 4. All algorithms were run 10 times, and the max iteration number is 100. For IQEA and IQEA+, the population size was set to 30 and the mutation probability was set to 0.025π . From the results in Table 5, we can see that the IQEA+ has better performance than IQEA.

4.1.2. Parameters Tuning Test for IQEA and IQEA+. Before further test of the algorithm on large scale CVRPs, we should first investigate the parameters of the IQEA+ such as population size, rotation angles, and mutation probabilities to find their impacts on the performances (Table 11).

(1) Population Size Test. Experiments were performed on the above problem in Section 4.1.1 with the population size of 20, 30, and 50, respectively. Each case was run 30 times, the iteration step was set to 500, and the average of steps and the average of running time of IQEA+ that reached the best answer were given in Table 6. We can see that, as the population size increases, the running time cost increases linearly, especially when the population size goes to 30, and both of the running steps and running time reach a stable state.

(2) The Impacts of Rotation Angles and Mutation Probabilities. Each case was run 30 times, while the population size was set to 30. The average running time in finding the best solutions are listed in Tables 7 and 8 under different rotation angles and mutation probabilities, respectively. Table 7 shows that the IQEA and IQEA+ can find optimal solution faster when the rotation angle equals 0.01π and 0.025π , while our proposed rotation angle scheme can find the best solutions faster than any other approaches.

Table 8 shows the average computation times under different mutation probabilities. When the mutation probability is smaller than the 0.01, IQEA+ has spent too much computation time and is very hard to find the optimal solutions. We can find that the mutation probability is very important for IQEA+. It is partly because the client vertex in the candidate route is selected out by comparing between different Q-bits, and it will very easily fall into local minimum points with a small mutation probability. However, if the mutation probability was set too large (see the mutation probability = 0.5), IQEA+ cannot find the optimal solution for the unstable searching which makes the algorithm jump out of the spaces where the best solution may lie.

Still, there is a need to ascertain how to arrange three local search procedures to test their best performance, so we

```

(1) for  $i = 1$  to  $K$  do
(2)   Randomly select 2 candidate routes;
(3)   Calculate the actual carrying capacity (AC) of each vehicle.
(4)   if  $AC < \text{Burden Rating (BR)}$  then
(5)     Run 1-1 exchange and 1-0 exchange 5 times.
(6)     if solutions improved and constraints meet then
(7)       Run 2-OPT 5 times
(8)     else
(9)       Roll back to other routes
(10)    end if
(11)   else
(12)     Run 1-1 exchange and 1-0 exchange 5 times.
(13)     while solutions improved do
(14)       Run 2-OPT 5 times
(15)     end while
(16)   end if
(17) end for

```

ALGORITHM 2: Pseudo codes of local search algorithm.

select the $A-n32-k5.vrp$, the optimal value of which is 784 to test their performance. Each case was run 5 times, and the comparison results were shown in Table 9, where A is 2-OPT, B is 1-1 exchange and C is 1-0 exchange. We can easily find that the $C-B-C-A$ scheme has the best result, so this scheme was used to test the CVRP data sets.

4.2. Large Scale CVRP Benchmarks Tests and Analysis

4.2.1. Parameters Setting Analysis

(1) *Experiments of Medium Scale CVRPs.* In order to make IQEA+ suitable for solving large scale CVRP benchmarks, we firstly test it by solving some medium scale cases. The same parameters settings as mentioned in Section 4.1 are still adopted. Each case is run 20 times to obtain the average value. The results are shown in Table 10.

Results of Table 10 show that the parameters setting of IQEA+ mentioned in Section 4.1 is relatively reasonable.

(2) *Parameter Settings for IQEA+ for Large Scale CVRPs.* In fact, related parameters of the most intelligent algorithms are confirmed according to the specific problems. Finding a universal parameter setting for intelligence algorithms is rather difficult. As discussed in Section 4.1, the parameters of IQEA+ are also confirmed by parameters setting test and/or experiences of similar problems.

Related parameters of IQEA+ for solving larger-scale CVRP should be also discussed. To simplify our analysis, we only retest two large scale cases “ $Bn78k10$ ” and “ $Pn101k$ ”. Also, these two cases with different number of vehicles can also show some performance information on local searching procedures. Those experiments are tested with ten runs and fixed 500 iterations each without any break intervention. The computation results show us the following.

- (a) Three important parameters of IQEA+, population size, rotation angle, and mutation probability, can cause great impacts of IQEA+'s performance.

- (b) The main influences of population size lies in the searching time but not in the quality of the searching results. Different rotation angles and mutation probabilities have great influences of IQEA+ on finding the optimal values, but the searching times in different settings are quite approximative to each other.
- (c) A large mutation probability seems to have better performance than that with a smaller mutation probability.
- (d) The longer the length of a single patrolling route, the worse the searching ability of the local search procedure. In “ $Bn78k10$ ” and “ $Pn101k$ ” analyses, the average lengths of the patrolling routes of the former case and the latter case are 7.8 and 25.3, respectively. So, the former case has better optimal results.

Finding a universal parameter setting of IQEA+ is rather challengeable and costly under several types of CVRP benchmarks. Therefore, based on results of IQEA+ for solving the medium scale CVRPs and two large scale CVRP cases and the experiences of [19, 22], the similar parameters' setting of IQEA+ in Section 4.1 is also be applied to solve large scale CVRPs.

4.2.2. *Experiments for Large Scale CVRP Benchmarks.* Large scale VRP benchmarks are also obtained from the website (<http://branchandcut.org/VRP/data/>), which were also introduced by Chen et al. [43] and Ai and Kachitvichyanukul [7]. There are sixteen CVRP cases, and the total number of clients is varying from 29 to 134, and the vehicle number is varying from 3 to 10. The settings of the IQEA+ for large scale CVRPs are as follows: the iteration is 500 times and the population size is 30. The setting of the rotation angle and mutation probability are the same as in Section 4.1. The searching performance of IQEA+ is compared with that from Chen's et al. [43] DPSO algorithm and Ai and Kachitvichyanukul's

TABLE 6: Outputs in finding the best solutions under different population size.

Pop size	IQEA		IQEA + 2-OPT		IQEA + 1-1 exchange		IQEA + 1-0 exchange		IQEA+	
	Time (s)	Steps	Time (s)	Steps	Time (s)	Steps	Time (s)	Steps	Time (s)	Steps
20	0.72	113	1.69	271	167	263	1.42	220	1.39	209
30	0.95	102	2.09	213	1.89	194	1.52	156	2.26	234
50	2.12	140	2.61	170	3.42	226	2.20	138	3.59	232

TABLE 7: CPU times (s) under different rotation angles.

Rotation angle	IQEA	IQEA+
0.005 pi [26]	2.06	2.18
0.01 pi [26]	1.88	1.99
0.025 pi [26]	1.82	1.92
0.05 pi [26]	1.95	2.07
Std/5	2.04	2.16
Std/10	1.92	2.03
Std/(10 * selfsgn)	1.38	1.49

TABLE 8: Optimal value under different mutation probabilities.

Probabilities	IQEA	IQEA+
0.01 [26]	71.8	70.0
0.025 [26]	70.7	68.9
0.05 [26]	72.3	70.2
0.1 [26]	69.5	68.4
0.2	69.1	68.4
0.5	72.3	70.9
Self-adapted P	67.8	67.5

TABLE 9: Different local search schemes for IQEA+.

No.	Scheme	Best objective	AVE	AVE.CPU time (s)
1	$A \rightarrow B \rightarrow C$	798	821.6	23.56
2	$B \rightarrow A \rightarrow C$	806	842	20.59
3	$C \rightarrow A \rightarrow B \rightarrow C$	784	828.8	21.61
4	$C \rightarrow B \rightarrow A \rightarrow C$	789	826	18.51
5	$C \rightarrow B \rightarrow C \rightarrow A$	784	791.6	15.74

[7] SR-2 algorithm; all the comparison results are shown in Table 12.

4.3. Discussions. In order to test the proposed IQEA+, firstly, it has been used to solve a simple CVRP to find different performances of the IQEA and IQEA+; then we test different body parts of IQEA to best play out its advantages. Three local search procedures combination schemes were also examined, and all of them were used to test the CVRP dataset.

Experiments results show that the proposed IQEA+ has potential in solving the CVRP to some extent with relatively high precision and speed; see Table 10 and several cases in Table 12. However, experiments also show some useful findings.

- (1) IQEA+ outperforms DPSO, SR-1, and SR-2 in searching precision but not in the searching efficiency; see case *Pn101k4* in Table 12;

- (2) Sometimes, IQEA+ does not show many advantages on searching performance; see case *Fn135k7* in Table 12.

- (3) IQEA can search quickly, while the local search procedures take more than a half of the total computing time, especially when problem scale becomes larger.

Three main reasons can explain our findings. (a) There is a weakness of the IQEA+ in the length of quantum chromosome which originated from the encoding mechanism. When the client vertexes and vehicle number increase, especially as the length of quantum chromosome reached to thousand level, algorithm searching efficiency becomes very low. (b) The tightness (demand/capacity) of demand and capacity is another important factor that impacts the searching performance; see case *Fn72k4* in Table 12. Different tightness may result in very different results: the more intensive are the tightness and the worse is the results, and the more intensive are the tightness, the greater chances of unfeasible routes are generated out. (c) Local search procedures are used to deal with the point-to-point operation and evaluation, which make it cost too much time.

5. Conclusions and Future Research

Although the CVRP has been presented for tens of years, still much attention has been put on it. Due to its NP nature, a lot of heuristic and metaheuristic algorithms have been proposed to provide better solutions for this problem. This paper proposes a new IQEA with the mixed local search procedure (named IQEA+) to solve the CVRPs: IQEA is the improvements of original QEA in the encoding mechanism for solving the CVRP. Moreover, in order to make full use of the information embedded in quantum chromosome, a double chain quantum chromosome is adopted. Local search procedures are to enhance the IQEA local search ability. The computational results show that the proposed IQEA+ is more effective in solving the CVRP than the DPSO and also has some advantages over other widely used SR-1 and SR-2 when the length of chromosome is within a certain range.

There are some obvious advantages of the proposed IQEA+:

- (1) less parameters to be set make the proposed IQEA+ very simple and easy to be implemented;
- (2) unlike the classic genetic algorithm, the updating and mutation operations were replaced by easier quantum rotation gate and the quantum Not gate procedures;

TABLE 10: Results of some medium-scale CVRP cases by IQEA+.

No.	Type	Problem instance			Object function		CPU time (s)
		Clients	Vehicles	Optimal	Best	AVE	
1	A-n32-k5	31	5	784	784	791.6	15.7
2	A-n34-k5	33	5	778	780	794.4	17.4
3	B-n31-k5	30	5	672	672	679.8	11.9
4	B-n34-k5	33	5	788	788	798.8	16.4
5	P-n19-k2	18	2	212	212	213.6	4.8
6	P-n20-k2	19	2	216	216	217.4	5.2
7	P-n40-k5	39	5	458	458	469.6	13.1
8	E-n23-k3	22	3	569	574	575.4	8.0
9	E-n22-k4	21	4	375	375	387.6	9.5
10	F-n45-k4	44	4	724	730	763.8	20.2

TABLE 11: The impacts of the parameters of IQEA+ under two large cases.

	Pn78k10			Pn101k4		
	Best value	AVE	AVE of CPU time	Best value	AVE	AVE of CPU time
Pop size						
20	1507	1651	78.5	852	1303	85
30	1256	1295	72.2	788	1322	136
50	1358	1704	192	984	1306	219
Rotation angle						
0.005 pi	1574	1628	233	1252	1337	133
0.01 pi	1562	1618	244	1145	1221	138
0.025 pi	1607	1582	250	1076	1123	136
0.05 pi	1469	1610	234	1021	1068	140
Std/5	1602	1557	181	904	937	137
Std/10	1487	1615	136	897	955	144
$\Delta 1$	1321	1465	132	885	992	136
Mutation probability						
0.01	2016	2048	130	1138	1230	143
0.025	1970	2066	133	1062	1217	144
0.05	1885	1918	144	1007	1193	137
0.1	1775	1848	137	913	1097	148
0.2	1539	1679	146	1282	1326	140
0.5	1493	1523	138	1223	1302	141
$\Delta 2$	1385	1505	134	895	1202	141

$\Delta 1 = \text{Std}/(10 * \text{selfsgn})$, $\Delta 2 = \text{Self-adapted P}$.

(3) the comprehensive improvement of QEA with the double chain, new rotation angle, and rotation direction and self-adaptive Not gate can simplify its searching procedure and improve its efficiency. Moreover, three local search procedures help the IQEA+ jump out of local minimum points and continue its searching toward the right path.

Future work can be focused on finding a new encoding mechanism to further increase the efficiency and keep the diversity of the quantum population. There still are improvement spaces in the IQEA+ framework, such as the significance of the rotation angle is designed as the problem-oriented and the mutation procedure is set as the fitness

function adapted. Moreover, the proposed IQEA+ can be used to solve some VRP variants.

Appendices

A. Proof of Theorem 1

Suppose that there are 4 unit circles, E_1, E_2, ϵ_1 , and ϵ_2 which can conveniently be expressed out and the entire proving process will be illustrated in them. As we all know, the tangent function is increasing in its defined intervals $[-\pi/2, \pi/2]$ and because all the numbers are should lie in $(0, 1)$, one has the following.

TABLE 12: Results of large-scale CVRPs with different algorithms.

Problem	Clients	Vehicles	Optimal value	Best objective function				AVE. IQEA+	CPU time (s)			
				Chen	SR-1	SR-2	IQEA+		Chen	SR-1	SR-2	IQEA+
An33k5	32	5	661	661	661	661	661	669.2	32	11	13	15.7
An46k7	45	7	914	914	914	914	914	970.6	129	19	23	72.0
An60k9	59	9	1354	1354	1366	1355	1363	1410.4	309	28	40	148.1
Bn35k5	34	5	955	955	955	955	955	967.6	38	12	14	21.2
Bn45k5	44	5	751	751	751	751	751	789.4	134	17	20	50.3
Bn68k9	67	9	1272	1272	1278	1274	1281	1349.8	344	33	50	65.4
Bn78k10	77	10	1221	1239	1239	1223	1256	1295.4	429	41	64	72.2
En30k3	29	3	534	534	541	534	534	551.2	28	11	16	13.4
En51k5	50	5	521	528	521	521	524	566	301	32	22	20.5
En76k7	75	7	682	688	691	682	703	739.4	527	38	60	150.0
Fn72k4	71	4	237	244	237	237	237	293.4	398	58	53	64.5
Fn135k7	134	7	1162	1215	1184	1162	1202	1414.2	1526	178	258	159.2
Mn101k10	100	10	820	824	821	820	956	1001.6	874	60	114	210.4
Mn121k7	120	7	1034	1038	1041	1036	1082	1207.4	1734	88	89	128.5
Pn76k4	75	4	593	602	599	594	594	612	496	51	48	64.9
Pn101k4	100	4	681	694	686	683	681	689.4	978	99	86	135.0

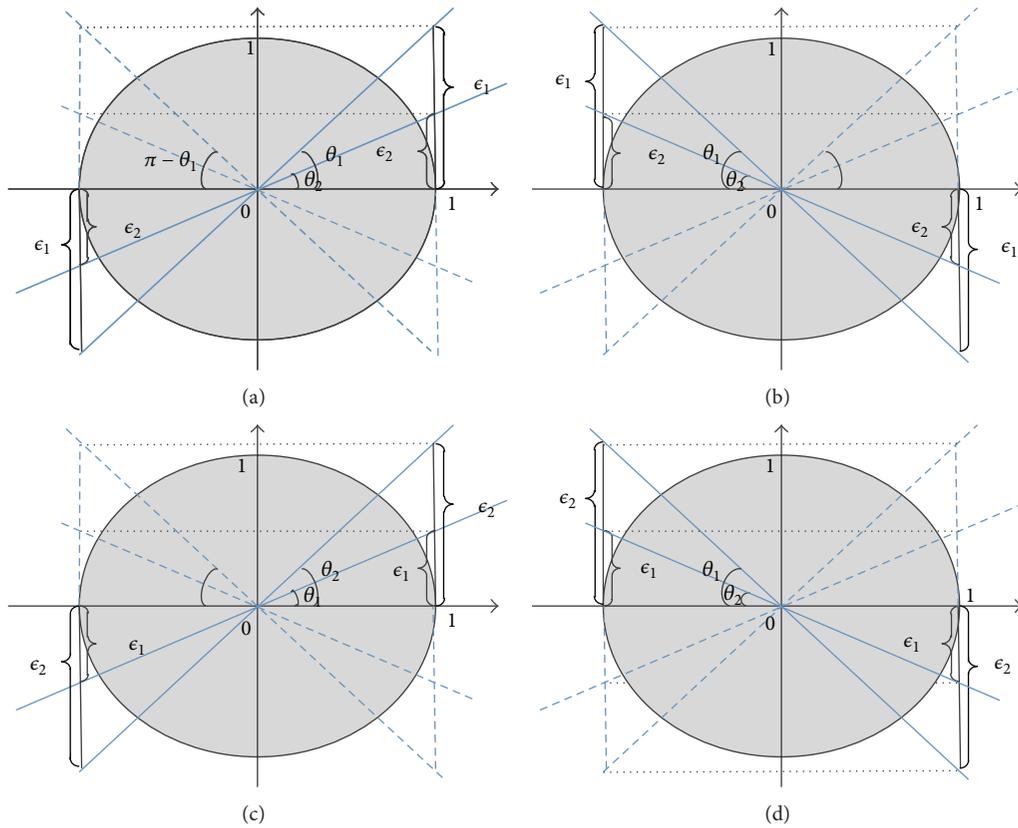
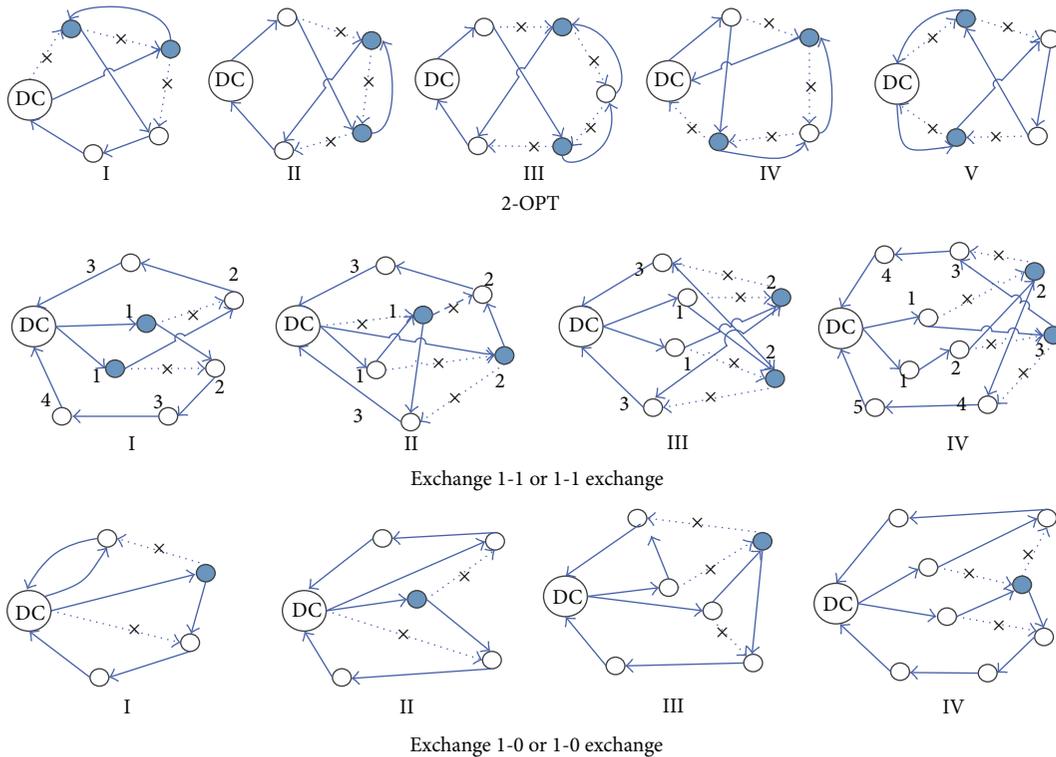


FIGURE 6: Illustrations of different rotation strategies.

(I) when $\epsilon_1 \geq \epsilon_2$, a further discussion is given as follows.

- (1) If $E_2 > 0$, θ_1 may lie in any quadrants, while θ_2 may in the first quadrant or the third quadrant. It is obvious that when θ_1 and θ_2 are all in the

first and the third quadrants, see Figure 6(a) (the solid line), the current angle needs to be added a positive number to catch up the best angle. When θ_1 is in the other two quadrants, see Figure 6(a) (the dash line), because we only care



The blue solid arrow denoted the real route, the blue dotted arrow denoted previous route, and the black cross means “delete”.

FIGURE 7: Illustrations of 3 local searches procedures.

about the absolute value of the rotation angle, θ_2 also needs to be added a positive number to catch up the best angle (the absolute value of θ_1).

- (2) If $E_2 < 0$, still θ_1 may lie in any quadrants, while θ_2 may lie in the second quadrant or the fourth quadrant. So when θ_1 and θ_2 are both in the second quadrant and the fourth quadrant, see Figure 6(b) (the solid line), the current angle needs to be subtracted a positive number to catch the best angle. When θ_1 is in the other two quadrants, see Figure 6(b) (the dash line), θ_2 also needs to be subtracted a positive number to catch up the best angle (the absolute value of θ_1).

So Theorem 1(1) has been proved.

- (II) When $\epsilon_1 < \epsilon_2$, the proven processes are very similar to those in Appendix A(I).

- (i) If $E_2 > 0$, the current angle needs to be subtracted a positive number to catch up the best angle; see Figure 6(c) (the solid line and the dash line).
- (ii) If $E_2 < 0$, the current angle needs to be added a positive number to catch up the best angle; see Figure 6(d) (the solid line and the dash line).

So Theorem 1(2) has been proved.
Now, Theorem 1 has been proved.

B. Detailed Operations of Three Local Search Procedures

The detailed local search procedures are shown in Figure 7. In Figure 7, we only try to show the sketches of three local search procedures.

Acknowledgments

This research is partially supported by the National Natural Science Foundation of China (nos. 70801030, 71131004, and 71371080), Humanities and Social Sciences Foundation of the Chinese Ministry of Education (no. 11YJC630275), and the Fundamental Research Funds for the Central Universities (HUST: 2012TS065; HUST: CXY12Q043; and HUST: CXY13Q034).

References

- [1] G. B. Dantzig and J. H. Ramser, “The truck dispatching problem,” *Management Science*, vol. 6, no. 1, pp. 80–91, 1959.
- [2] J. G. Carlsson, “Dividing a territory among several vehicles,” *INFORMS Journal on Computing*, vol. 24, no. 4, pp. 565–577, 2012.

- [3] J. G. Carlsson and E. Delage, "Robust partitioning for stochastic multivehicle routing," *Operations Research*, vol. 61, no. 3, pp. 727–744, 2013.
- [4] G. Laporte, "The vehicle routing problem: an overview of exact and approximate algorithms," *European Journal of Operational Research*, vol. 59, no. 3, pp. 345–358, 1992.
- [5] B. F. Moghaddam, R. García, and S. J. Sadjadi, "Vehicle routing problem with uncertain demands: an advanced particle swarm algorithm," *Computers and Industrial Engineering*, vol. 62, no. 1, pp. 306–317, 2012.
- [6] J. K. Lenstra and A. H. G. Kan, "Complexity of vehicle routing and scheduling problems," *Networks*, vol. 11, no. 2, pp. 221–227, 1981.
- [7] T. J. Ai and V. Kachitvichyanukul, "A particle swarm optimization for the vehicle routing problem with simultaneous pickup and delivery," *Computers and Operations Research*, vol. 36, no. 5, pp. 1693–1702, 2009.
- [8] G. Laporte, "Fifty years of vehicle routing," *Transportation Science*, vol. 43, no. 4, pp. 408–416, 2009.
- [9] B. L. Golden, S. Raghavan, and E. A. Wasil, *The Vehicle Routing Problem: Latest Advances and New Challenges*, vol. 43 of *Operations Research/Computer Science Interfaces Series*, Springer, New York, NY, USA, 2008.
- [10] C. H. Vera, F. D. Karl, and F. H. Richard, "A variable neighborhood search heuristic for periodic routing problems," *European Journal of Operational Research*, vol. 195, no. 3, pp. 791–802, 2009.
- [11] B. M. Baker and M. Ayechev, "A genetic algorithm for the vehicle routing problem," *Computers and Operations Research*, vol. 30, no. 5, pp. 787–800, 2003.
- [12] C. Rego and B. Alidaee, *Metaheuristic Optimization via Memory and Evolution*, vol. 30 of *Operations Research/Computer Science Interfaces Series*, Kluwer Academic, Boston, Mass, USA, 2005.
- [13] Y. Marinakis, M. Marinaki, and G. Dounias, "A hybrid particle swarm optimization algorithm for the vehicle routing problem," *Engineering Applications of Artificial Intelligence*, vol. 23, no. 4, pp. 463–472, 2010.
- [14] W. Y. Szeto, Y. Z. Wu, and C. H. Sin, "An artificial bee colony algorithm for the capacitated vehicle routing problem," *European Journal of Operational Research*, vol. 215, no. 1, pp. 126–135, 2011.
- [15] A. Narayanan and M. Moore, "Quantum-inspired genetic algorithms," in *Proceedings of the IEEE International Conference on Evolutionary Computation*, pp. 61–66, Nagoya, Japan, May 1996.
- [16] Z. Luo, P. Wang, Y. G. Li, W. F. Zhang, W. Tang, and M. Xiang, "Quantum-inspired evolutionary tuning of SVM parameters," *Progress in Natural Science*, vol. 18, no. 4, pp. 475–480, 2008.
- [17] A. D. S. Nicolau, R. Schirru, and A. M. M. de Lima, "Nuclear reactor reload using quantum inspired algorithm," *Progress in Nuclear Energy*, vol. 55, pp. 40–48, 2012.
- [18] J. L. Zhang, Y. W. Zhao, D. J. Peng, and W. L. Wang, "A hybrid quantum-inspired evolutionary algorithm for capacitated vehicle routing problem," in *Advanced Intelligent Computing Theories and Applications: With Aspects of Theoretical and Methodological Issues*, vol. 5226 of *Lecture Notes in Computer Science*, pp. 31–38, 2008.
- [19] J. Zhang, W. L. Wang, Y. W. Zhao, and C. Cattani, "Multiobjective quantum evolutionary algorithm for the vehicle routing problem with customer satisfaction," *Mathematical Problem in Engineering*, vol. 2012, Article ID 879614, 19 pages, 2012.
- [20] R. Zhang and H. Gao, "Real-coded quantum evolutionary algorithm for complex functions with high-dimension," in *Proceedings of the IEEE International Conference on Mechatronics and Automation (ICMA '07)*, pp. 2974–2979, Luoyang, China, August 2007.
- [21] P. L. Li and S. Y. Li, "Quantum-inspired evolutionary algorithm for continuous space optimization," *Chinese Journal of Electronics*, vol. 17, no. 1, pp. 80–84, 2008.
- [22] F. J. Hu and B. Wu, "Quantum evolutionary algorithm for vehicle routing problem with simultaneous delivery and pickup," in *Proceedings of the 48th IEEE Conference on Decision and Control*, pp. 5097–5101, Shanghai, China, December 2009.
- [23] J. Gao and J. Wang, "A hybrid quantum-inspired immune algorithm for multiobjective optimization," *Applied Mathematics and Computation*, vol. 217, no. 9, pp. 4754–4770, 2011.
- [24] M. Chen, Z. Ouyang, and H. Quan, "Evolutionary algorithm based on evaluating quantum chromosomes," *Computer Engineering and Applications*, vol. 43, no. 21, pp. 84–86, 2007.
- [25] J. C. Lee, W. M. Lin, G.-C. Liao, and T. P. Tsao, "Quantum genetic algorithm for dynamic economic dispatch with valve-point effects and including wind power system," *International Journal of Electrical Power and Energy Systems*, vol. 33, no. 2, pp. 189–197, 2011.
- [26] K. Han, K. Park, C. Lee, and J. Kim, "Parallel quantum-inspired genetic algorithm for combinatorial optimization problem," in *Proceedings of the Congress on Evolutionary Computation*, vol. 2, pp. 1422–1429, Seoul, Republic of Korea, May 2001.
- [27] T. C. Lu and J. C. Juang, "Quantum-inspired space search algorithm (QSSA) for global numerical optimization," *Applied Mathematics and Computation*, vol. 218, no. 6, pp. 2516–2532, 2011.
- [28] G. Benenti, G. Casati, and G. Strini, *Principles of Quantum Computation and Information: Basic Tools and Special Topics*, World Scientific, River Edge, NJ, USA, 2007.
- [29] Z. Li, R. Gunter, and K. Li, "Convergence performance comparison of quantum-inspired multi-objective evolutionary algorithms," *Computers and Mathematics with Applications*, vol. 57, no. 11, pp. 1843–1854, 2009.
- [30] M. Ying, "Quantum computation, quantum theory and AI," *Artificial Intelligence*, vol. 174, no. 2, pp. 162–176, 2010.
- [31] T. Zheng and M. Yamashiro, "Solving flow shop scheduling problems by quantum differential evolutionary algorithm," *The International Journal of Advanced Manufacturing Technology*, vol. 49, no. 5–8, pp. 643–662, 2010.
- [32] H. Su, Y. Yang, and L. Zhao, "Classification rule discovery with DE/QDE algorithm," *Expert Systems with Applications*, vol. 37, no. 2, pp. 1216–1222, 2010.
- [33] H. Su and Y. Yang, "Differential evolution and quantum-inquired differential evolution for evolving Takagi-Sugeno fuzzy models," *Expert Systems with Applications*, vol. 38, no. 6, pp. 6447–6451, 2011.
- [34] J. Sun, B. Feng, and W. Xu, "Particle swarm optimization with particles having quantum behavior," in *Proceedings of the Congress on Evolutionary Computation (CEC '04)*, vol. 1, pp. 325–331, Portland, Ore, USA, June 2004.
- [35] S. N. Omkar, R. Khandelwal, T. V. S. Ananth, G. N. Naik, and S. Gopalakrishnan, "Quantum behaved particle swarm optimization (QPSO) for multi-objective design optimization of composite structures," *Expert Systems with Applications*, vol. 36, no. 8, pp. 11312–11322, 2009.

- [36] P. L. Li and S. Y. Li, "Quantum-inspired evolutionary algorithm for continuous space optimization based on Bloch coordinates of qubits," *Neurocomputing*, vol. 72, no. 1-3, pp. 581-591, 2008.
- [37] J. Gu, X. Gu, and M. Gu, "A novel parallel quantum genetic algorithm for stochastic job shop scheduling," *Journal of Mathematical Analysis and Applications*, vol. 355, no. 1, pp. 63-81, 2009.
- [38] J. Gu, M. Gu, C. Cao, and X. Gu, "A novel competitive co-evolutionary quantum genetic algorithm for stochastic job shop scheduling problem," *Computers and Operations Research*, vol. 37, no. 5, pp. 927-937, 2010.
- [39] Y. Li, P. Pardalos, and M. Resende, "A greedy randomized adaptive search procedure for the quadratic assignment problem," in *Quadratic Assignment and Related Problems*, vol. 16 of *DIMACS Series on Discrete Mathematics and Theoretical Computer Science*, pp. 237-261, American Mathematical Society, Providence, RI, USA, 1994.
- [40] W. C. Chiang, "The application of a tabu search metaheuristic to the assembly line balancing problem," *Annals of Operations Research*, vol. 77, pp. 209-227, 1998.
- [41] O. Bräysy, "A reactive variable neighborhood search for the vehicle-routing problem with time windows," *INFORMS Journal on Computing*, vol. 15, no. 4, pp. 347-368, 2003.
- [42] N. Christofides, A. Mingozzi, and P. Toth, "The vehicle routing problem," in *Combinatorial Optimization*, vol. 11, pp. 315-338, John Wiley and Sons, Chichester, UK, 1979.
- [43] A. Chen, G. Yang, and Z. Wu, "Hybrid discrete particle swarm optimization algorithm for capacitated vehicle routing problem," *Journal of Zhejiang University A*, vol. 7, no. 4, pp. 607-614, 2006.



Hindawi

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

