

An Improved Quantum Ant Colony Optimization Algorithm for Solving Complex Function Problems

Changai Chen and Yanwen Xu

*Department of Information Technology, Henan University of TCM, Zhengzhou,
Henan 450008 China*

Abstract

In order to improve the slow convergence speed and avoid falling into the local optimum in ant colony optimization algorithm, an improved quantum ant colony optimization (IMAQACO) algorithm based on combing quantum evolutionary algorithm with ant colony optimization algorithm is proposed for solving complex function problems in this paper. In the IMAQACO algorithm, the quantum state vectors are used to represent the pheromone, the adaptively dynamical updating strategy is used to control pheromone evaporation factor, the quantum rotation gate is used to realize the ant movement and change the convergence tend of quantum probability amplitude, quantum non-gate is used to realize ant location variation, so the IMAQACO algorithm has better global search ability and population diversity than ACO algorithm. In order to test the optimization performance of IMAQACO algorithm, several benchmark functions are selected in here. The tested results indicate that the IMAQACO can effectively improve the convergence speed and avoid falling into the local optimum, and has a stronger global optimization ability and higher convergence speed in solving complex function problems.

Keywords: *Quantum evolution algorithm; ant colony optimization algorithm; quantum rotation gate; quantum non-gate; adaptive control; complex function*

1. Introduction

Now the intelligent algorithms are more and more concerned [1-2], such as neural network (NN), genetic algorithm (GA), particle swarm optimization(PSO) algorithm, ant colony optimization (ACO) algorithm, Tabu search algorithm, simulated annealing (SA) algorithm, bee colony optimization (BCO) algorithm and so on. These intelligent algorithms can effectively solve combinatorial optimization and NP-Hard problems. But they have their own advantages and disadvantages in solving complex problems [3-4]. At present, it is a hot research that integrate or fuse various optimization algorithms by using their respective advantages in order to construct better performance algorithms.

Ant colony optimization (ACO) algorithm was introduced by Dorigo [5]. It is a branch of newly developed form of artificial intelligence, called swarm intelligence, which studies “the emergent collective intelligence of groups of simple agents”. The ACO algorithm is a metaheuristic inspired by the behavior of real ants in their search for the shortest path to food. The ACO algorithm simulates the techniques employed by real ants to rapidly establish the shortest route from a food source to their nest and vice versa without the use of visual information. The basic idea of the ACO algorithm is to model the problem that is being solved, just as the search for a minimum cost path in a graph that uses artificial ants to search for good paths. The positive feedback mechanism and communication mechanism are two important characteristics in the ACO algorithm. The ACO algorithm has strong robustness and better search ability. But it also has some defects, such as slow convergence speed, easy falling into the local optimum and stagnation phenomenon and so on. In order

to solve these existing defects, some scholars proposed a lot of improved ACO algorithm. Leng *et al.* [6] proposed an improved ant colony optimization (ACO) algorithm based on the dynamic pheromone updating for cell scheduling of the flexible manufacturing process for considering the instrument constraint, manufacturing cost and time. Jalali *et al.* [7] proposed an improved ant colony optimization (ACO) algorithm for reservoir operation. Zhang *et al.* [8] proposed an improved ant colony optimization. The proposed algorithm made use of the trail information which was deposited between the task and the task selected position, and pheromone summation rules was adopted. Kanyapat and Paramote [9] proposed a new efficiency (IACO) algorithm based on the conventional ant colony optimization (ACO) to solve the communication network design when considering both economics and reliability. Li and Tian [10] proposed an improved ant colony algorithm based on multi-population for the multicast routing QoS. Yi and Lai [11] proposed a new ant colony algorithm to deal with p//T (p//T-ACO), and designed a computing model of mapping p//T to the ant colony optimization environment. Yu *et al.* [12] proposed an improved ant colony optimization (IACO), which possesses a new strategy to update the increased pheromone, called ant-weight strategy, and a mutation operation, to solve VRP. Kaveh and Talatahari [13] proposed an improved ant colony optimization (IACO) for constrained engineering design problems. Yang and Zhuang [14] proposed an improved ant colony optimization algorithm (IACO) for solving mobile agent routing problem. The ants cooperate using an indirect form of communication mediated by pheromone trails of scent and find the best solution to their tasks guided by both information (exploitation) which has been acquired and search (exploration) of the new route. Gan *et al.* [15] proposed an improved ACO algorithm based on scout characteristic for solving the stagnation behavior and premature convergence problem of the basic ACO algorithm on TSP. The main idea is to partition artificial ants into two groups: scout ants and common ants. Yu *et al.* [16] proposed a parallel improved ant colony optimization algorithm for solving multi-depot vehicle routing problem (MDVRP). Geng *et al.* [17] proposed a directional ant colony optimization (DACO) algorithm for solving nonlinear resource-leveling problems. The DACO algorithm can efficiently improve the convergence rate and the quality of solution for real-project scheduling. Raka and Milan [18] proposed an improved ACO algorithm based on pheromone correction heuristic strategy for the minimum weight vertex cover problem. Ding *et al.* [19] proposed a hybrid ant colony optimization (HACO) for vehicle routing problem with time windows. Xu *et al.* [20] proposed an improved ACO algorithm based on chaotic map. Li *et al.* [21] proposed an improved ant colony optimization (ACO) algorithm. The dynamic scheduling model based on improved ACO was set up. Cheng *et al.* [22] proposed a novel ant colony optimization method where the Metropolis Criterion is used to select the paths of ants to overcome the immature convergence. Lalbakhsh *et al.* [23] proposed a novel pheromone update strategy to improve the functionality of ant colony optimization algorithms for traveling salesman problem and classification rule mining. Su *et al.* [24] proposed an improved ant colony algorithm (IMACO) for virtual teams building in collaborative process planning. Shen and Chen [25] proposed an improved ACO based on RWA algorithm. Furthermore, a novel wavelength converter allocation strategy is proposed based on the improved ACO algorithm. Chong *et al.* [26] proposed a hybrid of ant colony optimization (ACO) and minimization of metabolic adjustment (MoMA) to identify gene knockout strategies to improve the production of succinic acid in *E. coli*. Tang *et al.* [27] proposed an improved ant colony optimization (IACO) is proposed for solving multi-depot heterogeneous vehicle routing problem with soft time windows. Pang *et al.* [28] proposed an improved ant colony optimization algorithm based on the optimal search library, the new heuristic

information and the improved pheromone update rules for solving the traveling salesman problem. Jiang *et al.* [29] proposed a co-evolutionary improved multi-ant colony optimization (CIMACO) algorithm for ship multi and branch pipe route design. Cheng *et al.* [30] proposed an improved ant colony optimization method for the integrated scheduling problem of production and distribution for manufacturers.

Quantum evolutionary algorithm (QEA) was proposed by Han in 2002. This algorithm an evolutionary algorithm based on quantum theory. It absorbs the idea of superposition state, coherence and entanglement in quantum computing, which makes the quantum algorithm to break through the limit of the traditional algorithm, and show better performance. Compared with the classical evolutionary algorithm (EA), the QEA has many advantages, such as better population diversity and global optimization ability, smaller population, and so on. At present, the QEA with unique computational performance has become a hot research topic, and has been applied to many optimization problems, especially solve the NP problems. In order to improve the slow convergence speed and avoid falling into the local optimum in ant colony optimization algorithm, the QEA are introduced in to ant colony optimization algorithm in order to an improved quantum ant colony optimization (IMAQACO) algorithm in this paper. And several benchmark functions are selected to test the optimization performance of IMAQACO algorithm.

The rest of this paper is organized as follows. Section 2 briefly introduces the ACO algorithm. Section 3 proposed Quantum evolutionary algorithm (QEA). Section 4 presents an improved quantum ant colony optimization (IMAQACO) algorithm. In this section, the adaptive adjustment strategy of quantum rotation gate, the position update of ant, the position mutation of ant and the flow description of IMAQACO algorithm are introduced in detail. Section 5 is experiment results and analysis. Finally, the conclusions are discussed in Section 6.

2. ACO Algorithm

The ACO algorithm is a metaheuristic inspired by the behavior of real ants in their search for the shortest path to food. The ACO algorithm consists of a number of cycles (iterations) of solution construction. In each iteration, a number of ants construct complete solutions by using heuristic information and the collected experiences of previous groups of ants. These collected experiences are represented by the pheromone trail which is deposited on the constituent elements of a solution. Small quantities are deposited during the construction phase while larger amounts are deposited at the end of each iteration in proportion to solution quality. Pheromone can be deposited on the components and/or the connections used in a solution depending on the problem.

In the ACO algorithm, the ACO algorithm simulates the optimization of ant foraging behavior. The procedure of the ACO algorithm is described in Figure1.

Assume that there are n cities and m ants, at the same time assuming that the initial intensity of pheromone on each edge is set to a very small non-zero positive constant τ_0 . In each cycle, each ant starts at a stochastic chosen city, then visits the other cities once and only once according to the transition rule based on the initial intensity of pheromone. When the ants complete the routes of one cycle, the length of one cycle will be computed. Then, the intensity of pheromone will be updated by using the pheromone update rule. The procedure of pheromone update rule is shown as follows:

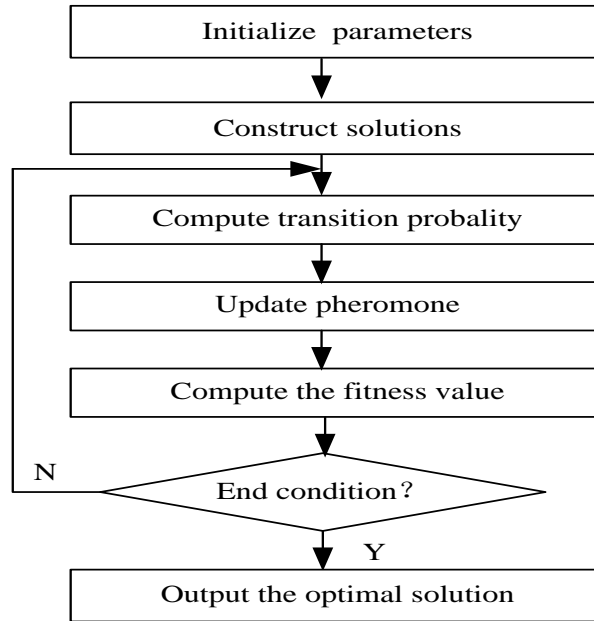


Figure 1. The Flow of the ACO Algorithm

(1) The transition rule

To visit the next city s with the probability $p_k(r, s)$,

$$p_k(r, s) = \begin{cases} \frac{\tau(r, s)^\alpha \cdot \eta(r, s)^\beta}{\sum_{u \in J_r^k} \tau(r, u)^\alpha \cdot \eta(r, u)^\beta} & \text{if } s \in J_r^k \\ 0 & \text{otherwise if } q > q_0 \text{ (Bias Exploitation)} \end{cases} \quad (1)$$

In two formula, $p_k(r, s)$ is the transition probability, $\tau(r, u)$ is the intensity of pheromone between city r and city u in the i^{th} population, $\eta(r, u)$ is the length of the path from city r to city u , J_r^k is the set of unvisited cities of the k^{th} ant in the i^{th} population, the parameter α and β are the control parameters, q is a uniform probability [0-1].

(2) The pheromone update rule

In order to improve the solution, the pheromone trails must be updated. Trail updating includes local updating and global updating. The local trail updating formula is given by:

$$\tau(r, u) = (1 - \rho)\tau(r, s) + \sum_{k=1}^m \Delta \tau_k(r, s) \quad (2)$$

In here, ρ ($0 < \rho < 1$) is the pheromone trail evaporating rate. $\Delta \tau_k(r, s)$ is the amount of pheromone trail added to the edge (r, s) by ant k between time t and $t + \Delta t$ in the tour. It is given by:

$$\Delta \tau_k(r, s) = \begin{cases} \frac{Q}{L_k} & (r, s) \in \pi_k \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

Where Q is a constant parameter, L_k is the distance of the sequence π_k toured by ant in Δt .

3. Quantum Evolutionary Algorithm (QEA)

In quantum algorithms, the smallest information unit is Qubit, which sometimes is called quantum bit [31]. A Qubit can not only represent the states of 0 and 1, but also represent arbitrary superposition state between two states. That's to say that a Qubit may be in $|1\rangle$ or $|0\rangle$, or in the intermediate state of the different superposition state of $|1\rangle$ and $|0\rangle$. So a Qubit state can be expressed as:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \quad (4)$$

Where α and β respectively represent the probability amplitude of $|1\rangle$ and $|0\rangle$, and meet the following normalized condition:

$$|\alpha|^2 + |\beta|^2 = 1 \quad (5)$$

Where $|\alpha|^2$ represents the probability of the observed value (zero) of quantum state, $|\beta|^2$ represents the probability of the observed value (one) of quantum state. A pair of real numbers (α and β) with meeting the above equations is called a probability amplitude of Qubit, denoted as $[\alpha, \beta]^T$. Then, arbitrary quantum individual P_j with m Qubits can be expressed as:

$$P_j = \begin{bmatrix} \alpha_1 & \alpha_2 & \alpha_3 & \dots & \alpha_m \\ \beta_1 & \beta_2 & \beta_3 & \dots & \beta_m \end{bmatrix} \quad (6)$$

Where $|\alpha_i|^2 + |\beta_i|^2 = 1, i = 1, 2, 3, \dots, m$.

The probability amplitude of quantum bit can be updated by using the quantum gate in QEA. The updated probability amplitude (α' and β') also meet the normalized condition $|\alpha'|^2 + |\beta'|^2 = 1$. The following quantum revolving door is used to update the probability amplitude:

$$U(\Delta\theta_i) = \begin{bmatrix} \cos(\Delta\theta_i) & -\sin(\Delta\theta_i) \\ \sin(\Delta\theta_i) & \cos(\Delta\theta_i) \end{bmatrix} \quad (7)$$

Where $\Delta\theta_i$ ($i = 1, 2, 3, \dots, m$) represents revolving angle of quantum bit state of $|1\rangle$ and $|0\rangle$.

4. An Improved Quantum Ant Colony Optimization (IMAQACO) Algorithm

The QEA a newly developed probability evolutionary algorithm. The evolutionary computing theory and quantum computation are introduced in to the ACO algorithm to propose a quantum ant colony optimization (QACO) algorithm. In the QACO algorithm, each ant carries a group of Qubits, which represent the current position information. The next city is selected by ant according to the constructed selection probability based on pheromone concentration and visibility. Then the quantum rotation gate is used to the Qubits of ant according to the ant movement. The quantum non-gate is used to realize the position mutation of ant in order to increase position diversity. Finally, the pheromone concentration and visibility are updated according to the moved position. Two probability amplitudes of Qubit in the QACO algorithm are considered as the current position information of ant. When the number of ants is equal, the QACO algorithm can enlarge search space. So it can better solve the slow convergence speed and easy falling into local optimum of ACO algorithm.

Although traditional QACO algorithm has been strengthened the solving performance, it is constrained by pheromone evaporation factor and rotation angle amplitude of

quantum rotation gate to make the slow convergence speed and easy falling into local optimum of ACO algorithm in solving complex optimization problems. In order to solve these existing problems, an improved quantum ant colony optimization (IMAQACO) algorithm is proposed in this paper. In the IMAQACO algorithm, the quantum state vectors are used to represent the pheromone, the adaptively dynamical updating strategy is used to control pheromone evaporation factor, the quantum rotation gate is used to realize the ant movement and change the convergence tend of quantum probability amplitude, quantum non-gate is used to realize ant location variation, so the IMAQACO algorithm has better global search ability and population diversity than ACO algorithm.

4.1. Adaptive Adjustment Strategy of Quantum Rotation Gate

In the conventional quantum algorithm, the quantum rotation gate $U(\Delta\theta_i)$ is used to update operation. Because $\Delta\theta_i$ represents rotation angle of quantum bit, the smaller value of $\Delta\theta_i$ could affect the convergence speed, the bigger value of $\Delta\theta_i$ could enable the divergence or premature convergence to local optimal solution. So in this paper, the value of $\Delta\theta_i$ is dynamically adjusted in $[0.1\pi, 0.05\pi]$ according to different evolution iterations. The specific calculation is given:

$$\Delta\theta = 0.1\pi - \frac{0.1\pi - 0.005\pi}{G} \times t \quad (8)$$

Where G is the total number of iterations in the whole evolution process, t the number of evolution iterations. In the initial iteration, the value of $\Delta\theta_i$ is bigger in order to accelerate updating the pheromone to the target position. With the increasing of the iteration, the value of $\Delta\theta_i$ is decreasing in order to guarantee the pheromone with updating ability in the local area. By dynamical updating the value of $\Delta\theta_i$, the global and local pheromone update ability can effectively be guaranteed.

4.2. The Position Update of Ant

In the IMAQACO algorithm, the increment of pheromone intensity on passed path in the ordinary ACO algorithm is translated into the update of rotation angle of the quantum rotation gate. Then the quantum rotation gate is used to update the Qubit of ant in order to complete the ant movement. So the ant position update is changed into the probability amplitude update of Qubit of ant. But the size and direction of rotation angle of quantum gate in the IMAQACO algorithm is obtained without checking list, it is automatically adjusted by using evolution equation. The mathematical expression of evolutionary equation can be expressed as follows:

$$\theta = T_1(P_i - X_i) + T_2(P - X_i) + T_3(P_j - X_i) \quad (9)$$

Where P_i is an extreme value of individual in the quantum ant population, P is the global extremum, P_j an extreme value of individual in the neighborhood quantum ant population. $T_i (i = 1, 2, 3)$ is the influence factor. The above equation can automatically adjust the size and direction of rotation angle of quantum gate.

4.3. The Position Mutation of Ant

In many conditions, the ACO algorithm is easy to fall into local optimum because of the loss population diversity in the search space. So the mutation operator is introduced into evolutionary algorithm in order to increase the population diversity and avoid the premature convergence. The IMAQACO algorithm uses a quantum mutation quantum of quantum non-gate. the specific steps are as follows:

(1) Several individuals are selected from the population according to the determined probability P_n .

(2) The selected individual is determined one or more mutation positions according to determined probability.

(3) The selected probability of Qubit is executed quantum non-gate operation. That's to say that the mutation operation of Qubit is completed.

Quantum mutation operation actually changes the superposition state of Qubit in order to make the original tendency "1" state to tendency "0" state, or vice versa. In fact, the superposition state of quantum chromosome is changed.

4.4. The Flow Description of IMAQACO Algorithm

Step 1: Initialize quantum ant colony $A(t)$, set population size, the maximum number of iteration and other parameters. When the number of iteration is $t = 0$, $\tau_{ij}(0) = 1$. And in order to ensure all states with the same probability in the search of initial state, all probability amplitudes $\alpha_i = \beta_i = \sqrt{2} / 2$.

Step 2: Construct the feasible solutions. The pseudo random selection rule is used in here. Each ant completed one path, which is regarded as one feasible solution, and the corresponding fitness value is calculated.

Step 3: Record the optimal solution, the pheromone is globally updated according to the optimal solution. The update strategy is adaptive rotation angle strategy.

Step 4: The adaptive adjustment strategy of quantum rotation gate is used to update quantum ant colony $A(t)$.

Step 5: The quantum non-gate is used to realize mutation operation of pheromone matrix.

Step 6: Determine whether the condition is satisfied. If the optimal solution keep the unchanging in the successive iteration, the IMAQACO algorithm finds the optimal solution and is terminated. Otherwise go to Step 2.

Step 7: Output global optimal solution.

5. Experiment Results and Analysis

In order to verify the performance of the proposed IMAQACO algorithm, several famous benchmark functions are tested in the experiment. The proposed IMAQACO algorithm is compared with ACO algorithm and QACO algorithm. The environments are followed: the Pentium CPU 2.30GHz, 4.0GB RAM with the Windows operating system, Matlab2012b. Because the initial values of parameters could seriously affect the experiment result, the most reasonable initial values of these parameters are obtained by testing and modifying. The obtained initial values of these parameters are: ants $m = 40$, pheromone factor $\alpha = 1.0$, heuristic factor $\beta = 1.0$, evaporation factor $\rho = 0.10$, pheromone amount $Q = 60$, maximum iteration times $T_{\max} = 500$. The specific expression, dimension, range and optimum value of famous benchmark functions are shown in Table 1.

Table 1. Benchmark Test Functions

index	Expression	Opt.	Dim.	Ran.
F_1	$f(x) = \sum_{i=1}^n x_i^2$	0	30	[-100,100]
F_2	$f(x) = -20 \exp\left[\sum_{i=1}^n \frac{x_i^2}{n}\right] - \exp\left(\sum_{i=1}^n \cos(2\pi x_i)/n\right) + 20 + e$	0	30	[-32,32]
F_3	$f(x) = \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i) + 10)$	0	30	[-5.12,5.12]
F_4	$f(x) = \max_i \{ x_i , 1 \leq i \leq n\}$	0	30	[-30,30]
F_5	$f(x) = \frac{(\sin \sqrt{x_1^2 + x_2^2})^2 - 0.5}{[1 + 0.001(x_1^2 + x_2^2)^2]} - 0.5$	-1	100	[-100,100]

After the proposed IMAQACO algorithm is independently run on 30 times for each famous benchmark function, the experimental results are shown in Table 2. The maximum value, the minimum value, the mean value and the standard deviation are used to describe the experiment results.

Table 2. The Experimental Results (30 times)

Function	Algorithm	Maximum value	Minimum value	Mean value	Standard deviation
F_1	ACO	4.625 781e-14	6.146 351e-16	9.241 442e-15	3.472 447e-08
	QACO	8.452 357e-20	3.283 572e-22	8.326 573e-21	2.341 869e-09
	IMAQACO	3.538 827e-30	3.573 258e-32	8.157 491e-30	1.537 249e-09
F_2	ACO	6.456 361e-04	3.152 648e-07	6.563 690e-05	3.749 368e-06
	QACO	7.562 481e-10	4.492 602e-14	8.582 479e-11	7.038 433e-08
	IMAQACO	6.802 605e-14	8.305 472e-17	3.504 168e-15	4.057 362e-08
F_3	ACO	5.627 581e-17	4.804 382e-20	6.048 162e-18	5.624 480e-06
	QACO	2.416 803e-21	4.602 583e-24	8.721 603e-21	2.057 239e-08
	IMAQACO	4.409 327e-29	8.067 427e-33	5.782 462e-30	4.357 304e-10
F_4	ACO	4.056 374e-01	5.673 061e-03	3.140 692e-02	2.380 572e-01
	QACO	7.140 851e-03	3.062 169e-07	2.581 405e-04	2.054 573e-03
	IMAQACO	5.562 056e-05	4.890 241e-07	3.705 376e-04	4.053 572e-05
F_5	ACO	-8.974 439e-01	-0.950 395e-01	-9.140 368e-01	3.056 285e-02
	QACO	-9.358 467e-01	-9.735 046e-01	-9.501 524e-01	6.063 793e-03
	IMAQACO	-9.405 672e-01	-9.905 362e-01	-9.604 263e-01	1.582 305e-04

As can be seen from Table 2, the proposed IMAQACO algorithm can obtain better experiment results for four famous benchmark functions of F_1, F_2, F_3 and F_5 than the QACO algorithm and for five famous benchmark functions of F_1, F_2, F_3, F_4 and F_5 than the ACO algorithm. But the optimization performance of the QACO algorithm is better for solving F_4 than the proposed IMAQACO algorithm. And the QACO algorithm

can obtain better experiment results than the ACO algorithm in solving five famous benchmark functions. In general, the optimization performance of the proposed IMAQACO algorithm is more stable in solving the high-dimensional complex benchmark functions, and it takes on the higher optimization ability and solving accuracy.

6. Conclusion

The ACO algorithm is simple, versatile, robust and parallel method. It has been successfully applied in quadratic assignment problems, dynamic routing problems, scheduling problems, TSP and so on. However, the ACO algorithm exists the long search time, local minimum and low accuracy in the actual application. In order to solve the existing problems, the quantum evolutionary algorithm is introduced into ant colony optimization algorithm to propose an improved quantum ant colony optimization (IMAQACO) algorithm for solving complex function problems. In this algorithm, the quantum state vectors are used to represent the pheromone, the adaptively dynamical updating strategy is used to control pheromone evaporation factor, the quantum rotation gate is used to realize the ant movement and change the convergence tend of quantum probability amplitude, quantum non-gate is used to realize ant location variation. Several benchmark functions are selected to test the optimization performance of IMAQACO algorithm. The experiment results indicate that the proposed IMAQACO can avoid falling into the local optimum, improve the convergence speed and take on stronger global optimization ability and higher convergence speed.

References

- [1] C. Blum and A. Roli, "Metaheuristics in combinatorial optimization: Overview and conceptual comparison", *ACM Computing Surveys*, vol. 35, no. 3, (2003), pp. 268-308.
- [2] E. Corchado and A. Abraham, "Hybrid intelligent algorithms and applications", *Information Sciences*, vol. 180, no. 14, (2010), pp. 2633-2634.
- [3] L. N. Xing, Y. W. Chen and H. P. Cai, "An intelligent genetic algorithm designed for global optimization of multi-minima functions", *Applied Mathematics and Computation*, vol. 178, no. 2, (2006), pp. 355-371.
- [4] S. K. Chaharsooghi and A. H. M. Kermani, "An effective ant colony optimization algorithm (ACO) for multi-objective resource allocation problem (MORAP)", *Applied Mathematics and Computation*, vol. 200, no. 1, (2008), pp. 167-177.
- [5] M. Dorigo and M. Gambardellal, "Ant colony system: A cooperative learning approach to the traveling salesman problem", *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, (1997), pp. 53-66.
- [6] S. Leng, X. B. Wei and W. Y. Zhang, "Improved ACO scheduling algorithm based on flexible process", *Transactions of Nanjing University of Aeronautics and Astronautics*, vol. 23, no. 2, (2006), pp. 154-160.
- [7] M. R. Jalali, A. Afshar and M. A. Mariño, "Improved ant colony optimization algorithm for reservoir operation", *Scientia Iranica*, vol. 13, no. 3, (2006), pp. 295-302.
- [8] Z. Q. Zhang, W. M. Cheng, B. Zhong and J. N. Wang, "Improved ant colony optimization for assembly line balancing problem", *Computer Integrated Manufacturing Systems, CIMS*, vol. 13, no. 8, (2007), pp. 1632-1638.
- [9] W. Kanyapat and W. Paramote, "Reliability optimization of topology communication network design using an improved ant colony optimization", *Computers and Electrical Engineering*, vol. 35, no. 5, (2009), pp. 730-747.
- [10] K. W. Li and J. Tian, "The multicast routing QoS based on the improved ACO algorithm", *Journal of Networks*, vol. 4, no. 6, (2009), pp. 505-510.
- [11] Y. Yi and J. L. Lai, "Computation model and improved ACO algorithm for p/T", *Journal of Systems Engineering and Electronics*, vol. 20, no. 6, (2009), pp. 1336-1343.
- [12] B. Yu, Z. Z. Yang and B. Z. Yao, "An improved ant colony optimization for vehicle routing problem", *European Journal of Operational Research*, vol. 196, no. 1, (2009), pp. 171-176.
- [13] A. Kaveh and S. Talatahari, "An improved ant colony optimization for constrained engineering design problems", *Engineering Computations (Swansea, Wales)*, vol. 27, no. 1, (2010), pp. 155-182.
- [14] J. G. Yang and Y. B. Zhuang, "An improved ant colony optimization algorithm for solving a complex combinatorial optimization problem", *Applied Soft Computing Journal*, vol. 10, no. 2, (2010), pp. 653-660.

- [15] R. W. Gan, Q. S. Guo, H. Y. Chang and Y. Yi, "Improved ant colony optimization algorithm for the traveling salesman problems", *Journal of Systems Engineering and Electronics*, vol. 21, no. 2, (2010), pp. 329-333.
- [16] B. Yu, Z. Z. Yang and J. X. Xie, "A parallel improved ant colony optimization for multi-depot vehicle routing problem", *Journal of the Operational Research Society*, vol. 62, no. 1, (2011), pp. 183-188.
- [17] J. Q. Geng, L. P. Weng and S. H. Liu, "An improved ant colony optimization algorithm for nonlinear resource-leveling problems", *Computers and Mathematics with Applications*, vol. 61, no. 8, (2011), pp. 2300-2305.
- [18] J. Raka and T. Milan, "An ant colony optimization algorithm with improved pheromone correction strategy for the minimum weight vertex cover problem", *Applied Soft Computing Journal*, vol. 11, no. 8, (2011), pp. 5360-5366.
- [19] Q. L. Ding, X. P. Hu, L. J. Sun and Y. Z. Wang, "An improved ant colony optimization and its application to vehicle routing problem with time windows", *Neurocomputing*, vol. 98, no. 12, (2012), pp. 101-107.
- [20] H. L. Xu, X. Qian and L. Zhang, "Study of ACO algorithm optimization based on improved tent chaotic mapping", *Journal of Information and Computational Science*, vol. 9, no. 6, (2012), pp. 1653-1660.
- [21] Y. C. Li, L. P. Liu, S. J. Zhou and Z. C. Yang, "Dynamic resource scheduling in construction project group management based on improved ACO algorithm", *Energy Education Science and Technology Part A: Energy Science and Research*, vol. 31, no. 2, (2013), pp. 1111-1116.
- [22] B. Y. Cheng, Q. Wang, S. L. Yang and X. X. Hu, "An improved ant colony optimization for scheduling identical parallel batching machines with arbitrary job sizes", *Applied Soft Computing Journal*, vol. 13, no. 2, (2013), pp. 765-772.
- [23] P. Lalbakhsh, B. Zaeri and A. Lalbakhsh, "An improved model of ant colony optimization using a novel pheromone update strategy", *IEICE Transactions on Information and Systems*, vol. 96, no. 11, (2013), pp. 2309-2318.
- [24] Y. Y. Su, H. Dong and D. Liang, "Improved ant colony optimization for virtual teams building in collaborative process planning", *Sensors and Transducers*, vol. 165, no. 2, (2014), pp. 245-250.
- [25] J. H. Shen and J. Chen, "An improved ACO based RWA algorithm and its application in wavelength converter allocation issue of the intelligent optical networks", *Journal of Computational Information Systems*, vol. 10, no. 8, (2014), pp. 3341-3349.
- [26] S. K. Chong, M. S. Mohamad, A. H. Mohamed Salleh, Y. W. Choon, C. K. Chong and S. Deris, "A hybrid of ant colony optimization and minimization of metabolic adjustment to improve the production of succinic acid in *Escherichia coli*", *Computers in Biology and Medicine*, vol. 49, no. 1, (2014), pp. 74-82.
- [27] Y. L. Tang, Y. G. Cai and Q. J. Yang, "Improved ant colony optimization for multi-depot heterogeneous vehicle routing problem with soft time windows", *Journal of Southeast University (English Edition)*, vol. 31, no. 1, (2015), pp. 94-99.
- [28] S. C. Pang, T. M. Ma and T. Liu, "An improved ant colony optimization with optimal search library for solving the traveling salesman problem", *Journal of Computational and Theoretical Nano-science*, vol. 12, no. 7, (2015), pp. 1440-1444.
- [29] W. Y. Jiang, Y. Lin, M. Chen and Y. Y. Yu, "A co-evolutionary improved multi-ant colony optimization for ship multiple and branch pipe route design", *Ocean Engineering*, vol. 102, no. 4, (2015), pp. 63-70.
- [30] B. Y. Cheng, J. Y. T. Leung and K. Li, "Integrated scheduling of production and distribution to minimize total cost using an improved ant colony optimization method", *Computers and Industrial Engineering*, vol. 83, no. 5, (2015), pp. 217-225.
- [31] K. H. Han and J. H. Kim, "Quantum-inspired evolutionary algorithms with a new termination criterion Gate, and Two-Phase Scheme", *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 2, (2004), pp. 156-169.

Authors



Changai Chen, Lecturer, received the Master degree in Control Science and Engineering from Wuhan University of Technology in 2009, Wuhan, China. The main research directions: Artificial Intelligence, Computer Simulation.



Yanwen Xu, Lecturer, received the Master degree in Computer application from Zhengzhou University in 2009, Zhengzhou, China. The main research directions: Computer application technology, Database technology.

