IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received January 17, 2019, accepted January 31, 2019, date of publication February 5, 2019, date of current version February 22, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2897580

An Improved Ant Colony Optimization Algorithm Based on Hybrid Strategies for Scheduling Problem

WU DENG[1,2,3], (Member, IEEE), JUNJIE XU[1], AND HUIMIN ZHAO[1,3],

[1]College of Electronic Information and Automation, Civil Aviation University of China, Tianjin 300300, China
[2]Co-innovation Center of Shandong Colleges and Universities: Future Intelligent Computing, Yantai 264005, China
[3]Liaoning Key Laboratory of Welding and Reliability of Rail Transportation Equipment, Dalian Jiaotong University, Dalian 116028, China

Corresponding author: Huimin Zhao (hm_zhao1977@126.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61771087, Grant 51605068, and Grant 51875072, in part by the Innovative Talents Promotion Plan of Liaoning Colleges and Universities under Grant LR2017058, and in part by the Liaoning BaiQianWan Talents Program.

ABSTRACT In this paper, an improved ant colony optimization(ICMPACO) algorithm based on the multi-population strategy, co-evolution mechanism, pheromone updating strategy, and pheromone diffusion mechanism is proposed to balance the convergence speed and solution diversity, and improve the optimization performance in solving the large-scale optimization problem. In the proposed ICMPACO algorithm, the optimization problem is divided into several sub-problems and the ants in the population are divided into elite ants and common ants in order to improve the convergence rate, and avoid to fall into the local optimum value. The pheromone updating strategy is used to improve optimization ability. The pheromone diffusion mechanism is used to make the pheromone released by ants at a certain point, which gradually affects a certain range of adjacent regions. The co-evolution mechanism is used to interchange information among different sub-populations in order to implement information sharing. In order to verify the optimization performance of the ICMPACO algorithm, the traveling salesmen problem (TSP) and the actual gate assignment problem are selected here. The experiment results show that the proposed ICMPACO algorithm can effectively obtain the best optimization value in solving TSP and effectively solve the gate assignment problem, obtain better assignment result, and it takes on better optimization ability and stability.

INDEX TERMS Co-evolution mechanism, ACO, pheromone updating strategy, pheromone diffusion mechanism, hybrid strategy, assignment problem.

I. INTRODUCTION

Ant colony optimization (ACO) algorithm was proposed by Dorigo in 1992 [1]. It is a heuristic evolutionary algorithm based on population, which is inspired by the research results of the collective behavior of the real ants in nature. It has been proved that the ACO algorithm takes on a better optimization performance in solving optimization problems. The ACO algorithm relies on the activities of many individualities and feedback of information. Although the activity of ant is very simple, the activity of whole ant colony is acceptable. The ACO algorithm has the characteristics of distributed computing, positive feedback and heuristic search. In essence, it is a heuristic global optimization algorithm in the evolutionary algorithm [2]–[11]. In process of the evolution, the information interaction based on pheromone plays a very important role.

Due to the advantages of the ACO algorithm, it is widely applied in solving combinatorial optimization problems, such as the traveling salesman problem, assignment problem, job-shop scheduling problem, vehicle routing problem, graph coloring problem and network routing problem and so on [12]–[25]. A lot of experts have devoted themselves to the research of the ACO algorithm, and some improved ACO algorithms are proposed to solve the complex optimization problems. Some better results and effects

The associate editor coordinating the review of this manuscript and approving it for publication was Dongxiao Yu.

are obtained in recent years. But with the increasing of the complexity of large-scale optimization problems, these improved ACO algorithms have some inherent shortcomings in solving large-scale optimization problems, such as the slow convergence speed, local optimum value, and so on [26]–[39]. Therefore, in order to improve the optimization performance of the ACO algorithm, the multi-population strategy, co-evolution mechanism, pheromone updating strategy, and pheromone diffusion mechanism are introduced into the ACO algorithm in order to propose a novel multi-population co-evolution ant colony optimization (ICMPACO) algorithm. The traveling salesmen problem (TSP) and actual gate assignment problem are selected to verify the effectiveness of the ICMPACO algorithm.

II. RELATED WORK

For the studies of the ACO algorithm, some experts and scholars proposed a lot of improved ACO algorithms in recent years. Coelho et al. [40] proposed a modified continuous approach of ACO combined with a differential evolution method. Rada-Vilela et al. [41] proposed eight multi-objective ACO algorithms for the time and space assembly line balancing problem. He and Ma [42] proposed a multi-objective ACO algorithm to study non-redundant linear sensor network design problems. Juang *et al.* [43] proposed a co-evolution continuous ACO algorithm to address the accuracy-oriented fuzzy systems design problems. Yang et al. [44] proposed an improved ACO algorithm, which combines swarm intelligence with local search to improve the efficiency and accuracy. Myszkowski *et al.* [45] proposed a hybrid ACO algorithm based on classical heuristic priority rules with ACO algorithm. Ariyasingha and Fernando [46] proposed a multi-objective ACO for solving most real world combinatorial optimization problems. Jiang *et al.* [47] proposed a co-evolutionary multi-ant colony optimization algorithm for ship multi and branch pipe route design under various kinds of constraints. Zuo *et al.* [48] proposed a multi-objective optimization method based on the ACO algorithm for task-scheduling problems. Bagherinejad and Dehghani [49] proposed a swarm intelligence-based algorithm named Non-dominated sorting ACO algorithm for capacitated multivehicle allocation of customers. Krynicki *et al.* [50] proposed a multi-pheromone variant of ACO system. Chen and Wang [51] proposed a novel offensive strategy based on multi-group ACO for the 2D soccer simulation league. Marzband *et al.* [52] proposed a multi-layer ACO algorithm for real time experimental implementation of optimum energy management system. Khan and Baig [53] proposed a novel method to find the relevant feature subset by using ACO minimum-redundancy-maximum-relevance. Zhong and Ai [54] proposed a novel ACO algorithm, called modified ACO algorithm for multi-objective single-model assembly line balancing problem. Huang and Yu [55] proposed several novel hybrid ant colony optimization-based algorithms to resolve multi-objective job-shop scheduling problem with equal-size lot splitting. Zhou *et al.* [56] proposed a multi-objective multi-population ACO algorithm for continuous domain. Vijayalakshmi *et al.* [57] proposed a novel robust energy efficient ACO routing algorithm to enhance the performance of Max-Min-Path approach. Tiwari and Vidyarthi [58] proposed an improved auto controlled ACO algorithm using the lazy ant concept. Sharifipour et al. [59] proposed a memetic ACO algorithm based on evolution strategies for structural test data generation. Akka and Khaber [60] proposed an improved ACO algorithm that uses a stimulating probability to help the ant in its selection of the next grid and employ new heuristic information based on the principle of unlimited step length to expand the vision field and increase the visibility accuracy. Chen and Shen [61] proposed a new population-based evolutionary optimization algorithm, elite-mixed continuous ACO algorithm with central initialization for improving the accuracy of Takagi-Sugeno-Kang-type recurrent fuzzy network designs. The other methods are also proposed in recent year [62]–[68].

As can be known from the above analysis of these related works, the researchers have proposed hybrid ACO algorithm, the multi-objective ACO, co-evolution continuous ACO, the multi-group ACO, the multi-layer ACO, the multi-objective multi-population ACO, the auto controlled ACO, memetic ACO and elite-mixed continuous ACO and so on. These improved ACO algorithms are used to solve the complex optimization problems and obtain better optimization results. But they still exist the slow convergence speed, and are easy to fall into local optimum value. Therefore, it is necessary to further and deeply study an improved ACO algorithm with the better optimization performance. In this paper, the multi-population strategy, co-evolution mechanism, pheromone updating strategy and pheromone diffusion mechanism are introduced into the ACO algorithm to propose a novel improved ant colony optimization(ICMPACO) algorithm.

III. THE ACO ALGORITHM

The ACO algorithm consists of a number of iterations. In each iteration, a number of ants construct complete solutions by using heuristic information and the collected experiences of previous populations of ants. These collected experiences are represented by using the pheromone trail, which is deposited on the constituent elements of a solution. The pheromone can be deposited on the components and/or the connections in a solution depending on the solving problem. The procedure of pheromone update rule is described as follows.

A. THE TRANSITION RULE

An ant is a simple computational agent in the ACO algorithm. It iteratively constructs a solution for the problem at hand. At each iteration of the algorithm, each ant moves from a state r to state s, corresponding to a more complete intermediate solution. The k^{th} ant from state r to state s is selected among the unvisited states memorized in J_r^k according to the following formula:

$$s = \arg\max_{u \in J_r^k} [\tau_i(r,u)^{\alpha} \cdot \eta(r,u)^{\beta}] \quad if\ q \le q_0(Exploitation) \tag{1}$$

The trail level represents a posteriori indication of the desirability of that move. Trails are updated usually when all ants have completed their solution, increasing or decreasing the level of trails corresponding to moves that were part of "good" or "bad" solutions, respectively.

In general, the kth ant moves from state r to state s with the probability $p_k(r,s)$,

$$p_k(r,s) = \begin{cases} \dfrac{\tau(r,s)^{\alpha} \cdot \eta(r,s)^{\beta}}{\sum\limits_{u \in J_r^k} \tau(r,u)^{\alpha} \cdot \eta(r,u)^{\beta}} & if\ s \in J_r^k \\ 0 & otherwise \end{cases} \tag{2}$$

In here, $p_k(r,s)$ is the transition probability, $\tau\ (r,u)$ is the pheromone concentration between the state r and the state u in the i^{th} population, $\eta\ (r,u)$ is the length of the trail from the state r and the state u, J_r^k is the set of unvisited states 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].

B. THE PHEROMONE UPDATE RULE

To improve the quality of solution, the pheromone trails must be updated. Trail updating includes local and global updating. The local trail updating formula is described as follow:

$$\tau(r,u) = (1-\rho)\tau(r,s) + \sum_{k=1}^{m} \Delta\tau_k(r,s) \tag{3}$$

In the formula (3), ρ $(0 < \rho < 1)$ is the pheromone trial 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 described as follow:

$$\Delta\tau_k(r,s) = \begin{cases} \frac{Q}{L_k} & \text{(r,s)} \in \pi_k \\ 0 & otherwise \end{cases} \tag{4}$$

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

IV. A NEW MULTI-POPULATION CO-EVOLUTION ANT COLONY OPTIMIZATION(ICMPACO) ALGORITHM

In the actual applications of the ACO algorithm, the ACO algorithm takes on the positive feedback, parallelism and better optimization performance in solving optimization problems. But it exists deficiencies of premature convergence and difficultly determining control parameters, slow convergence speed, and so on. The co-evolution algorithm is a global optimization algorithm inspired by the co-evolutionary phenomenon in nature. It adopts the idea of decomposition and coordination to decompose the complex optimization problem into multiple interacting optimization sub-problems, which are optimized separately and coordinate wholly. Therefore, the multi-population strategy, the co-evolution mechanism, the pheromone updating strategy and pheromone diffusion mechanism are introduced into the ACO algorithm in order to propose a new multi-population co-evolution ant colony optimization(ICMPACO) algorithm for solving large-scale optimization problem in this paper. In the ICMPACO algorithm, the multi-population strategy is used to divide ants into elite ants and common ants to improve the convergence rate and avoid to fall into local optimum value. The pheromone updating strategy is used to improve the optimization ability. The pheromone diffusion mechanism is used to make the pheromone released by ants at a certain point, which gradually affects a certain range of adjacent regions. The co-evolution mechanism is used to interchange information among different sub-populations to implement the information sharing. These strategies and mechanisms are fully used to improve the optimization performance of the ACO algorithm.

A. MULTI-POPULATION STRATEGY

In the ACO algorithm, only one kind of ants is used to generate new solutions, and the ant colony size, selection parameter and convergence parameter are used to control the solutions. In general, it is difficult to determine the appropriate values of parameters in order to obtain the improved ACO algorithm with fast convergence speed and avoiding premature convergence. Therefore, a multi-population strategy is used to improve the ACO algorithm. This strategy will divide ants into elite ants and common ants. The elite ants can obtain information from solution archive, and generate solutions by adopting a Gaussian kernel function and a probability selection strategy. The difference is that the elite ants have a set of own parameters. In essence, the elite ants are used to improve the convergence rate of the ACO algorithm. The common ants are used to generate new solutions with the slower speed by adopting the single Gaussian function and the average value of each dimension in order to avoid to fall into local optimum value. The Gaussian function for the normal ants is described as follow.

$$f_N^i(x) = \frac{1}{\sigma_{i,N}\sqrt{2\pi}} e^{-\frac{(x-\mu_{i,N})^2}{2\sigma_{i,N}^2}} \tag{5}$$

$$\mu_{i,N} = \sum_{k=1}^{K} s_{i,k} \tag{6}$$

$$\sigma_{i,N} = \xi_N \sum_{e=1}^{K} \frac{|s_{i,e} - \overline{s_i}|}{K-1} \tag{7}$$

where $f_N^i(x)$ is the Gaussian function for normal ants in the i^{th} dimension, $\mu_{i,N}$ is its sample value, and $\sigma_{i,N}$ is calculated standard deviation. s_i is the average value of solutions in the i^{th} dimension, ξ_N is a constant which is used to control the convergence rate of common ants. Therefore, the common ants can effectively enlarge the search range and enhance global search ability.

B. PHEROMONE UPDATING STRATEGY

In the ACO algorithm, the pheromone updating is a key problem, it includes the local pheromone updating and the global pheromone updating. In order to improve the optimization performance of the ACO algorithm in solving complex optimization problem, the new pheromone updating strategy and pheromone diffusion mechanism are proposed to improve the ACO algorithm.

1) THE LOCAL PHEROMONE UPDATING STRATEGY

Before the first iteration of the ACO algorithm is executed, the pheromones on each edge are equal constants. When any ant in the ACO algorithm completes the current iteration, the local pheromone updating strategy is carried out on the each passed edge for ant. The expression of the local pheromone updating strategy is described as follow:

$$\tau_{x,y}^{(i)} = (1-\rho_L)\tau_{x,y}^{(i)} + \rho_L \Delta\tau_0^{(i)} \tag{8}$$

where, $\rho_L \in (0, 1)$ is local pheromone evaporating coefficient, $1-\rho_L$ is the pheromone residue factor, $\tau_0^{(i)}$ is the initial pheromone value. When the node value is 1, $\tau_0^{(i)}$ is a suitable small negative number. When the node value is 0, $\tau_0^{(i)}$ is 0.

2) THE GLOBAL PHEROMONE UPDATING STRATEGY

In one iteration, after all ants in the ACO algorithm complete their solutions, the passed nodes are carried out the global pheromone updating strategy. The expressions of the local pheromone updating strategy are described as follows:

$$\tau_{x,y}^{(i)} = (1-\rho_G)\tau_{x,y}^{(i)} + \rho_G \Delta\tau_G^{(i)} \tag{9}$$

$$\Delta\tau_G^{(i)} = \begin{cases} F_G^{(i)}, & (x, j) \in \text{Global optimal solution} \\ F_I^{(i)}, & (x, j) \in \text{Iterative optimal solution} \\ 0, & otherwise \end{cases} \tag{10}$$

where, $\rho_G \in (0, 1)$ is global pheromone evaporating coefficient, $1-\rho_G$ is the pheromone residue factor, $F_G^{(i)}$is global optimal solution and $F_I^{(i)}$ is iterative optimal solution.

3) THE PHEROMONE DIFFUSION MECHANISM

In the pheromone updating mechanism, the ant uses a single pheromone release mode. This mode can only affect the subsequent ants with the passed same point, and cannot guide the search of ants in a certain range of neighboring regions. Thus, it will affect the optimization performance. In this paper, on the basis of the local pheromone updating strategy or global pheromone updating strategy, the pheromone diffusion mechanism is used to improve the ACO algorithm. The possibility of better solution in the adjacent neighborhood is usually larger than that in other regions. Therefore, the pheromone diffusion mechanism can make the pheromone released by ants at a certain point, which gradually affects a certain range of adjacent regions. The other ants try to avoid to search the adjacent neighborhood of the poor solution in order to tend to search the adjacent neighborhood of the better solution for improving the optimization performance in solving large-scale optimization problems. The pheromone updating mechanism is described as follows:

$$\tau_{x,y}^{(i)} = (1-\rho_D)\tau_{x,y}^{(i)} + \rho_L \Delta\tau_{x,y}^{(i)} \tag{11}$$

$$\Delta\tau_{x,y}^{(i)} = \begin{cases} \frac{1}{N+1} \times \frac{\tau_x^{(i)}}{d_r(o_x, o_y)}, & d_r(o_x, o_y) < 1 \\ 0, & otherwise \end{cases} \tag{12}$$

where, N is the number of obtained solutions in this iteration, $\tau_x^{(i)}$ is the left guiding pheromone concentration on the source object o_x, $d_r(o_x, o_y) = 1/(f+1)$ is the correlation distance between two objects.

C. THE CO-EVOLUTION MECHANISM

The co-evolution mechanism is a new kind of evolutionary mechanism based on co-evolution theory in recent years. It admits the biological diversity, emphasizes a certain dependence between organisms and organisms, organisms and environment in the process of evolution. It uses co-evolution theory to construct the competition relation or cooperation relation among two or more populations in order to improve the optimization performance by the interaction of multiple populations. It emphasizes the existing interaction among different sub-populations, and affects each other and coevolves together. Therefore, the co-evolution mechanism is introduced into the ACO algorithm to realize information interaction among different sub-populations.

D. THE MODEL AND STEPS OF THE ICMPACO ALGORITHM

The model of the ICMPACO algorithm based on co-evolution mechanism, pheromone updating strategy, pheromone diffusion mechanism and hybrid strategy is shown in Figure 1.

The steps of the ICMPACO algorithm are described as follows:

Step 1. Divide the optimization problem into several sub-problems, and each sub-problem corresponds to one sub-population.

Step 2. Initialize the parameters of the ICMPACO algorithm. These parameters include the number of ants(k), pheromone amount(Q), the maximum number of iterations (T), the parameter(α and β), volatility coefficient(ρ), etc.

Step 3. Randomly select the initial position for each ant.

Step 4. Each sub-population independently execute the search process. The transition probability of the next state is calculated according to Formula(2).

Step 5. Locally update the pheromone concentration of the passed path of ants in each sub-population according to Formula(8).

Step 6. Locally update the pheromone concentration of the adjacent path according to the pheromone diffusion mechanism (Formula(11) and Formula(12)) for each sub-population.

Step 7. Globally update the pheromone concentration for each passed path according to formula (8) and Formula(10).

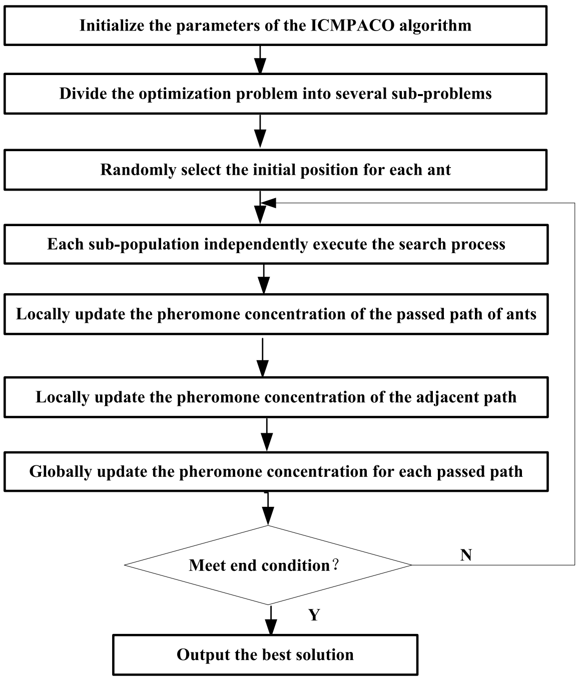


FIGURE 1. **The model of the ICMPACO algorithm.**

Step 8. If each ant executes Step 4 ~ Step 7 in this iteration, then continue to execute next step, otherwise go to Step 4.

Step 9. Determine whether the maximum number of iterations(T) is achieved or the obtained solution has met the requirement. If this end condition does not meet, then execute Step 4 in order to start a new evolution, otherwise go to Step10.

Step 10. After ten iterations are completed, the obtained solutions of all sub-populations are exchanged in order to select better solutions.

V. APPLICATION OF THE ICMPACO ALGORITHM FOR SOLVING TSP

A. TRAVELING SALESMEN PROBLEM

Traveling salesmen problem(TSP)[69,70] is one which has commanded much attention of mathematicians and computer scientists specifically, because it is so easy to describe and so difficult to solve. This problem can simply be stated as follow: a search for the shortest closed tour that visits each city once and only once. The TSP can be represented by a complete directed graph $G = (N, A)$, where N is a set of n nodes (vertices), also called cities, and A is a set of arcs and $D = d_{ij}$ is the cost (distance) matrix associated with each arc $(i, j) \in A$. The cost matrix D can be either symmetric or asymmetric. The TSP is the problem of finding a shortest closed tour visiting each of the $n = |N|$ nodes of G exactly once. The distances between the cities are independent direction of traversing arcs, that is, $d_{ij} = d_{ji}$ is for every pair of nodes in symmetric TSP. In the asymmetric TSP at least for one pair of nodes (i, j), we have $d_{ij} \neq d_{ji}$. All TSP instances used in the empirical studies from the TSPLIB benchmark library.

Define the variables:

$$x_{ij} = \begin{cases} 1 & \text{if } \textit{the arc}(i,j) \textit{ is in the tour} \\ 0 & \textit{otherwise} \end{cases} \tag{13}$$

The TSP can be formulated by the following generalization of a well-known integer program formulation.

Objective function:

$$z = \min \sum_{i} \sum_{j}^{d_{ij}x_{ij}} \tag{14}$$

The constraints are written as follows:

$$\sum_{i=1}^{n} x_{ij} = 1, \quad j = 1, 2, 3, \cdots, n \tag{15}$$

$$\sum_{j=1}^{n} x_{ij} = 1, \quad i = 1, 2, 3, \cdots, n \tag{16}$$

$$x_{ij} \in \{0, 1\}, \quad i, j = 1, 2, 3, \cdots, n \tag{17}$$

$$\sum_{i,j \in S}^{n} x_{ij} \leq |S| - 1, \quad 2 \leq |S| \leq N - 2 \tag{18}$$

In these formulations, the objective function (14) represents the total cost to be minimized. Constraints (15) ensure that each position j is occupied by only one city, while constraint (16) guarantees that each city i is assigned to exactly one position. Constraint (17) represents the integrality constraints of variables zero–one $x_{ij}(x_{ij} \geq 0)$. Constraint (18) assures that each city in the final route will be visited one time and that no sub-routes will be formed.

B. EXPERIMENT ENVIRONMENT AND PARAMETERS

In order to demonstrate the optimization performance of the proposed ICMPACO algorithm, eight TSP standard instances from TSPLIB standard library(http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/) are selected in this paper. According to the characteristic of TSPLIB, the distance between any two cities is computed by the Euclidian distance and then rounded off after the decimal point. The basic ACO algorithm and improved ACO(IACO) based on swarm intelligence with local search are selected to compare the optimization performance with the proposed ICMPACO algorithm. The experiment environments are: Matlab2014b, the Pentium CPU i7, 8.0GB RAM with Windows10. The values of parameters in these algorithms could be a complicated problem itself, the change of parameters could affect the optimum value. In the simulation experiments, the alternative values were tested and modified for some functions to obtain the most reasonable initial values of these parameters. These selected values of the parameters take on the optimal solution and the most reasonable running time of these algorithms to efficiently complete the problem solving. So the most reasonable initial values of these parameters are obtained by testing and modifying. The obtained initial values of these parameters are shown in Table 1.

TABLE 1. The set of the parameters.

Algorithms	ACO	IACO	ICMPACO
Ants(k)	30	30	30
Pheromone factor(α)	1	1	1
Heuristic factor(β)	5	5	5
Volatility coefficient(ρ)	0.1	0.1	0.1
Pheromone amount(Q)	100	100	100
Initial concentration($\tau_{ij}(0)$)	1.5	1.5	1.5
Maximum iterations(T)	200	200	200

C. EXPERIMENT RESULTS AND ANALYSIS

In the simulation experiment, for each TSP standard instance, the basic ACO algorithm, IACO algorithm and the proposed ICMPACO algorithm are executed 10 times, respectively. The indexes of maximum value, minimum value, average value and variance of the 10 results are used to describe and compare the experiment results. The experiment results are shown in Table 2. In here, the optimal value represents the obtained best value. The maximum value represents the obtained maximum value in the simulation test of 10 times, the minimum value represents the obtained minimum value in the simulation test of 10 times, the average value represents the average value of 10 times. The variance represents the variance between the maximum value and the minimum value.

The experiment results of the basic ACO algorithm, the IACO algorithm and the proposed ICMPACO algorithm in solving TSP standard instances of dantzig42, eil51, berlin52, eil101, pr107, ch130, kroA200 and rat783 are shown in Table 2. As can be seen from Table 2, it can clearly see the optimization performance and effect of the basic ACO algorithm, IACO algorithm and the ICMPACO algorithm in solving TSP. For three optimization algorithms, the experiment results show that the ICMPACO algorithm can obtain the best optimization values in solving all TSP standard examples, especially for berlin52, eil51 and dantzig42. The ICMPACO algorithm can obtain 7548.6, 429.8871 and 703.1199, respectively, which are almost close to the optimal value of 7542, 426 and 699. It shows that the ICMPACO

TABLE 2. The experiment results of different TSP.

Instances	Algorithms	Optimal value	Maximum value	Minimum value	Average value	Variance
dantzig42	ACO	699	726.2402	707.7596	718.5473	9.2403
	IACO		737.1451	704.6353	717.58993	16.2549
	ICMPACO		718.8354	**703.1199**	**711.02192**	**7.8578**
eil51	ACO	426	455.8169	443.3749	449.91115	6.221
	IACO		452.6667	440.6427	446.17787	6.012
	ICMPACO		**439.9814**	**429.8871**	**435.24398**	**5.0472**
berlin52	ACO	7542	7757.4	7663.6	7687.21	46.9
	IACO		7681.5	7589	7622.4	66.25
	ICMPACO		**7613.7**	**7548.6**	**7621.36**	**37.55**
eil101	ACO	629	708.2028	683.7806	694.7517	12.2111
	IACO		702.5375	680.3677	692.7951	11.0849
	ICMPACO		**686.246**	**668.236**	**677.4336**	**9.0050**
pr107	ACO	44303	46585	46124	46414.6	230.5
	IACO		46292	45690	46050.3	301
	ICMPACO		**46103**	**45649**	**45970.6**	**227**
ch130	ACO	6110	6473.2	6311.2	6399.8	81
	IACO		6467.8	6307.5	6384.83	80.15
	ICMPACO		**6273.5**	**6183.4**	**6235.95**	**45.05**
kroA200	ACO	29368	37062	32041	33763	132.5
	IACO		36849	31974	32871	**127.3**
	ICMPACO		**36134**	**31267**	**32086**	136.4
rat783	ACO	8806	11508	9932	10791	**70.4**
	IACO		11352	9453	9804	82.1
	ICMPACO		**10891**	**9229**	**9672**	85.3

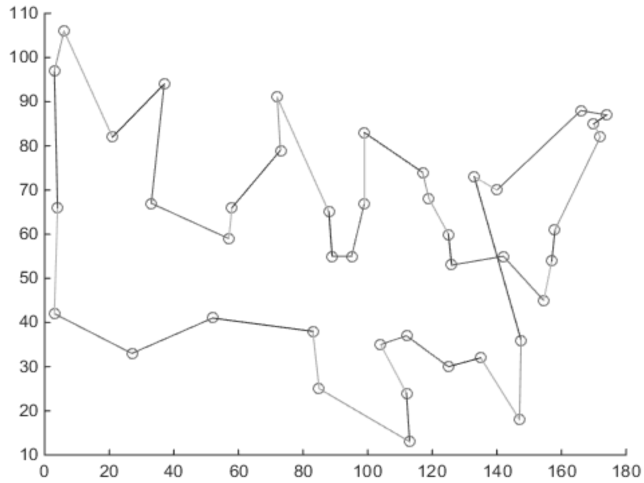


FIGURE 2. The optimal path of dantzig42(703.1199).

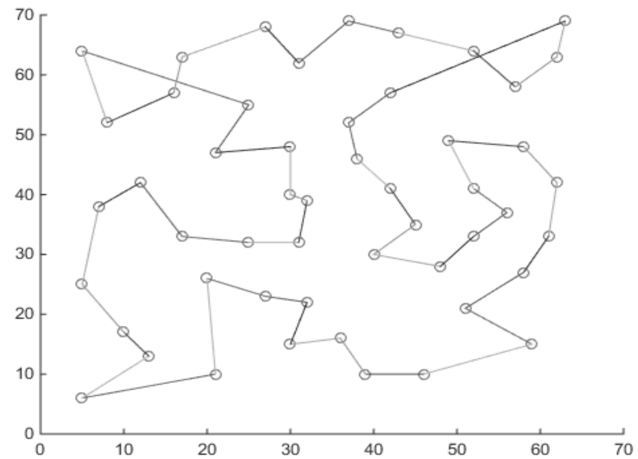


FIGURE 3. The optimal path of eil51 (429.8871).

algorithm has better optimization ability than the basic ACO algorithm and the IACO algorithm in solving these TSP standard instances. From the average value, the ICMPACO algorithm can also obtain the best average value. It shows that the optimization performance of the ICMPACO algorithm is more obvious advantage. It can be seen that the variance of the optimization performance of the proposed ICMPACO algorithm in solving the TSP is also the smallest value for dantzig42, eil51, berlin52, eil101, pr107, ch13, which shows that the ICMPACO algorithm has better stability than the basic ACO algorithm and the IACO algorithm.

In order to further describe the optimization performance of the proposed ICMPACO algorithm, the best routes found by the ICMPACO algorithm for TSP and their costs (their route lengths) are illustrated in Figure 2. ∼Figure 9. Note that the way the network grows, as far as possible like an expanding ring, reduces the possibility of crossings in the routes, which are characteristic of locally optimal routes.

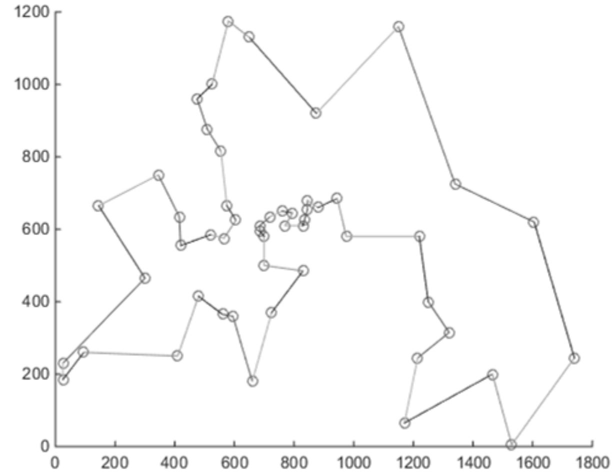


FIGURE 4. The optimal path of rat783 (7548.6).

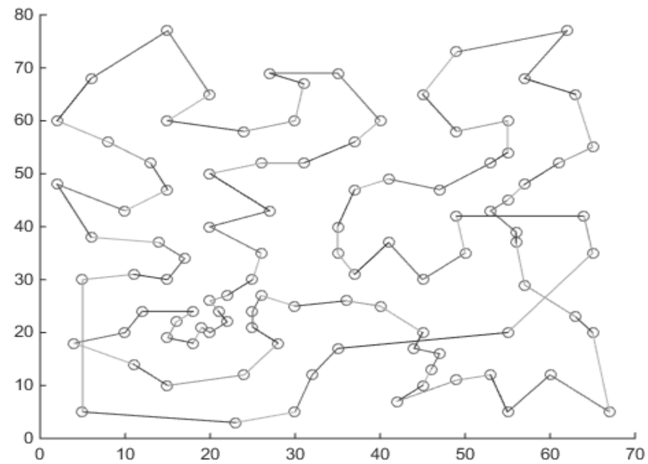


FIGURE 5. The optimal path of eil101(668.236).

VI. APPLICATION OF THE ICMPACO ALGORITHM FOR SOLVING GATE ASSIGNMENT PROBLEM

A. EXPERIMENT DATA AND EXPERIMENT ENVIRONMENT A

Gate assignment problem is a combinatorial optimization problem with multi-objective properties [71]–[73]. The better gate assignment is beneficial to the perfect combination of safety and efficiency. The service satisfaction for passengers is an important operation index, so the shortest walk distances of passengers is selected as the optimization objective function. The robustness realization is the balanced idle time for each gate in order to make the personnel and equipment with the relatively balanced work time and ensure the smooth work progress, so the most balanced idle time for each gate is selected as the optimization objective function. The apron is generally far away from the terminal, the flights need to rely on taxiing for seriously affecting the satisfaction of passengers. Therefore, the least flights to the apron are selected as the optimization objective function. In summary, the shortest walk distances of passengers, the most balanced idle time for

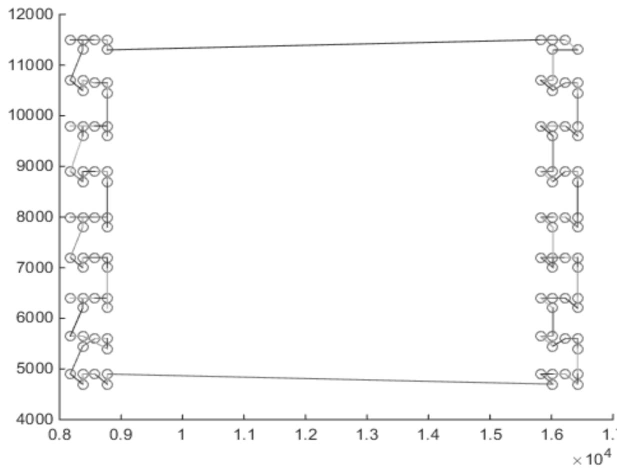


FIGURE 6. The optimal path of pr107(45649).

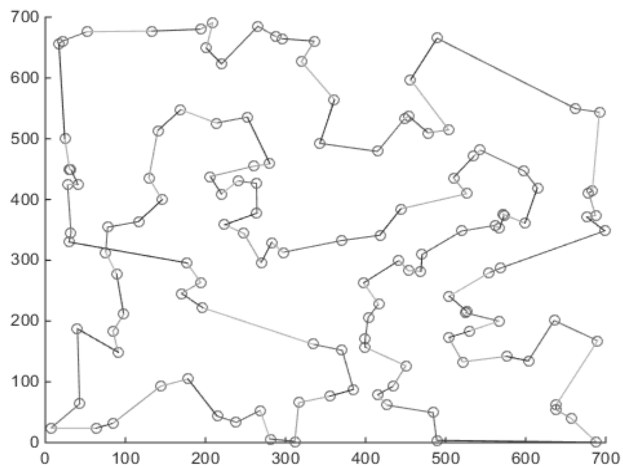


FIGURE 7. The optimal path of ch130(6183.4).

each gate and the least flights to the apron are selected as the optimization objectives for constructing a multi-objective optimization model, which is carried out the non-quantized processing to obtain the normalized objective function as follow.

$$F=\mu_1[\sum_{i=1}^{n}\sum_{k=1}^{m}S_{\mathrm{ik}}^2+\sum_{k=1}^{m}SS_k^2]+\mu_2\sum_{i=1}^{n}\sum_{k=1}^{m}q_{ik}f_{\mathrm{k}}\mathrm{y}_{ik}+\mu_3\sum_{\mathrm{k}=1}^{\mathrm{m}}g_{\mathrm{i}} \tag{19}$$

The experiment data came from Guangzhou Baiyun airport of China on July 26, 2015. There are 20 available gates and 158 flights. The gates are divided into large gate, medium gate and small gate according to the size of the available aircraft, and the flights are also divided into large flight, medium flight and small flight. All flights that are not assigned to the gates only can park to the apron. The information of gates is described in Table 3. The information of flights is described in Table 4. The experiment environments are: Matlab2016b, the Pentium CPU i7, 8.0GB RAM with Windows10.

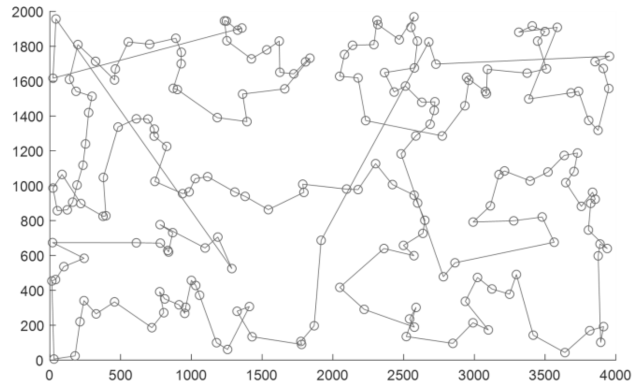


FIGURE 8. The optimal path of kroA200 (31267).

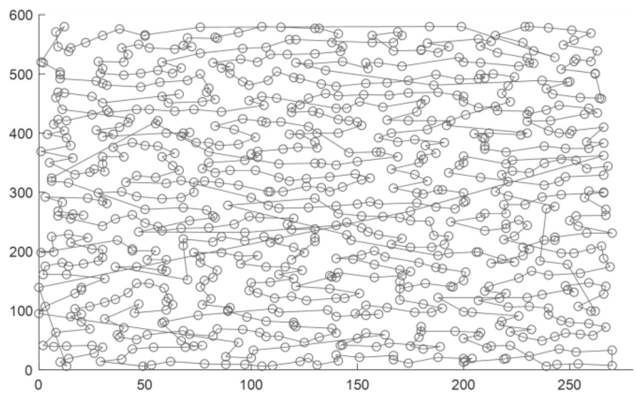


FIGURE 9. The optimal path of rat783(9229).

TABLE 3. The information of gates.

Gates	Type	Distance(m)
G1	S	190
G2	M	260
G3	L	400
G4	S	333
G5	M	384
G6	L	135
G7	S	440
G8	M	150
G9	L	230
G10	L	115
G11	M	215
G12	S	535
G13	L	170
G14	L	585
G15	M	450
G16	L	920
G17	L	1000
G18	M	426
G19	S	265
G20	L	1300

B. EXPERIMENTAL RESULT

The ICMPACO algorithm is used to solve the gate assignment model. The experiments were continuously carried out

TABLE 4. The information of flights.

Flights	Arrival time	Departure time	Passengers	Type
F1	0:05:00	7:15:00	482	M
F2	0:05:00	7:45:00	273	S
F3	0:20:00	10:30:00	312	M
F4	0:25:00	1:20:00	340	M
F5	0:30:00	10:00:00	198	S
F6	0:35:00	8:10:00	184	S
F7	0:45:00	6:40:00	500	L
F8	0:50:00	7:35:00	298	M
F9	0:55:00	1:40:00	140	S
F10	0:55:00	12:30:00	730	L
⋮	⋮	⋮	⋮	⋮
F157	22:50:00	32:20:0	212	S
F158	22:55:00	32:15:00	378	M

20 times. The best one time in 20 times was selected to analyze in here. The obtained assignment result is shown Table 5.

As can be seen from Table 5, there are 132 flights, which are assigned to 20 gates. And there are 26 flights, which are parked to the apron. The assigned efficiency is 83.5%. From the number of assigned result for each gate, the number of flights are more balanced for each gate. There are 16 gates, which have been assigned 5 flights or more 5 flights. The Gate 13 parks 11 flights, Gate 6 parks 10 flights, Gate 1, Gate 4, Gate 9 and Gate 10 park 9 flights, Gate 3, Gate 8 and Gate 18 park 7 flights. There are 4 gates, which park 6 flights for each gate. There are 2 gates, which park 5 flights for each gate. There are 3 gates, which gates park 4 flights for each gate. As can be seen from the comprehensive result of gate assignment, the proposed ICMPACO algorithm can fast and effectively assign these flights to 20 gates or the apron, and obtain the ideal assignment result. Therefore, this proposed ICMPACO algorithm takes on better optimization performance in solving gate assignment problem, and has more effective searching ability.

C. COMPARISON AND ANALYSIS OF RESULTS

In order to further demonstrate the optimization performance of the proposed ICMPACO algorithm, the basic ACO algorithm and similarity ACO(IACO) algorithm are used to solve the gate assignment problem. These parameters are set the same in Table 1. The experiments were carried out for 5 consecutive simulations. The calculation and comparison results are shown Table 6 and Figure 10.

As can be seen from Table 6 and Figure 10., the basic ACO algorithm is used to solve the gate assignment problem, the best assignment result is to assign 124 flights to gates, and the average assignment result is to assign 121 flights to gates and the average running time is 10.3252s. The IACO algorithm is used to solve the gate assignment problem, the best assignment result is to assign 127 flights to gates, and the average assignment result is to assign 123.2 flights to gates and the average running time is 17.9454s. The ICMPACO algorithm is used to solve the gate assignment problem,

TABLE 5. The gate assignment results.

Gate	Flights											Total
G1	F15	F31	F42	F56	F68	F82	F94	F121	F156			9
G2	F12	F26	F39	F100	F114	F141						6
G3	F7	F27	F40	F50	F62	F93	F102					7
G4	F16	F19	F24	F28	F34	F46	F60	F72	F146			9
G5	F5	F77	F95	F113	F135							5
G6	F3	F47	F57	F75	F80	F96	F111	F123	F138	F149		10
G7	F9	F20	F38	F65	F76	F97						6
G8	F17	F21	F29	F41	F132	F143	F158					7
G9	F6	F30	F36	F48	F58	F78	F90	F133	F140			9
G10	F4	F18	F22	F32	F43	F53	F69	F84	F131			9
G11	F23	F85										2
G12	F14	F51	F70	F86								4
G13	F1	F33	F44	F54	F71	F87	F98	F115	F122	F134	F144	11
G14	F11	F35	F45	F55	F130	F142						6
G15	F2	F112	F116	F128								4
G16	F10	F66	F92	F99	F124	F147						6
G17	F13	F37	F49	F59	F81	F101						6
G18	F8	F52	F67	F103	F117	F136	F151					7
G19	F25	F118	F125	F137	F157							5
G20	F61	F79	F104	F154								4
Total												132

TABLE 6. The calculation and comparison results.

Times	The basic ACO algorithm		The IACO algorithm		The ICMPACO algorithm	
	Assigned Flights	Running time(s)	Assigned Flights	Running time(s)	Assigned Flights	Running time(s)
1	118	9.6483	123	15.1462	131	30.5634
2	123	10.2847	122	17.3329	132	33.2903
3	119	10.8602	119	16.4627	129	28.8704
4	124	9.7681	127	18.1606	127	35.0641
5	121	11.0645	125	17.6245	131	29.9689
Average value	**121**	**10.3252**	**123.2**	**17.9454**	**130**	**31.5514**

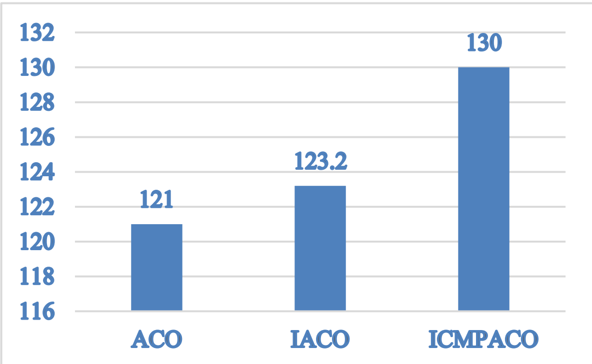


FIGURE 10. The comparison results for average value.

the best assignment result is to assign 132 flights to gates, and the average assignment result is to assign 120 flights to gates and the average running time is 31.5514s. Therefore, for the basic ACO algorithm, the IACO algorithm and the ICMPACO algorithm in solving the gate assignment problem, the best assignment result and the average assignment result of the ICMPACO algorithm is better than those of the basic ACO algorithm and the IACO algorithm. That's to say, the solution quality is the best by using the ICMPACO algorithm. But from the experiment results, we can see that time complexity of the ICMPACO algorithm is worse than time complexity of the basic ACO algorithm and the IACO algorithm.

In general, although the ICMPACO algorithm uses more time to solve gate assignment problem, the solution quality of the ICMPACO algorithm has been improved by comparing the solution quality of the basic ACO algorithm and IACO algorithm. The ICMPACO algorithm can effectively improve the comprehensive optimization performance of gate assignment problem. Therefore, the proposed ICMPACO algorithm takes on the ability to escape the local minimum value and improve the global search ability. It can effectively provide a valuable reference for assigning the gates.

VII. CONCLUSION AND FUTURE WORK

In this paper, a new multi-population co-evolution ant colony optimization (ICMPACO) algorithm based on combining the multi-population strategy, the co-evolution mechanism, the pheromone updating strategy and pheromone diffusion mechanism is proposed to solve the large-scale complex optimization problems. And the optimization performance of the ICMPACO algorithm is compared with the basic ACO algorithm and the IACO algorithm in solving the traveling salesmen problem and gate assignment problem. The proposed ICMPACO algorithm can obtain the best optimization value in solving these TSP standard examples and the gate assignment problem. It can assign 132 flights to 20 gates and the assigned efficiency reaches 83.5%, and fast obtain the ideal gate assignment result. Therefore, the proposed ICMPACO algorithm takes on better optimization ability and stability than the ACO algorithm and IACO algorithm.

Because the ICMPACO algorithm exists the longer computation time in solving complex optimization problem, the ICMPACO algorithm need to further be studied in order to reduce the time complexity. In the future work, the ICMPACO algorithm will be studied deeply.

REFERENCES

[1] M. Dorigo, V. Maniezzo, and A. Colorni, "Ant system: Optimization by a colony of cooperating agents," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 26, no. 1, pp. 29–41, Feb. 1996.
[2] S. Jiang, M. Lian, C. Lu, Q. Gu, S. Ruan, and X. Xie, "Ensemble prediction algorithm of anomaly monitoring based on big data analysis platform of open-pit mine slope," *Complexity*, vol. 2018, Aug. 2018, Art. no. 1048756. doi: 10.1155/2018/1048756.
[3] Q.-K. Pan, "An effective co-evolutionary artificial bee colony algorithm for steelmaking-continuous casting scheduling," *Eur. J. Oper. Res.*, vol. 250, no. 3, pp. 702–714, 2016.
[4] J. Luo, H. L. Chen, Q. Zhang, Y. T. Xu, H. Huang, and X. H. Zhao, "An improved grasshopper optimization algorithm with application to financial stress prediction," *Appl. Math. Model.*, vol. 64, pp. 654–668, Dec. 2018.
[5] B. Cao, J. Zhao, P. Yang, P. Yang, X. Liu, and Y. Zhang, "3D deployment optimization for heterogeneous directional wireless sensor networks on smart city," *IEEE Trans. Ind. Informat.*, to be published. doi: 10.1109/TII.2018.2884951.
[6] L. Shen *et al.*, "Evolving support vector machines using fruit fly optimization for medical data classification," *Knowl.-Based Syst.*, vol. 96, pp. 61–75, Mar. 2016.
[7] J. Wen, Z. F. Zhong, Z. Zhang, L. K. Fei, Z. H. Lai, and R. Z. Chen, "Adaptive locality preserving regression," *IEEE Trans. Circuits Syst. Video Technol.*, to be published. doi: 10.1109/TCSVT.2018.2889727.
[8] Z. R. Ren, R. Skjetne, and Z. Gao, "A crane overload protection controller for blade lifting operation based on model predictive control," *Energies*, vol. 12, no. 1, p. 50, 2019.

[9] Q. Zhang, H. L. Chen, J. Luo, Y. T. Xu, C. Wu, and C. Li, "Chaos enhanced bacterial foraging optimization for global optimization," *IEEE Access*, vol. 6, pp. 64905–64919, 2018.

[10] W. Deng, R. Chen, B. He, Y. Q. Liu, L. F. Yin, and J. H. Guo, "A novel two-stage hybrid swarm intelligence optimization algorithm and application," *Soft Comput.*, vol. 16, no. 10, pp. 1707–1722, 2012.

[11] S. K. Guo, R. Chen, H. Li, J. Gao, and Y. Q. Liu, "Crowdsourced Web application testing under real-time constraints," *Int. J. Softw. Eng. Knowl. Eng.*, vol. 28, no. 6, pp. 751–779, 2018.

[12] H. M. Zhao, R. Yao, L. Xu, Y. Yuan, G. Y. Li, and W. Deng, "Study on a novel fault damage degree identification method using high-order differential mathematical morphology gradient spectrum entropy," *Entropy*, vol. 20, no. 9, p. 682, 2018.

[13] Z. R. Ren, R. Skjetne, Z. Y. Jiang, Z. Gao, and A. S. Verma, "Integrated GNSS/IMU hub motion estimator for offshore wind turbine blade installation," *Mech. Syst. Signal Process.*, vol. 123, pp. 222–243, May 2019.

[14] Q. Zhang *et al.*, "The mono (catecholamine) derivatives as iron chelators: Synthesis, solution thermodynamic stability and antioxidant properties research," *Roy. Soc. Open Sci.*, vol. 5, no. 6, 2018, Art. no. 171492.

[15] H. M. Zhao, M. Sun, W. Deng, and X. H. Yang, "A new feature extraction method based on EEMD and multi-scale fuzzy entropy for motor bearing," *Entropy*, vol. 19, no. 1, p. 14, 2017.

[16] J. M. Nilakantan, Z. X. Li, Q. H. Tang, and P. Nielsen, "Multi-objective co-operative co-evolutionary algorithm for minimizing carbon footprint and maximizing line efficiency in robotic assembly line systems," *J. Cleaner Prod.*, vol. 156, pp. 124–136, Jul. 2017.

[17] T. Li, Z. Hu, Y. Jia, J. Wu, and Y. Zhou, "Forecasting crude oil prices using ensemble empirical mode decomposition and sparse Bayesian learning," *Energies*, vol. 11, no. 7, pp. 1–23, 2018.

[18] S. K. Guo, R. Chen, M. M. Wei, H. Li, and Y. Q. Liu, "Ensemble data reduction techniques and multi-RSMOTE via fuzzy integral for bug report classification," *IEEE Access*, vol. 6, pp. 5934–5950, 2018.

[19] W. Deng, H. M. Zhao, L. Zou, G. Y. Li, X. H. Yang, and D. Q. Wu, "A novel collaborative optimization algorithm in solving complex optimization problems," *Soft Comput.*, vol. 21, no. 15, pp. 4387–4398, 2017.

[20] D. Q. Wu, J. Z. Huo, G. F. Zhang, and W. H. Zhang, "Minimization of logistics cost and carbon emissions based on quantum particle swarm optimization," *Sustainability*, vol. 10, no. 10, p. 3791, 2018.

[21] W. Deng, R. Yao, H. Zhao, X. Yang, and G. Li, "A novel intelligent diagnosis method using optimal LS-SVM with improved PSO algorithm," *Soft Comput.*, pp. 1–18, Nov. 2017. doi: 10.1007/s00500-017-2940-9.

[22] L. Kang, H. L. Du, H. Zhang, and W. L. Ma, "Systematic research on the application of steel slag resources under the background of big data," *Complexity*, vol. 2018, Oct. 2018, Art. no. 6703908. doi: 10.1155/2018/6703908.

[23] W. J. Yu, X. B. Li, H. Yang, and B. Huang, "A multi-objective metaheuristics study on solving constrained relay node deployment problem in WSNS," *Intell. Automat. Soft Comput.*, to be published. doi: 10.1080/10798587.2017.1294873.

[24] F. Zhang *et al.*, "Real-time Calibration and registration method for indoor scene with joint depth and color camera," *J. Pattern Recognit. Artif. Intell.*, vol. 32, no. 7, 2018, Art. no. 1854021.

[25] Y. Zhou, T. Li, J. Shi, and Z. Qian, "A CEEMDAN and XGBOOST-based approach to forecast crude oil prices," *Complexity*, vol. 2019, Feb. 2019, Art. no. 4392785.

[26] Z. Z. Liu, D. H. Chu, C. Song, X. Xue, and B. Y. Lu, "Social learning optimization (SLO) algorithm paradigm and its application in QoS-aware cloud service composition," *Inf. Sci.*, vol. 326, pp. 315–333, Jan. 2016.

[27] F. Q. Zhang, Z. W. Wang, J. Chang, J. Zhang, and F. Tian, "A fast framework construction and visualization method for particle-based fluid," *EUPASIP J. Image Video Process.*, vol. 2017, no. 1, p. 79, 2017.

[28] L. Wang, X. Xu, G. Liu, B. Chen, and Z. Chen, "A new method to estimate wave height of specified return period," *J. Oceanol. Limnol.*, vol. 35, no. 5, pp. 1002–1009, 2017.

[29] M. Goran and G. G. Tihana, "Co-evolutionary multi-population genetic programming for classification in software defect prediction: An empirical case study," *Appl. Soft Comput. J.*, vol. 55, pp. 331–351, Jun. 2017.

[30] J. C. Zhou, Z. X. Du, Y. H. Liao, and A. H. Tang, "An optimization design of vehicle axle system based on multiobjective cooperative optimization algorithm," *J. Chin. Inst. Eng.*, vol. 41, no. 8, pp. 635–642, 2018. doi: 10.1080/02533839.2018.1534559.

[31] W. Deng, S. Zhang, H. Zhao, and X. Yang, "A novel fault diagnosis method based on integrating empirical wavelet transform and fuzzy entropy for motor bearing," *IEEE Access*, vol. 6, pp. 35042–35056, 2018.

[32] A. M. Yang, X. L. Yang, J. C. Chang, B. Bai, F. B. Kong, and Q. B. Ran, "Research on a fusion scheme of cellular network and wireless sensor for cyber physical social systems," *IEEE Access*, vol. 6, pp. 18786–18794, 2018.

[33] W. Ding, Z. Guan, Q. Shi, and J. Wang, "A more efficient attribute self-adaptive co-evolutionary reduction algorithm by combining quantum elitist frogs and cloud model operators," *Inf. Sci.*, vol. 293, pp. 214–234, Feb. 2015.

[34] Y. Q. Liu, X. K. Yi, Z. G. Zhai, and J. X. Gu, "Feature extraction based on information gain and sequential pattern for English question classification," *IET Softw.*, vol. 12, no. 6, pp. 520–526, 2018. doi: 10.1049/iet-sen.2018.0006.

[35] Y. Peng and B. L. Lu, "Discriminative extreme learning machine with supervised sparsity preserving for image classification," *Neurocomputing*, vol. 261, pp. 242–252, Oct. 2017.

[36] F. Huang, C. Yao, W. Liu, Y. Li, and X. Liu, "Landslide susceptibility assessment in the Nantian area of China: A comparison of frequency ratio model and support vector machine," *Geomatics, Natural Hazards Risk*, vol. 9, no. 1, pp. 919–938, 2018.

[37] F. R. Sun, Y. D. Yao, and X. F. Li, "The heat and mass transfer characteristics of superheated steam coupled with non-condensing gases in horizontal wells with multi-point injection technique," *Energy*, vol. 143, pp. 995–1005, Jan. 2018.

[38] S. K. Guo, Y. Q. Liu, R. Chen, X. Sun, and X. X. Wang, "Improved SMOTE algorithm to deal with imbalanced activity classes in smart homes," *Neural Process. Lett.*, pp. 1–24, Oct. 2018. doi: 10.1007/s11063-018-9940-3.

[39] J. Su, Z. Sheng, L. B. Xie, G. Li, and A. X. Liu, "Fast splitting based tag identification algorithm for anti-collision in UHF RFID system," *IEEE Trans. Commun.*, to be published. doi: 10.1109/TCOMM.2018.2884001.

[40] L. dos Santos Coelho and D. L. de Andrade Bernert, "A modified ant colony optimization algorithm based on differential evolution for chaotic synchronization," *Expert Syst. Appl.*, vol. 37, no. 6, pp. 4198–4203, 2010.

[41] J. Rada-Vilela, M. Chica, Ó. Cordón, and S. Damas, "A comparative study of multi-objective ant colony optimization algorithms for the time and space assembly line balancing problem," *Appl. Soft Comput.*, vol. 13, no. 11, pp. 4370–4382, 2013.

[42] Y.-J. He and Z.-F. Ma, "Optimal design of linear sensor networks for process plants: A multi-objective ant colony optimization approach," *Chemometrics Intell. Lab. Syst.*, vol. 135, pp. 37–47, Jul. 2014.

[43] C. F. Juang, C. W. Hung, and C. H. Hsu, "Rule-based cooperative continuous ant colony optimization to improve the accuracy of fuzzy system design," *IEEE Trans. Fuzzy Syst.*, vol. 22, no. 4, pp. 723–735, Aug. 2014.

[44] J. Y. Yang, R. F. Ding, Y. Zhang, M. Q. Cong, F. Wang, and G. A. Tang, "An improved ant colony optimization (I-ACO) method for the quasi travelling salesman problem (Quasi-TSP)," *Int. J. Geograph. Inf. Sci.*, vol. 29, no. 9, pp. 1534–1551, 2015.

[45] P. B. Myszkowski, M. E. Skowroński, Ł. P. Olech, and P. O. Oślizło, "Hybrid ant colony optimization in solving multi-skill resource-constrained project scheduling problem," *Soft Comput.*, vol. 19, no. 12, pp. 3599–3619, 2015.

[46] I. D. I. D. Ariyasingha and T. G. I. Fernando, "Performance analysis of the multi-objective ant colony optimization algorithms for the traveling salesman problem," *Swarm Evol. Comput.*, vol. 23, pp. 11–26, Aug. 2015.

[47] 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 Eng.*, vol. 102, pp. 63–70, Jul. 2015.

[48] L. Y. Zuo, L. Shu, S. B. Dong, C. Zhu, and T. Hara, "A multi-objective optimization scheduling method based on the ant colony algorithm in cloud computing," *IEEE Access*, vol. 3, pp. 2687–2699, 2015.

[49] J. Bagherinejad and M. Dehghani, "A multi-objective robust optimization model for location-allocation decisions in two-stage supply chain network and solving it with non-dominated sorting ant colony optimization," *Sci. Iranica*, vol. 22, no. 6, pp. 2604–2620, 2015.

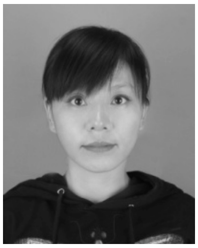
[50] K. Krynicki, M. E. Houle, and J. Jaen, "An efficient ant colony optimization strategy for the resolution of multi-class queries," *Knowl.-Based Syst.*, vol. 105, pp. 96–106, Aug. 2016.

[51] S. B. Chen, G. Lv, and X. F. Wang, "Offensive strategy in the 2D soccer simulation league using multi-group ant colony optimization," *Int. J. Adv. Robot. Syst.*, vol. 13, no. 1, p. 25, 2016.

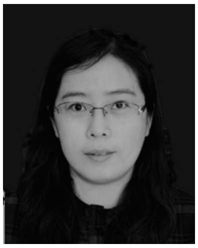
[52] M. Marzband, E. Yousefnejad, and A. Sumper, "Real time experimental implementation of optimum energy management system in standalone microgrid by using multi-layer ant colony optimization," *Int. J. Elect. Power Energy Syst.*, vol. 75, pp. 265–274, Feb. 2016.

[53] A. Khan and A. R. Baig, "Multi-objective feature subset selection using mRMR based enhanced ant colony optimization algorithm (mRMR-EACO)," *J. Exp. Theor. Artif. Intell.*, vol. 28, no. 6, pp. 1061–1073, 2016.
[54] Y. G. Zhong and B. Ai, "A modified ant colony optimization algorithm for multi-objective assembly line balancing," *Soft Comput.*, vol. 21, no. 22, pp. 6881–6894, 2017.
[55] R.-H. Huang and T.-H. Yu, "An effective ant colony optimization algorithm for multi-objective job-shop scheduling with equal-size lot-splitting," *Appl. Soft Comput.*, vol. 57, pp. 642–656, Aug. 2017.
[56] J. Z. Zhou *et al.*, "A multi-objective multi-population ant colony optimization for economic emission dispatch considering power system security," *Appl. Math. Model.*, vol. 45, pp. 684–704, May 2017.
[57] P. Vijayalakshmi, S. A. J. Francis, and J. A. Dinakaran, "A robust energy efficient ant colony optimization routing algorithm for multi-hop ad hoc networks in MANETs," *Wireless Netw.*, vol. 22, no. 6, pp. 2081–2100, 2016.
[58] P. K. Tiwari and D. P. Vidyarthi, "Improved auto control ant colony optimization using lazy ant approach for grid scheduling problem," *Future Gener. Comput. Syst.*, vol. 60, pp. 78–89, Jul. 2016.
[59] H. Sharifipour, M. Shakeri, and H. Haghighi, "Structural test data generation using a memetic ant colony optimization based on evolution strategies," *Swarm Evol. Comput.*, vol. 40, pp. 76–91, Jun. 2018.
[60] K. Akka and F. Khaber, "Mobile robot path planning using an improved ant colony optimization," *Int. J. Adv. Robot. Syst.*, vol. 15, no. 3, pp. 1–7, 2018.
[61] C. C. Chen and L. P. Shen, "Improve the accuracy of recurrent fuzzy system design using an efficient continuous ant colony optimization," *Int. J. Fuzzy Syst.*, vol. 20, no. 3, pp. 817–834, 2018.
[62] J. Wen, N. Han, X. Fang, L. Fei, K. Yan, and S. Zhan, "Low-rank preserving projection via graph regularized reconstruction," *IEEE Trans. Cybern.*, to be published. doi: 10.1109/TCYB.2018.2799862.
[63] S. Zhang, W. Shen, D. Li, X. Zhang, and B. Chen, "Nondestructive ultrasonic testing in rod structure with a novel numerical Laplace based wavelet finite element method," *Latin Amer. J. Solids Struct.*, vol. 15, no. 7, pp. 1–17, 2018.
[64] J. Wen, Y. Xu, Z. Li, Z. Ma, and Y. Xu, "Inter-class sparsity based discriminative least square regression," *Neural Netw.*, vol. 102, pp. 36–47, Jun. 2018.
[65] H. Fu, Z. Li, Z. Liu, and Z. Wang, "Research on big data digging of hot topics about recycled water use on micro-blog based on particle swarm optimization," *Sustainability*, vol. 10, no. 7, p. 2488, 2018.
[66] W. Yu *et al.*, "Multi-objective optimum design of high-speed backplane connector using particle swarm optimization," *IEEE Access*, vol. 6, pp. 35182–35193, 2018.
[67] J. Wen *et al.*, "Robust sparse linear discriminant analysis," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 2, pp. 390–403, Feb. 2018. doi: 10.1109/TCSVT.2018.2799214.
[68] B. Chen, G. Liu, L. Wang, K. Zhang, and S. Zhang, "Determination of water level design for an estuarine city," *J. Oceanol. Limnol.*, vol. 37, no. 1, pp. 20–29, 2018.
[69] A. E. Yildirim and A. Karci, "Applications of artificial atom algorithm to small-scale traveling salesman problems," *Soft Comput.*, vol. 22, no. 22, pp. 7619–7631, 2018.
[70] M. Cárdenas-Montes, "Creating hard-to-solve instances of travelling salesman problem," *Appl. Soft Comput.*, vol. 71, pp. 268–276, Oct. 2018.
[71] W. Deng, H. M. Zhao, X. H. Yang, J. Xiong, M. Sun, and B. Li, "Study on an improved adaptive PSO algorithm for solving multi-objective gate assignment," *Appl. Soft Comput.*, vol. 59, pp. 288–302, Oct. 2017.
[72] C. Yu, D. Zhang, and H. Y. K. Lau, "An adaptive large neighborhood search heuristic for solving a robust gate assignment problem," *Expert Syst. Appl.*, vol. 84, pp. 143–154, Oct. 2017.
[73] W. Deng, H. Zhao, X. Yang, D. Li, Y. Li, and J. Liu, "Research on a robust multi-objective optimization model of gate assignment for hub airport," *Transp. Lett.*, vol. 10, no. 4, pp. 229–241, 2018.

WU DENG (M'76) received the B.S. degree in electrical technology, and the M.S. degree in computer application technology from Dalian Jiaotong University, Dalian, China, in 2001 and 2006, respectively, and the Ph.D. degree in computer application technology from Dalian Maritime University, Dalian, in 2012. Since 2019, he has been a Professor with the College of Electronic Information and Automation, Civil Aviation University of China, Tianjin, China. His research interests include artificial intelligence, optimization method, and fault diagnosis.



JUNJIE XU received the B.S. and M.S. degrees in computer science and technology from Dalian Maritime University, Dalian, China, in 2007 and 2010, respectively, and the Ph.D. degree in computer application technology from Dalian Maritime University, Dalian, in 2016. Since 2016, she has been a Lecturer with the College of Computer Science and Technology, Civil Aviation University of China, Tianjin, China. Her research interest includes artificial intelligence and information safety.



HUIMIN ZHAO received the B.S. degree in electrical technology, and the M.S. degree in traffic information engineering and control from Dalian Jiaotong University, Dalian, China, in 2000 and 2004, respectively, and the Ph.D. degree in mechanical engineering from Dalian Maritime University, Dalian, in 2013. Since 2019, she has been a Professor with the College of Electronic Information and Automation, Civil Aviation University of China, Tianjin, China. Her research interests include artificial intelligence, signal processing, and fault diagnosis.