IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received June 30, 2020, accepted July 12, 2020, date of publication July 15, 2020, date of current version July 24, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3009399

Dynamic Multi-Role Adaptive Collaborative Ant Colony Optimization for Robot Path Planning

DEHUI ZHANG [1], XIAOMING YOU [1], SHENG LIU [2], AND HAN PAN [1]
[1]College of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai 201620, China
[2]School of Management, Shanghai University of Engineering Science, Shanghai 201620, China

Corresponding author: Xiaoming You (yxm6301@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61673258 and Grant 61075115, and in part by the Shanghai Natural Science Foundation under Grant 19ZR1421600.

ABSTRACT Aiming at the problems of poor diversity and slow convergence of ant colony algorithm, dynamic multi-role adaptive collaborative ant colony optimization (MRCACO) is proposed in this paper, and it applies to robot path planning. Firstly, an adaptive dynamic complementary algorithm is proposed to form a heterogeneous multi-colony together with ACS and MMAS, which complement each other in performance. Secondly, a multi-role adaptive cooperation mechanism is proposed to realize the exchange and sharing of information. The mechanism includes two strategies: one is an elite attribute learning strategy, which highlights the role of elite attribute and improves the comprehensive performance of ACS and MMAS; The second is the pheromone balancing strategy, which is executed when the algorithm is stagnant to make the algorithm jump out of the local optimal. Further, the effectiveness and superiority in the algorithm are demonstrated by the experimental analysis of multiple TSP instances. Finally, the algorithm presented in this paper is applied to the path planning of the robot, two different deadlock rollback strategies are proposed to solve the deadlock problem and improve the efficiency of the algorithm. The results of a practical application show that the algorithm is feasible to solve the path planning problem.

INDEX TERMS Multi-colony ant colony optimization, path planning, adaptive dynamic complementary algorithm, multi-role adaptive collaborative mechanism, deadlock rollback.

I. INTRODUCTION

Traveling Salesman Problem (TSP) is a classical combinatorial optimization problem, which refers to the shortest path problem in which a traveler starts from a certain starting point, passes through all given demand points, each demand point only passes once, and finally returns to the starting point. Path planning refers to finding an appropriate path from the beginning to the end to avoid all obstacles in the process of movement according to certain evaluation criteria in the environment with obstacles. The final result of the traveling salesman problem is the shortest loop path, while the final result of the robot path planning is the shortest line segment path. At present, intelligent algorithms to solve these two kinds of problems include particle swarm optimization [1], [2], genetic algorithm [3], ant colony algorithm, etc.

Ant Colony Optimization (ACO) [4] was first proposed by Italian scholar Marco Dorigo in 1996 as an algorithm to solve travel agents and distributed Optimization problems based on the ant foraging mechanism. In 1997, Dorigo proposed the Ant Colony System (ACS) [5], which added the global pheromone updating mechanism based on the Ant System and improved the convergence speed of the algorithm. In 2000, Stutzle proposed the Max-min Ant System (MMAS) [6], he proposed that by limiting the range of pheromones, the gap between pheromones in each path would be reduced and the diversity of the algorithm would be improved. The above are classical ant colony algorithms, which have high efficient searching ability, but there are still some problems such as easy to fall into local optimization and slow convergence speed.

To solve the problem of poor diversity and slow convergence of the traditional ant colony algorithm, scholars began to improve the single-colony ant colony algorithm, generally, the improvement direction is parameter optimization [7] and the combination with other algorithms [8], [9]. However, due to the limitation of mechanism, the improvement of single-colony tends to focus on the optimization of unilateral

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

performance, and most of them adopt the method of improving different characteristics in different periods. It is difficult to achieve a satisfactory balance between diversity and convergence. On this basis, the researchers set out to discuss the improvement of the multi-colony algorithm, through the role assignment of each subpopulation, intraspecies competition and interspecies cooperation to balance the convergence and diversity of the algorithm, and improve its performance.

Multi-colony ant colony optimization was originally proposed by Gambardella to solve the time window problem of vehicle paths. Xu proposed a heterogeneous two-colony ant colony optimization based on heuristic information for TSP solution, introduced exchange factors, and carried out information exchange regularly, balancing the convergence and diversity of the algorithm in large-scale problems, but the adaptability of the algorithm still needs to be improved [10]. He proposed a two-colony ant colony algorithm, which improved the diversity of solutions through heterogeneous evolution and information exchange, as the alternating current frequency was related to the number of iterations, the algorithm pattern was relatively fixed [11].

To sum up, the multi-colony algorithm is relatively simple in construction, the communication mode is mostly pheromone matrix exchange and the communication frequency is also fixed, which fails to fully reflect the role of each subpopulation, and the self-adaptability of the algorithm needs to be improved. To solve these problems, scholars have introduced the principle of information entropy to improve the adaptability of the algorithm and make the algorithm more rigorous. Reference [12] proposed an improved artificial bee colony algorithm based on information entropy, which used the value of information entropy to assess uncertainty, and the selection process of following peaks was judged based on information entropy. In reference [13], an adaptive double-colony ant colony algorithm with entropy is proposed, the ant colony is divided into red ant colony and black ant colony by using information entropy, and the number of red and black ants is determined by entropy value so that the algorithm achieves self-adaptability. Reference [14] proposed a multi-colony ant colony algorithm for virtual machine deployment, which determined the information exchange strategy among Ant colonies according to the information entropy of each colony to ensure the balance between convergence and diversity. In the above improved algorithm, the application of information entropy provides theoretical support for the index measurement in the algorithm process, improves the adaptability of the algorithm, and makes the algorithm more rigorous.

Ant colony optimization will eventually be applied to practical problems. This paper will improve the ant colony algorithm to solve the robot path planning problem. In reference [15], an application of an ant colony algorithm based on a negative feedback mechanism in robot path planning is proposed, the negative feedback mechanism is used to improve the diversity of solutions to obtain the optimal path. In reference [16], an improved ant colony algorithm was proposed to solve the path planning problem, and the performance of the algorithm was improved by combining pheromone diffusion with geometric local optimization. In reference [17] an improved ant colony algorithm was proposed to be applied to the planning of multi-scenic spots, this algorithm eliminated the restrictions of the taboo table of ant colony algorithm, realized partial point traversal of the connected graph, and introduced a temporary weight matrix to improve the overall efficiency of the algorithm. The above algorithm is the application of the single-colony algorithm in path planning, lack of collaboration between multi-colony.

Ant colony optimization is a parallel self-organizing algorithm with advantages of positive feedback and strong robustness. It was originally used to solve TSP problems, after years of development, it has gradually penetrated other fields, such as path planning problems, large-scale integrated circuit design, routing problems in communication networks, load balancing problems, and vehicle scheduling problems. Ant colony algorithm has become a common method to solve the problem of path planning, therefore, our research group is studying the use of ant colony optimization to solve the problem of robot path planning. This paper proposes the application of dynamic multi-role adaptive collaborative ant colony optimization in path planning to explore the impact of multi-colony cooperation.

The main contributions of this paper are as follows:

1. Firstly, an adaptive dynamic compensation algorithm (ADCA) is proposed, in which dynamic grouping strategy and pheromone diffusion mechanism exist, and the diversity and convergence of the algorithm can be adjusted through cooperative communication between different groups. Heterogeneous multi-colony is formed by ADCA, ACS, and MMAS. These three sub-populations have different characteristics, and achieve inter-population performance complementation through the mechanism in contribution 2.

2. Secondly, the multi-role adaptive cooperation mechanism is proposed as the communication mode between heterogeneous populations, to explore the influence of multi-colony information interaction on the algorithm, this mechanism contains two learning strategies: elite attribute learning strategy and pheromone balancing strategy. In the elite attribute learning strategy, the concept of population comprehensive performance was proposed to measure population performance, the subpopulation with low performance can communicate with the excellent population through elite attribute learning strategy so as to improve its comprehensive performance. In the pheromone balancing strategy, when the algorithm falls into a stagnant, the pheromone will be merged and divided among the sub-populations to increase its diversity, so that the algorithm jumps out of the local optimum.

3. Finally, the MRCACO algorithm is used to solve the path planning problem, and it is further improved. We propose two deadlock fallback strategies to solve the deadlock problem through adaptive fallback of the path, thereby improving the accuracy of the algorithm.

TABLE 1. Analyze the advantages and disadvantages of the metaheuristic algorithm.

Abbreviation	Algorithm of reference	advantages	disadvantages
SA	Simulated Annealing [18]	Simple,	Easy to fall into local optimum,
TS	Tabu Search [19]	flexible,	no adaptability.
HC	hill climbing [20]	easy to implement.	
GA	Genetic Algorithm [3]	Self-organization,	Fixed mode,
ES	Evolutionary Strategy [21]	self-adaptation,	many parameters,
EG	Evolutionary Programming [22]	self-learning.	easy to fall into local optimum.
ACO	Ant Colony Optimization [4]– [6], [10], [11], [13]– [17], [23]	High robustness,	May fall into local optimum.
PSO	Particle Swarm Optimization [1], [2]	intelligence,	
ABC	Artificial Bee Colony [12], [24]	information interaction,	
FOA	Fruit Fly Optimization Algorithm [25], [26]	group collaboration.	

The structure of this paper is as follows: Section II briefly introduces the content of metaheuristic, ACS, MMAS, and the concepts of information entropy and raster modeling. Section III introduces the specific contents of MRCACO. Section IV the MRCACO algorithm is used to solve the TSP problem and its performance is analyzed. Section V is applying the MRCACO algorithm to robot path planning. Firstly, some improvements needed to be done to MRCACO to make it more suitable; Secondly, the performance of MRCACO and improved MRCACO in path planning is analyzed. Section VI provides a summary and future work.

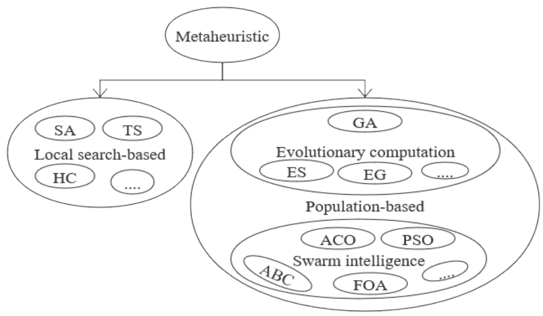


FIGURE 1. Metaheuristic algorithm.

II. RELATED WORK

A. METAHEURISTIC ALGORITHM OVERVIEW

Metaheuristic algorithm is the product of the combination of random algorithm and local search algorithm, today, metaheuristic algorithms have been successfully applied in engineering, computer networks, biological system modeling, prediction, pattern recognition, data clustering, feature selection, and other fields. Fig.1 shows some typical metaheuristic algorithms, the abbreviations in the figure are explained in Table.1, which briefly summarizes the advantages and disadvantages of different types of metaheuristic algorithms. At present, a swarm intelligence algorithm tends to solve combinatorial optimization problems. In this paper, ant colony optimization is selected to solve the robot path planning problem.

B. ANT COLONY SYSTEM

1) PATH CONSTRUCTION

In ACS, the current position of ant k is at i, and the next city to be visited j is selected according to the pseudo-random proportion rule, whose rule is shown in (1):

$$j = \begin{cases} \arg \max\limits_{j \in allowed} \{\tau_{ij} \cdot \eta_{ij}^{\beta}\} & q \leq q_0 \\ J & else \end{cases} \tag{1}$$

where q is a random variable uniformly distributed on the interval [0, 1]; q_0 is a parameter whose range is [0, 1]; J is a variable generated from the probability distribution given by (2).

$$P_{ij}(t) = \begin{cases} \dfrac{[\tau_{ij}(t)]^{\alpha}[\eta_{ij}]^{\beta}}{\sum\limits_{s \in allowed} [\tau_{is}(t)]^{\alpha}[\eta_{is}]^{\beta}} & j \in allowed \\ 0 & else \end{cases} \tag{2}$$

where α is the information heuristic factor; β is the expected heuristic factor; η_{ij} is the heuristic function, whose expression is (3):

$$\eta_{ij} = 1/d_{ij} \tag{3}$$

2) PHEROMONE UPDATE

Local pheromone update rules: When the ant makes a path selection, that is, it updates the pheromone immediately after it goes from the current city i to the next city j. The formula is shown in (4):

$$\tau_{ij} = (1 - \rho) \cdot \tau_{ij} + \rho \cdot \tau_0 \tag{4}$$

where ρ is the evaporation coefficient of local pheromone, whose range is (0, 1); τ_0 is the initial value of the pheromone.

Global pheromone update rule: The pheromone is updated after one iteration of all ants, and only the ants on the global optimal path can update the pheromone, thereby speeding up the convergence speed of the algorithm. Equation(5) is:

$$\tau_{ij} = (1 - \xi) \cdot \tau_{ij} + \xi \cdot \Delta\tau_{ij}^{bs} \tag{5}$$

$$\Delta\tau_{ij}^{bs} = 1/C^{bs} \tag{6}$$

where ξ is the global pheromone evaporation coefficient and C^{bs} is the length of the global optimal path; $\Delta\tau_{ij}^{bs}$ is the pheromone added on the global optimal path, calculated according to (6).

C. MAX-MIN ANT SYSTEM

1) PHEROMONE RANGE LIMITATION

In order to avoid fast convergence of the algorithm, the MMAS algorithm limits the pheromones of each side to a certain range $[\tau_{\min}, \tau_{\max}]$. If $\tau_{ij} \leq \tau_{\min}$, we set $\tau_{ij} = \tau_{\min}$; If $\tau_{ij} \geq \tau_{\max}$, we set $\tau_{ij} = \tau_{\max}$.

$$\tau_{\max} = (1/\rho) \cdot (1\big/T^{gb}) \tag{7}$$

$$\tau_{\min} = \tau_{\max}\big/2n \tag{8}$$

where T^{gb} is the global optimal path.

2) PHEROMONE UPDATE

When an iteration is completed, the current optimal path or the global optimal path is updated with pheromone, so that the optimal solution can be effectively utilized and the exploration ability of the algorithm is enhanced. Pheromone update rules are shown in (9) and (10):

$$\tau_{ij}(t+1) = (1-\rho)\cdot\tau_{ij}(t) + \Delta\tau_{ij}^{best} \tag{9}$$

$$\Delta\tau_{ij}^{best} = 1/f(s^{best}) \tag{10}$$

where $f(s^{best})$ is the current optimal or global optimal path.

D. INFORMATION ENTROPY

Information Entropy is a word borrowed from thermodynamics by C. E. Shannon in 1948 to solve the problem of quantitative measurement of information. It's also one of several ways to measure diversity. Entropy can be written explicitly:

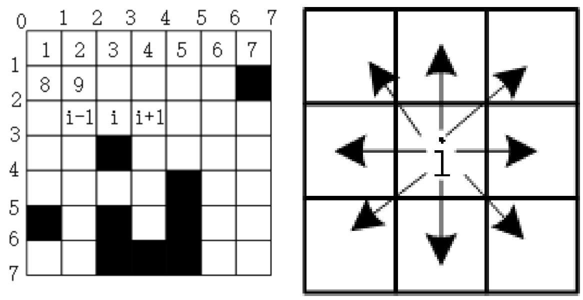
$$H(X) = -\sum_{i=1}^{n} P(x_i)\log_b P(x_i) \tag{11}$$

where b is the base of the logarithm. $P(x_i)$ is the probability mass function.

The entropy of the unknown result is maximized if each probability is fair. Therefore, this paper uses information entropy to measure the diversity of populations.

E. ENVIRONMENTAL MODELING

When the ant colony algorithm solves the problem of path planning simulation, it needs to rasterize the environment map. Raster modeling is the use of the same size raster to divide the plane environment, and the divided raster uniform number to form a raster map. This method is used to build a raster map of M*M scale (in Fig.2(a), M=7), where the white grid represents the free passable grid and the black grid represents the impassable obstacle grid. There are 8 passable directions in grid i, and the next feasible grid that can be selected in these 8 directions constitutes the adjacency node matrix of grid i (Fig.2(b)).



(a) Raster map (b) Adjacency node matrix

FIGURE 2. Grid map.

III. DYNAMIC MULTI-ROLE ADAPTIVE COLLABORATIVE ANT COLONY OPTIMIZATION

A. ADAPTIVE DYNAMIC COMPLEMENTARY ALGORITHM

In this paper, the CACS algorithm in reference [23] is regarded as an adaptive dynamic complementary algorithm (ADCA). ADCA, MMAS, and ACS constitute the heterogeneous multi-colony proposed in this paper, after ADCA was added, three sub-populations realized adaptive complementary in performance. The adaptive dynamic complementary algorithm uses a dynamic division of labor to improve ACS. The purpose of the improvement is to enable a single population to adaptively balance diversity and convergence.

1) DYNAMIC GROUPING STRATEGY

ADCA introduces a dynamic grouping rate ε to dynamically divide the m ants into two categories: There are $m(1-\varepsilon)$ search ants that play a role in maintaining diversity and $m\varepsilon$ tracking ants which play a role in ensuring convergence. The division rules are: the path is taken by the ant after each iteration is sorted in ascending order, the before $m\varepsilon$ ants corresponding to the sorted path become tracking ants, the rest are search ants. The formula of ε is (12). According to (12), each ant may dynamically change its role in each generation, enabling it to adapt balance diversity and convergence.

$$\varepsilon = iter/maxiter \tag{12}$$

where $iter$ is the number of current iterations; $maxiter$ is the maximum number of iterations.

2) PATH CONSTRUCTION RULES BASED ON DYNAMIC GROUPING

The probability rule that the ants in ADCA choose the next city is: tracking ants follow the (2) for path construction, and the search ants follow the (13) for path construction.

$$p_{ij-s}{}^{k}(t) = \begin{cases} p_{ij}^{k}(t) \times rand, f > \mu \\ p_{ij}^{k}(t), other \end{cases} \tag{13}$$

$$f = 1 - \varepsilon \tag{14}$$

where μ is the threshold value of environmental fitness factor, which is a constant between (0, 1).

3) PHEROMONE UPDATE RULES BASED ON DYNAMIC GROUPING

The local pheromone update rule of ADCA is (15), where $\Delta\tau_{ij}^{*}(t)$ is a pheromone reward that is unique to tracking ants, and that comes from pheromone diffusion. Global pheromone update rule is (5).

$$\tau_{ij}(t) = (1-\rho)\,\tau_{ij}(t) + \rho\tau_0 + \Delta\tau_{ij}^{*}(t) \tag{15}$$

B. MULTI-ROLE ADAPTIVE COLLABORATIVE MECHANISM

1) ELITE ATTRIBUTE LEARNING STRATEGY

In this paper, ACS is used to maintain convergence, MMAS is used to maintain diversity. The purpose of balancing the diversity and convergence of the algorithm is achieved through communication between the two, and the overall accuracy of the algorithm is ultimately improved. The communication mode is the elite attribute learning strategy, and learning rules are: when the diversity of ACS subpopulation is below the threshold, first k sub-populations with high diversity were selected from the sub-populations of MMAS, then select the subpopulation with the highest comprehensive performance among these k sub-populations, finally they communicate with each other. Based on such a selective communication mode, ACS can not only learn elite attributes in MMAS, but also improve the comprehensive performance of the population, and this learning style is adaptive. Similarly, when the convergence of MMAS subpopulation is below the threshold, first k subpopulation with high convergence was selected from the ACS subpopulation, then select the subpopulation with the highest comprehensive performance among these k sub-populations, finally they communicate with each other.

Equation(16) is the measurement standard for the comprehensive performance of the population. The comprehensive performance of the population includes three attribute factors: Div_i represents diversity, whose value is the ratio of the current information entropy of population i to the global maximum information entropy; Sol_i represents the strength of the solution, and its value is the ratio of the standard optimal solution of the algorithm to the current optimal solution of population i; Con_i represents convergence, and its value is the ratio of the global optimal convergent iteration number to the current optimal convergent iteration number of population i.

$$Per_i = Div_i \times Sol_i \times Con_i \tag{16}$$

Fig.3 is a schematic diagram of the elite attribute learning strategy. For example, when the convergence of the MMAS subpopulation is lower than the threshold (red box in Fig.3), the elite attribute learning strategy is implemented. In Fig.3, the yellow box is the high-quality attribute in the two sub-populations, and the population attribute after learning is composed of all the high-quality attributes of the two populations (new MMAS in Fig.3).

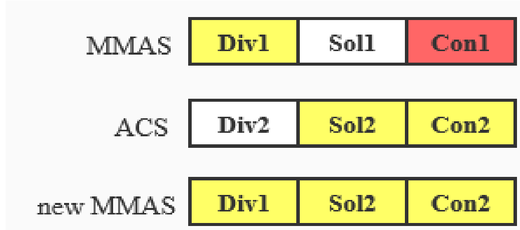


FIGURE 3. Elite attribute learning strategy.

2) PHEROMONE BALANCING STRATEGY

The pheromone balancing strategy is used between ACS, MMAS, and ADCA sub-populations. The reasons for using the pheromone balancing strategy are as follows: 1. The elite attribute learning strategy can balance the diversity and convergence by maintaining the characteristics of two sub-populations, while ADCA can ensure diversity and convergence simultaneously through itself, these two ways of balancing diversity and convergence may have some shortcomings in performance. 2. The elite attribute learning strategy and the pheromone diffusion strategy in ADCA may lead to the excessive accumulation of pheromones in some paths, so that the algorithm falls into the local optimal. The pheromone balancing strategy combines the pheromones of three sub-populations to make them complement each other in performance, and make the algorithm jump out of the local optimal.

The use time of the pheromone balancing strategy is: when the subpopulation falls into a local optimum. The use mode is: the ADCA subpopulation with good diversity and high comprehensive performance is first selected for performance complementary, then learn according to (17). The pheromone balancing strategy is to determine the execution time and the execution object according to the current environment of the algorithm, it's also adaptive.

$$P_{new} = \frac{1}{3}\left(P_{ACS} + P_{MMAS} + P_{ADCA}\right) \tag{17}$$

where P_{new} is the new pheromone matrix after the pheromone balancing of the three populations, P_{ACS}, P_{MMAS}, and P_{ADCA} are the original pheromone matrix of ACS, MMAS, and ADCA respectively.

3) THE FRAMEWORK OF MULTI-ROLE ADAPTIVE COLLABORATIVE MECHANISM

Fig.4 is the framework diagram of the multi-role adaptive collaborative mechanism. Multi-colony are divided into three sub-populations, ACS, MMAS, and ADCA, which play different roles: ACS guaranteed population convergence, MMAS guaranteed population diversity, ADCA balanced population diversity and convergence. Among them, the cooperation between ACS and MMAS achieves the purpose of balancing the diversity and convergence, and has the relationship of performance complementary with ADCA. The communication mechanism is divided into elite attribute learning strategy and pheromone balancing strat-

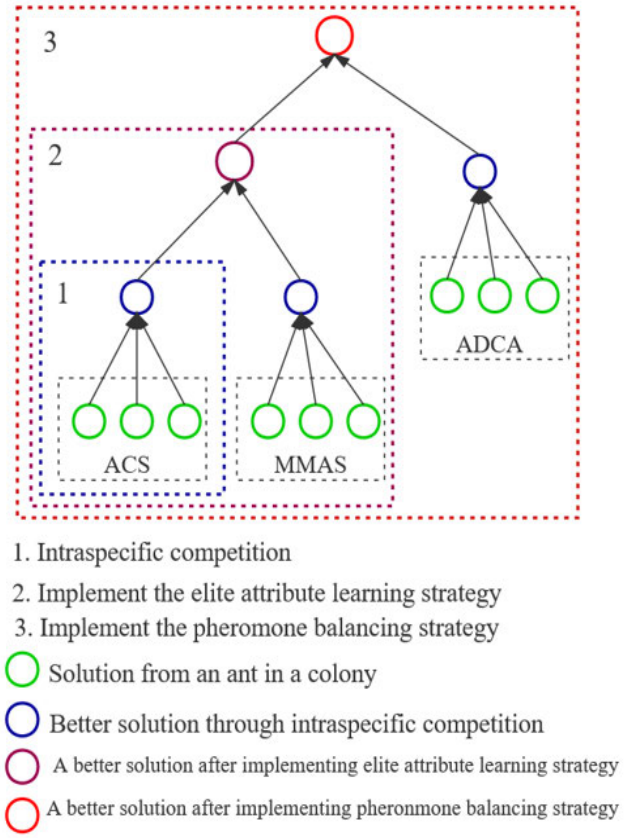


FIGURE 4. Multi-role adaptive collaborative mechanism.

egy. Dynamic adaptive collaborative learning was conducted among the three sub-populations according to the current environmental information of the algorithm, and the two communication strategies are also implemented adaptively according to the feedback of information.

C. ALGORITHM FRAMEWORK

The execution process of MRCACO algorithm is as follows:

Step1. Initialize the parameters in MRCACO, initialize the pheromone matrix and the distance between nodes.

Step2. The subpopulation of ACS, MMAS, and ADCA are respectively constructed according to their path construction methods, and their local pheromones are updated for each step of movement.

Step3. After the completion of one iteration for all subpopulations, update the global pheromone of each subpopulation to retain the current optimal solution of the whole algorithm at this time. If the current optimal solution is better than the historical optimal solution, the historical optimal solution is replaced, otherwise, it is not replaced.

Step4. When the diversity of ACS is lower than the threshold or the convergence of MMAS is lower than the threshold, implement the elite attribute learning strategy.

Step5. When the global optimal solution remains unchanged for T consecutive times, that is, the algorithm is stalled, implement the pheromone balancing strategy.

Step6. Increase the number of iterations and return to Step2.

Step7. Reach the maximum number of iterations and output the global optimal solution of the algorithm.

Algorithm 1 MRCACO Algorithm for TSP

Initialize the pheromone and the parameters
Calculate the distance between cities
while termination condition is not satisfied **do**
    Construct ant solutions for ACS,MMAS, and ADCA with (1),(2), and (2),(13)
    Update local pheromone for MMAS,ACS, and ADCA with (4),(9), and (15)
    Update global pheromones for MMAS,ACS,ADCA with (5)
    if subpopulation has a low performance **then**
        Executive elite attribute learning strategy
    end if
    if sub-populations fall into local optimum **then**
        Executive pheromone balancing strategy
    end if
    $nc = nc + 1$
end while
Return the global optimal

In this framework, the number of iteration is nc, the number of subpopulation ant is m, the number of city is n, all sub-populations run in parallel in the computer.

Through the analysis of the algorithm framework, we know that the time complexity of MRCACO is $O(nc * m * (n - 1))$, and the maximum time complexity is $O(nc*m*n)$. As we have known, the maximum time complexity of ACS and MMAS is $O(nc * m * n)$, the time complexity of ADCA is also analyzed in reference [23], which is $O(nc * m * n)$. So, the MRCACO algorithm does not increase the time complexity.

IV. PERFORMANCE ANALYSIS OF MRCACO ALGORITHM

A. SIMULATION ENVIRONMENT AND PARAMETER SETTING

To verify the performance of MRCACO, the experimental test environment in this paper is as follows: windows10 operating system, matlab2016a test simulation software. Although the MRCACO algorithm uses multiple ACS, MMAS, and ADCA subpopulations, we only carry out interspecific communication and do not change their internal operation. Therefore, the parameters of ACS, MMAS, and ADCA can be determined separately to achieve the best effect respectively. Since ADCA comes from the algorithm in reference [23], in order to make a fair comparison with it, the parameter values of ADCA in this paper are the same as those in reference [23].

For the parameter values of ACS and MMAS, we adopted the Taguchi's method [27] to determine the value of each parameter, the levels are based on pre-experiments, the best scheme of each parameter is found out by orthogonal experiment. Each combination scheme for parameters was tested

TABLE 2. Experimental factors and levels of MMAS.

Level	α	β	ρ
Level 1	1	1	0.1
Level 2	2	2	0.2
Level 3	3	3	0.3

TABLE 3. Taguchi's test scheme and test results of MMAS.

$no.$	α	β	ρ	$results$
1	1	1	0.1	436.2
2	1	2	0.2	433.8
3	1	3	0.3	428.9
4	2	1	0.3	439.8
5	2	2	0.1	435.2
6	2	3	0.2	433.5
7	3	1	0.2	439.3
8	3	2	0.3	436.4
9	3	3	0.1	433.2

Note: The $no.$ is the number of the test, and the $results$ is the average value after 10 tests in each group.

TABLE 4. Analysis of test results of MMAS.

T	α	β	ρ
T_1	**1298.9**	1315.3	**1304.6**
T_2	1308.5	1305.4	1306.6
T_3	1308.9	**1295.6**	1305.1
t_1	**432.97**	438.43	**434.87**
t_2	436.17	435.13	435.53
t_3	436.30	**431.87**	435.03
max	436.30	438.43	435.53
min	432.97	431.87	434.87
$range$	3.33	6.56	0.66
$scheme$	**Level 1**	**Level 3**	**Level 1**

Note: T_i $(i = 1, 2, 3)$ are the sum of results. t_i $(i = 1, 2, 3)$ are the means of every level. $range$ is the difference by the max minus the min, which will be applied to determine which one factor is important, and larger $range$ is generally more important. And $scheme$ is the project of every factor by orthogonal test to obtain the best result.

TABLE 5. Experimental factors and levels of ACS.

factor	Level 1	Level 2	Level 3
α	1	2	3
β	2	3	4
ρ	0.1	0.2	0.3
ξ	0.2	0.3	0.4

10 times, and the average value was taken for analysis, taking eil51 as an example to determine the parameters. Through the above experimental analysis, the parameters in this paper are shown in Table.8.

TABLE 6. Taguchi's test scheme and test results of ACS.

$no.$	α	β	ρ	ξ	$results$
1	1	2	0.1	0.2	429.9
2	1	3	0.2	0.3	428.9
3	1	4	0.3	0.4	427.9
4	2	2	0.2	0.4	430.6
5	2	3	0.3	0.2	428.5
6	2	4	0.1	0.3	427.7
7	3	2	0.3	0.3	431.5
8	3	3	0.1	0.4	429.5
9	3	4	0.2	0.2	427.9

TABLE 7. Analysis of test results of ACS.

T	α	β	ρ	ξ
T_1	**1286.7**	1292.0	**1287.1**	**1286.3**
T_2	1286.8	1286.9	1287.4	1288.1
T_3	1288.9	**1283.5**	1287.9	1288.0
t_1	**428.90**	430.67	**429.03**	**428.77**
t_2	428.93	428.97	429.13	429.37
t_3	429.63	**427.83**	429.30	429.33
max	429.63	430.67	429.30	429.37
min	428.90	427.83	429.03	428.77
$range$	0.73	2.84	0.27	0.60
$scheme$	**Level 1**	**Level 3**	**Level 1**	**Level 1**

TABLE 8. Parameter setting.

ACO	α	β	ρ	ξ	q_0	Q	μ	C_2
ACS	1	4	0.1	0.2	0.8	/	/	/
MMAS	1	3	0.1	/	0.8	/	/	/
ADCA	1	4	0.1	0.3	0.8	1	0.7	0.1

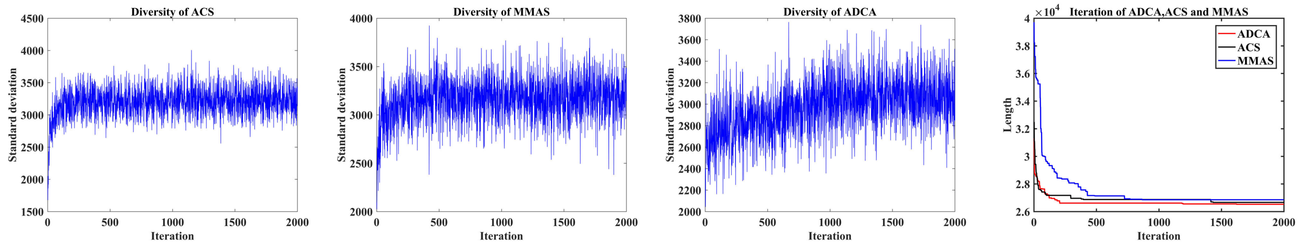
B. EXPERIMENT ANALYSIS OF MRCACO

1) PERFORMANCE ANALYSIS OF SELECTED COLONY

MRCACO is a hybridization of three different variants of ACOs including ACS, MMAS, and ADCA, wherein we claim that ACS guaranteed population convergence, MMAS guaranteed population diversity, ADCA balanced population diversity and convergence. We provide the following theoretical basis and experimental evidence to support this claim.

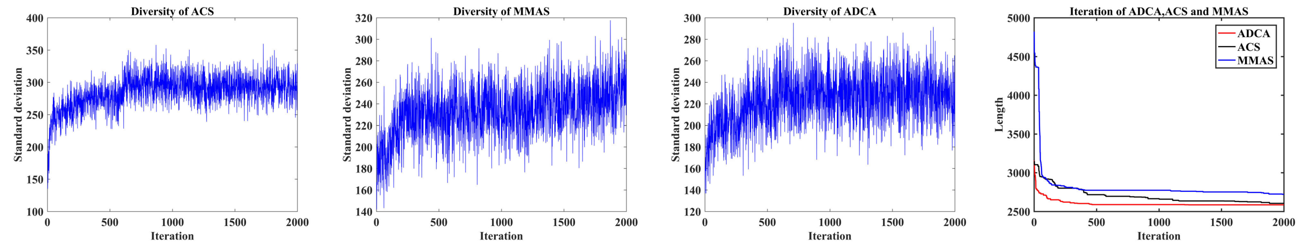
ACS and MMAS are recognized as classical ant colony optimization, as for the claims that ACS can accelerate convergence speed and MMAS can increase diversity, in the introduction, we give the theoretical basis by introducing the references [5], [6]. In subsection A of Section III, we give the theoretical basis for ADCA to balance diversity and convergence, ADCA algorithm comes from reference [23], in which it is introduced in detail how ADCA balances the diversity and convergence of the algorithm.

KroA150 and A280 were selected to participate in the comparative experiment of the diversity and convergence of the three algorithms. As the diversity in reference [23] is measured by standard deviation, for comparison, the standard deviation is also used here to measure diversity, the experimental results are shown in Fig.5 and Fig.6. By comparing the Fig.5(a), (b), (c) and Fig.6(a), (b), (c), we can see that the diversity of MMAS and ADCA is relatively good, and it can



(a) Diversity of ACS (b) Diversity of MMAS (c) Diversity of ADCA (d) The path comparison of ACO

FIGURE 5. Comparison of experimental data of kroA150.



(a) Diversity of ACS (b) Diversity of MMAS (c) Diversity of ADCA (d) The path comparison of ACO

FIGURE 6. Comparison of experimental data of a280.

be seen from Fig.5(d) and Fig.6(d) that ADCA and ACS have better convergence, thus providing experimental evidence for our claim.

2) STRATEGY TESTING AND PERFORMANCE ANALYSIS

The multi-role adaptive cooperative mechanism proposed in this paper includes three sub-populations: ACS, MMAS, ADCA. Two communication strategies: elite attribute learning strategy and pheromone balancing strategy. To verify the performance and effect of the two communication strategies and ADCA, the TSP instances set kroB100 and kroA150 were selected for testing, the selected instances set was tested for 15 times, with 2000 iterations per experiment. This experiment was analyzed from the following aspects: the optimal solution (Best), the error rate of the optimal solution (Er), the worst solution (Worst), the average solution (Mean), iteration number of optimal solution (Convergence), statistical experimental data were obtained in Table.9. The error rate of the optimal solution is expressed by (18).

First, ACS and MMAS, were selected for the test, without any communication mechanism, and only the optimal solution between the two sub-populations was retained, called AM algorithm, AM is the most basic algorithm. Then, elite attribute learning strategy is added to AM algorithm for testing, called AME algorithm, AME is used to test the performance of elite attribute learning strategy. Further, the ADCA subpopulation was added to the AME for testing, at this time, only the elite attribute learning strategy exists in the algorithm, and ADCA does not participate in communication, the optimal solution between the three sub-populations is retained, called AMEC algorithm, AMEC

TABLE 9. Performance analysis of algorithms composed of different strategies.

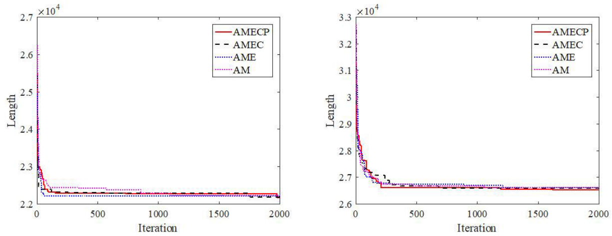
TSP	Opt	ACO	Best	Er	Worst	Mean	Convergence
kroB100	22141	AM	22237	0.43	22389	22312.13	1099
		AME	22220	0.36	22365	22283.2	**55**
		AMEC	22193	0.23	22342	22276.13	1754
		AMECP	**22179**	**0.17**	**22337**	**22259.47**	1975
kroA150	26524	AM	26619	0.36	27154	26864.87	1059
		AME	26616	0.35	27170	26873.27	1208
		AMEC	26604	0.30	27098	26840.6	**681**
		AMECP	**26528**	**0.01**	**26986**	**26748.73**	1621

tests the role of ADCA sub-populations. Finally, the AMEC algorithm is tested with a pheromone balancing strategy, called AMECP/MRCACO algorithm, which measures the performance of the pheromone balancing strategy.

$$Er = \frac{L_{ACO} - L_{min}}{L_{min}} \times 100\% \qquad (18)$$

where L_{ACO} is the optimal solution found for the algorithm, L_{min} is the standard optimal solution for the TSP instances.

Firstly, compare AM and AME algorithms to explore the effect of the elite attribute learning strategy. As can be seen from Table.9 and Fig.7, in both kroB100 and kroA150, the AME algorithm is more accurate at optimal, worst, average solutions. The Convergence of AME algorithm on kroB100 is 55, compared with AM algorithm, the solution is more accurate and the convergence rate is much faster, but at this point, the AME algorithm falls into the local optimal. This indicates that the elite attribute learning strategy can



(a) kroB100 (b) kroA150

FIGURE 7. Comparison of convergence rates of different algorithms.

Hypothesis Test Summary

	Null Hypothesis	Test	Sig.	Decision
1	The distributions of MRCACO, JCACO, ACS, MMAS, PCCACO, EDHACO and DACS are the same.	Related-Samples Friedman's Two-Way Analysis of Variance by Ranks	.000	Reject the null hypothesis.

Asymptotic significances are displayed. The significance level is .05.

FIGURE 8. Hypothesis Test Summary.

improve the accuracy of the solution, but may fall into the local optimal.

Then, compare AME and AMEC algorithms to explore the significance of adding ADCA algorithms. As can be seen from Table.9, after the addition of ADCA, the optimal solution accuracy of the algorithm is also improved. In the case of kroB100, AMEC is better than the AME algorithm in every respect, in kroA150, the optimal solution of AMEC is better than AME, which indicates that the ADCA algorithm plays a role of performance compensation for AME.

Finally, compare AMEC and AMECP algorithms to study the effect of pheromone balancing strategy. As can be seen from Table.9 and Fig.7, in both cases of kroB100 and kroA150, the accuracy of AMECP on the best solution, worst solution, and the average solution are much improved than AMEC, especially kroA150, and the algorithm does not fall into the local optimum. This shows that the pheromone balancing strategy can make the algorithm jump out of the local optimum and finally improve the accuracy of the solution.

To sum up, each strategy proposed in this paper can play a corresponding role, and link with each other.

3) STATISTICAL TEST OF THE ALGORITHMS

Since ant colony algorithm is a probabilistic algorithm, when we analyze the performance difference between the algorithms through experimental results, we cannot judge whether the difference is due to chance variation or our improvement work, we need to conduct significance test to detect whether the improved algorithm in this paper differs from traditional algorithm and other improved ant colony algorithms and whether the difference is significant. Since the Friedman test does not require the assumption of normality and homogeneity of variance, this paper uses the Friedman test for the significance test. We selected the experimental data from eil51, kroB150, and kroA200 respectively to conduct the Friedman test in SPSS25 software, JCACO [28], PCCACO [29], EDHACO [30], and DACS [31] are published ant colony optimization.

First, we give the null hypothesis H_0: there is no significant difference in the performance of the seven algorithms. Then, input the data into SPSS25 software, and the final result is given in Fig.8. The significance level in Fig.8 is $p = 0 < 0.5$, so decision-makers reject the null hypothesis, which means that the performance of the seven algorithms is significantly different. It can be seen from Fig.9 that the average ranks of MRCACO, JCACO, ACS, MMAS, PCCACO, EDHACO, and DACS are 1.71, 3.14, 3.88, 6.1, 3.12, 5.26, 4.79, respectively. As the response rates of different frequencies are different, the pairwise comparison is needed, the pairwise comparison results are shown in Fig.10. From Fig.10, we can see that the adjustment significance of MRCACO and the other six algorithms is less than 0.05. In summary, MRCACO is different from JCACO, ACS, MMAS, PCCACO, EDHACO, and DACS. In other words, the performance comparison between MRCACO and other algorithms has statistical significance in the subsequent experiments.

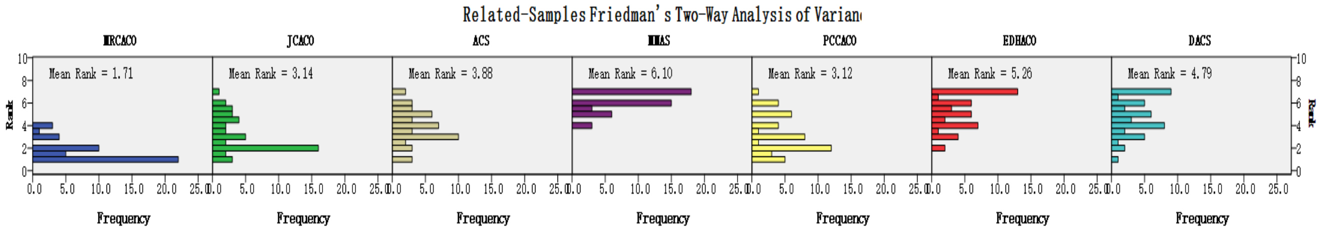
4) COMPARATIVE ANALYSIS OF MRCACO AND TRADITIONAL ANT COLONY ALGORITHM

To compare the performance of MRCACO, ADCA, ACS, and MMAS algorithms and better verify the effect of MRCACO, this paper selects 8 TSP instances of different scales for experiments. This experiment was analyzed from the following aspects: the optimal solution (Best), the worst solution (Worst), the average solution (Mean), iteration number of optimal solution (Convergence), the error rate of the optimal solution (Er), and Standard deviation (dev). Experimental data are shown in Table.10. In this paper, the standard deviation is used to measure the stability of the algorithm, and the (19).

$$dev = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(l_i - l_{avg}\right)^2} \tag{19}$$

where N is the number of times each TSP instance is tested (in this paper $N = 15$), l_i is the current optimal solution for each experiment, l_{avg} is the average solution of N experiments.

As can be seen from Table.10, the MRCACO algorithm was tested in the 8 TSP instances selected, and the accuracy of the optimal solution, the worst solution, and the average solution were better than that of ADCA, ACS, and MMAS, and the stability of MRCACO algorithm is better. In the small-scale TSP instance set within 150, MRCACO can find the optimal solution, and the convergence speed of the



Total N	45
Test Statistic	133.487
Degrees of Freedom	6
Asymptotic Sig. (2-sided test)	.000

FIGURE 9. **Related-Samples Friendman's Two-Way Analysis of Variance.**

Each node shows the sample average rank.

Sample1-Sample2	Test Statistic	Std. Error	Std. Test Statistic	Sig.	Adj.Sig.
MRCACO-PCCACO	-1.411	.455	-3.098	.002	.041
MRCACO-JCACO	-1.433	.455	-3.147	.002	.035
MRCACO-ACS	-2.167	.455	-4.758	.000	.000
MRCACO-DACS	-3.078	.455	-6.758	.000	.000
MRCACO-EDHACO	-3.544	.455	-7.783	.000	.000
MRCACO-MMAS	-4.389	.455	-9.637	.000	.000

FIGURE 10. **Pairwise comparison.**

algorithm is also good. In the medium-scale TSP instance between 150 and 300, although MRCACO could not find the standard optimal solution, the accuracy of the solution was improved compared with the other three algorithms, and the error rate of the optimal solution was kept within 1%. In the large-scale TSP instance above 300, although MRCACO could not find the standard optimal solution, the accuracy and stability of MRCACO algorithm still had certain advantages over the other three algorithms.

To sum up: The MRCACO algorithm proposed in this paper can improve the accuracy and stability of the solution, and the algorithm can jump out of the local optimum.

This paper uses the standard deviation (Equation(19)) to measure the stability of the algorithm, and Fig.11 shows the standard deviation of the TSP participating in the experiment. It can be seen from Fig.11 that the standard deviation of MRCACO is lower than the other three algorithms, which

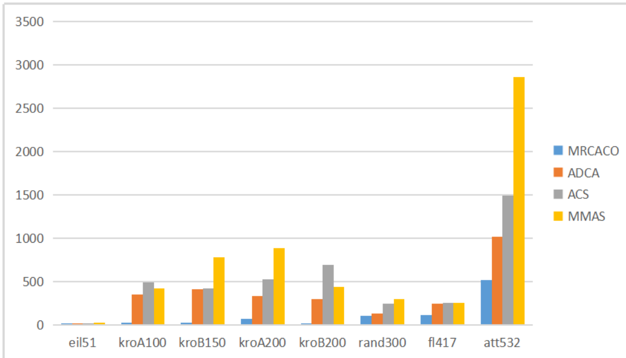


FIGURE 11. **Comparison of stability of different algorithms.**

indicate that the stability of MRCACO is better than that of ADCA, ACS, and MMAS.

Fig.12 shows the path diagram of the optimal solution obtained by using the MRCACO algorithm in the 8 TSP instances participating in the experiment.

5) COMPARATIVE ANALYSIS OF MRCACO AND OTHER OPTIMIZATION ALGORITHMS

MRCACO is also compared with other optimization algorithms to verify its performance, as shown in Table.11. The fairness of the comparison between intelligent algorithms is usually considered. The fairness of the algorithm selected in this paper can be explained from the determination of parameters. The parameter value of the intelligent algorithm is closely related to the actual problem and the scale of the problem. The authors usually use experimental tests (including orthogonal test, Taguchi's Design, Contour Plot, and so on) to find the best set of parameters to achieve the optimal effect. Moreover, the optional range of parameters among similar algorithms is the same. Therefore, the comparison between the optimization algorithms in this paper is fair.

TABLE 10. Performance comparison of MRCACO, ADCA, ACS, MMAS in different TSP instances.

TSP	Opt	ACO	Best	Worst	Mean	*Er*	*dev*	Convergence
eil51	426	MRCACO	**426**	**427**	**426.73**	**0**	**0.733**	198
		ADCA	**426**	428	427	**0**	0.867	**159**
		ACS	**426**	435	428	**0**	6.867	1092
		MMAS	427	441	432	0.23	8.400	352
kroA100	21282	MRCACO	**21282**	**21415**	**21304.40**	**0**	**22.400**	**297**
		ADCA	**21282**	21704	21356	**0**	348.000	750
		ACS	**21282**	21926	21433	**0**	492.333	609
		MMAS	21346	22075	21652	0.30	422.533	1272
kroB150	26130	MRCACO	**26130**	**26520**	**26319.67**	**0**	**23.667**	**1059**
		ADCA	**26130**	26881	26468	**0**	412.867	1768
		ACS	26358	26941	26498	0.87	418.500	1367
		MMAS	26704	27887	27107	2.20	780.067	1559
kroA200	29368	MRCACO	**29394**	**29654**	**29511**	**0.09**	**65.000**	1561
		ADCA	29401	29854	29525	0.11	328.533	843
		ACS	29486	29926	29604	0.40	521.267	**299**
		MMAS	30232	31532	30650	2.94	881.400	1541
kroB200	29437	MRCACO	**29447**	**29993**	**29766.07**	**0.03**	**15.933**	1305
		ADCA	29695	30224	29924	0.87	299.733	1715
		ACS	29819	30888	30194	1.29	693.200	**1114**
		MMAS	30423	31514	31073	3.35	440.867	1708
rand300	11865	MRCACO	**11969**	**12311**	**12135.4**	**0.88**	**102.600**	1493
		ADCA	12060	12345	12216	1.64	128.867	**1400**
		ACS	12022	12387	12139	1.32	247.667	1721
		MMAS	12346	12932	12637	4.05	294.667	1985
fl417	11861	MRCACO	**11969**	**12208**	**12093.2**	**0.91**	**112.000**	1930
		ADCA	12163	12561	12319	2.55	242.000	1923
		ACS	12193	12584	12330	2.80	254.267	1989
		MMAS	12664	13987	13116	6.77	257.733	**1890**
att532	86729	MRCACO	**88576**	**90504**	**90006.73**	**2.13**	**517.733**	1940
		ADCA	91680	94463	93450	5.71	1013.300	**1687**
		ACS	89652	92531	91044	3.37	1487.300	1986
		MMAS	93211	98259	95398	7.47	2861.000	1878

It can be seen from Table.11, compared with other bionic algorithms, the MRCACO algorithm can find better solutions in most TSP instances.

According to the comparison and analysis of the above experimental data, it can be seen that the MRCACO algorithm has certain advantages over traditional ant colony optimization, improved ant colony optimization, other intelligent algorithms, and the solution quality and convergence speed are improved to some extent.

V. APPLICATION RESEARCH OF MRCACO ON PATH PLANNING

The MRCACO algorithm proposed in section III is suitable for solving TSP. When the MRCACO algorithm is used to solve the robot path planning problem, the following problems need to be given attention to: 1. Whether heuristic functions are applicable. 2. When using ant colony optimization to solve the path planning problem, the actual map environment is usually converted to a raster map for simulation. In the raster map, path planning inevitably leads to path deadlock, so the deadlock problem has to be solved. 3. Pheromones are stored in the grid. 4. The result of path planning is a line segment from the start point to the terminal point and the grid in the environment does not have to go through it all.

A. IMPROVED HEURISTIC FUNCTION

In solving the TSP problems, the value of the heuristic function η_{ij} is (3). However, when solving the path planning problem, the distance between the current grid i and the next grid j is 1 or 1.414, and the path length is not significantly different, so the effect is not obvious. Therefore, (20) is adopted in this paper to define the heuristic function, Where d_{ij} is the distance from the current grid i to the next grid j, d_{jG} is the

TABLE 11. Comparison of MRCACO and other algorithms in TSP instances.

TSP	Index	MRCACO	JCACO 2019 [28]	PCCACO 2019 [29]	EDHACO 2019 [30]	DACS 2018 [31]	PACO-3opt 2018 [32]	DSMO 2020 [33]	GAACO 2020 [34]
eil51	Best	**426**	**426**	**426**	**426**	**426**	**426**	428.86	**426**
	Mean	426.73	427	427	431	433	**426.35**	436.96	426.7
kroA100	Best	**21282**	**21282**	**21282**	**21282**	**21282**	**21282**	21298.21	**21282**
	Mean	**20304.4**	21322	21383	21355.13	21634	21326.8	22024.27	21338.6
kroB100	Best	22179	**22141**	-	22237	22235	-	22308	**22141**
	Mean	22259.47	**22248**	-	22445.6	22607	-	23022.37	22249.85
kroA150	Best	**26528**	26621	26654	26727	26792	-	27591.44	**26528**
	Mean	26748.73	26845	**26715**	27003.9	26957	-	28354.09	26798.45
kroB150	Best	**26130**	**26130**	**26130**	26328	26147	-	26601.94	**26130**
	Mean	26319.67	**26233**	26241	26873.4	26739	-	27576.16	26365.7
kroA200	Best	29394	29406	29391	29694	29539	29533	30481.35	**29368**
	Mean	29511	29542	29485	30391	30218	29644.5	31828.64	**29465.4**
kroB200	Best	**29447**	29525	29541	-	-	-	30716.5	29460
	Mean	**29766.07**	29982	29878	-	-	-	31781.62	29787.6
rand300	Best	**11969**	11976	-	-	-	-	-	-
	Mean	12135.4	**12082**	-	-	-	-	-	-
fl417	Best	**11969**	**11969**	-	-	-	11972	12218.98	-
	Mean	12092.2	12216	-	-	-	**11991.9**	12950.77	-
att532	Best	**88576**	88895	-	-	-	-	-	-
	Mean	**90006.73**	90842	-	-	-	-	-	-

distance from the next grid j to the terminal grid G.

$$\eta = \frac{1}{d_{ij} + d_{jG}} \tag{20}$$

B. DEADLOCK ROLLBACK STRATEGY

Deadlock refers to the phenomenon that the ant is forced to terminate the path search after it transfers to a non-terminal grid and cannot find the next grid that meets the transfer conditions. The deadlocked ant does not reach the terminal point, and its search path is invalid, which is equivalent to reducing the number of effective ants in the algorithm, which is not conducive to the convergence speed and accuracy of the algorithm.

Path deadlock is generally divided into two cases (Fig.13): 1. Obstacle deadlock caused by getting into the obstacle concave zone. 2. Self-deadlock due to self-path planning. This paper proposes two different deadlock rollback strategies to make them jump out of the current deadlock quickly, improve the algorithm efficiency, and to some extent, make use of part of the path that the ant has searched.

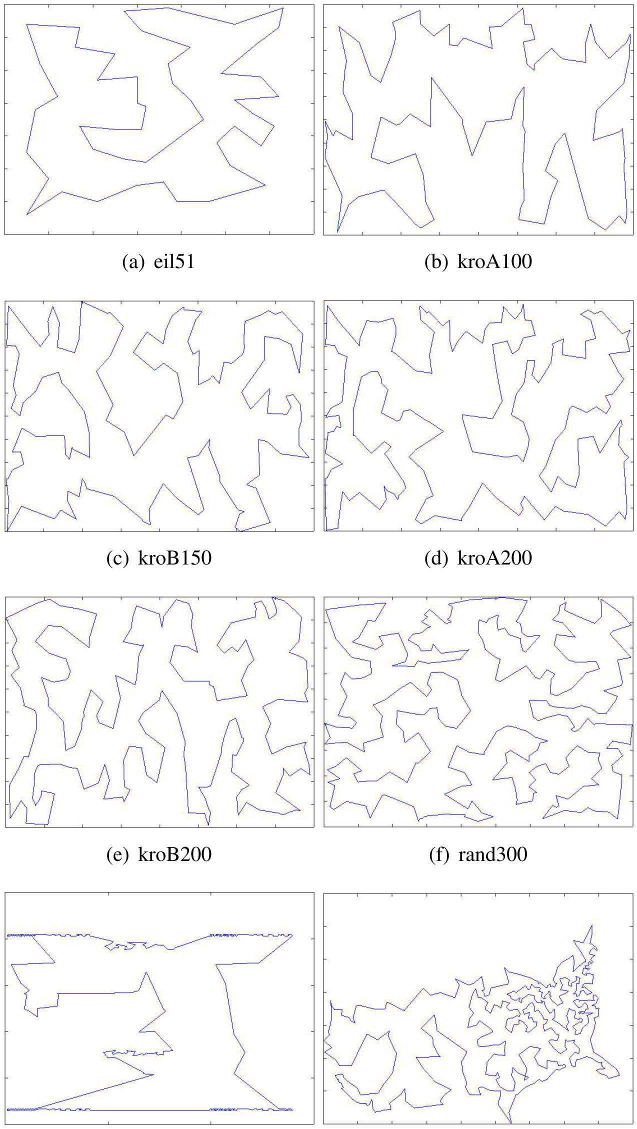
1) ROLLBACK STRATEGY FOR OBSTACLE DEADLOCK

Any ant that moves to a concave obstacle zone is bound to have an obstacle deadlock. The rollback strategy for the obstacle deadlock is as follows: Step1, set up a global taboo table, which has an effect on all ants in all iterations. Step2, when the ant is trapped in an obstacle deadlock, find the grid node closest to the occurrence of deadlock, and there is at least one feasible node in the adjacency matrix of this grid node. Step3, add all nodes between the deadlock node and the rollback node (including the deadlock node but not the rollback node) into the global taboo table to prevent other ants from entering these nodes again.

For example, in Fig.13(a), grid 1 is the starting point, grid 49 is the terminal point, the ant passes through the path 1-2-3-4-11-18-25-32-39 which occurs an obstacle deadlock, according to the rollback strategy, grid 11 is the rollback node, grid 18,25,32,39 are placed into the global taboo table. After the rollback strategy is executed, the next node selected is grid 5, as shown in Fig.14(a).

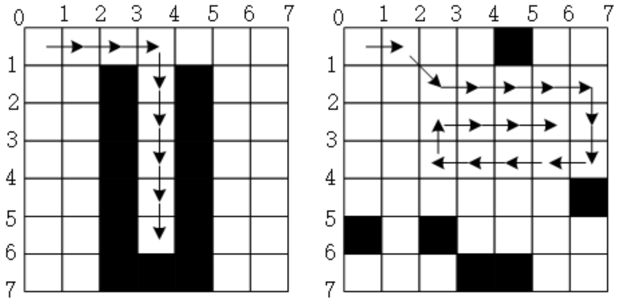
2) ROLLBACK STRATEGY FOR SELF-DEADLOCK

The main reason for the deadlock in Fig.13(b) is that it is surrounded by its path. The rollback strategy for self-deadlock is as follows: Step1, bind a local taboo table to each ant in each iteration, the local taboo table only affects the bound ants to prevent the ants from entering these grids again in the next path planning. Step2, when the ant is trapped in a self-deadlock, find the grid node closest to the occurrence of deadlock, and there is at least one feasible node in the adjacency matrix of this grid node. Step3, add all nodes between the deadlock node and the rollback node (including the deadlock node but not the rollback node) to the local taboo table of the ant, and these nodes are punished by pheromone to prevent the offspring ants from following the path and falling into a self-deadlock. The pheromone penalty rule is



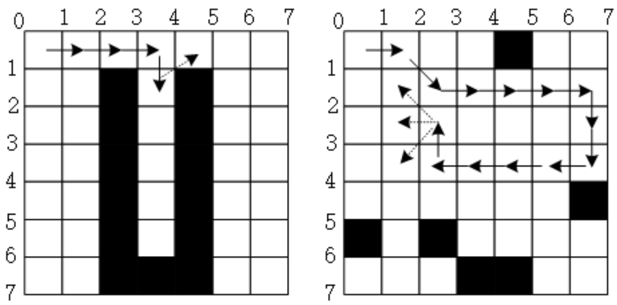
(a) eil51 (b) kroA100 (c) kroB150 (d) kroA200 (e) kroB200 (f) rand300 (g) fl417 (h) att532

FIGURE 12. Optimal solution path for each TSP instance found by MRCACO.



(a) Obstacle deadlock (b) Self-deadlock

FIGURE 13. Figure of deadlock types.



(a) Rollback strategy for obstacle deadlock. (b) Rollback strategy for self-deadlock.

FIGURE 14. Deadlock rollback strategies.

that in a local pheromone update, the node only volatiles, and no pheromone is added.

For example, in Fig.13(b), grid 1 is the starting point, grid 49 is the terminal point, the ant passes through the path 1-2-10-11-12-13-14-21-28-27-26-25-24-17-18-19-20 which occurs self-deadlock, according to the rollback strategy, grid 17 is the rollback node, and grid 18,19,20 are placed into the local taboo table and punish the pheromones on grid 18,19,20. After the rollback strategy is executed, the next node selected is generated in grid 9, 16, 23, as shown in Fig.14(b).

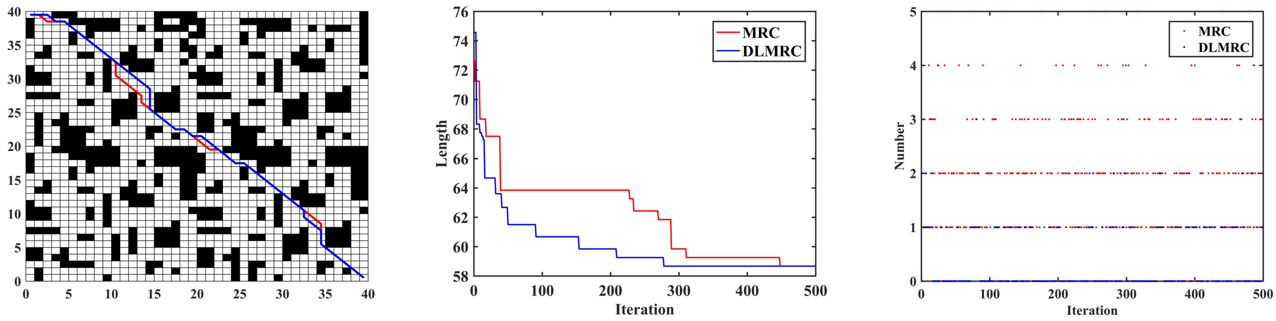
C. CASE APPLICATION AND RESULT ANALYSIS

1) THE APPLICATION OF MRCACO IN SIMULATION MAP

The MRCACO algorithm is applied to the path planning of the robot, to simplify the procedure, this paper considers the robot as a particle during the experimental simulation. When the robot conducts oblique line transfer to the upper left, lower left, upper right, and lower right grids, it is not necessary to require free grids on both sides of the oblique line.

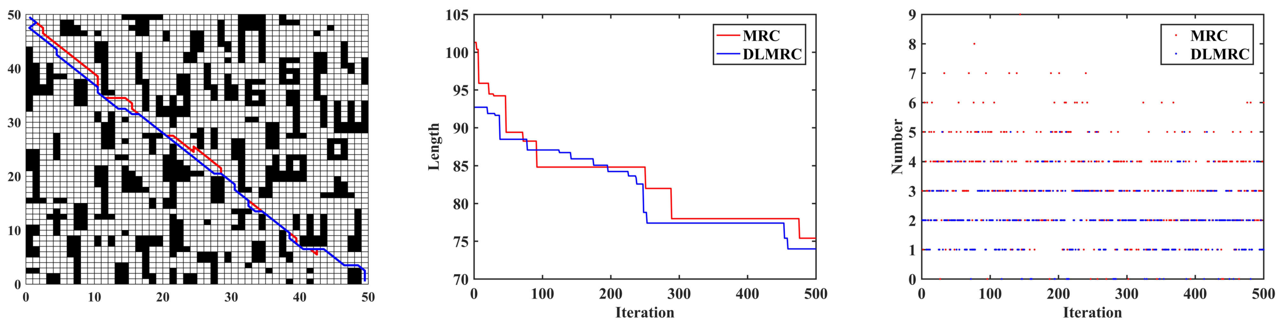
To verify the effectiveness of the MRCACO algorithm in path planning and the effectiveness of the deadlock rollback strategy, MRCACO without a deadlock rollback strategy is regarded as an MRC algorithm, and MRCACO with deadlock rollback strategy is regarded as DLMRC algorithm. MRC and DLMRC were used to carry out repeated path planning for 15 times in map 1(scale 40*40), map 2(scale 50*50), and map 3(scale 60*60), the optimal path, convergence rate and the number of invalid ants of the two algorithms are compared. The simulation results are shown in Fig.15,16, and 17 respectively.

Fig.15(a), 16(a), and 17(a) are the route maps of the optimal paths found by the two algorithms, where blue represents DLMRC and red represents MRC. Fig.15(b), 16(b), and 17(b) are the path iteration diagram of the two algorithms. Fig.15(c), 16(c), and 17(c) are scatter plots of number of invalid ants compared between the two algorithms. As can be seen from these figures, DLMRC can find a better path than MRC, and the convergence speed of the algorithm is also greatly improved. Fig,15(c) shows that when DLMRC algorithm is used, the number of invalid ants is basically stable at 0. In Fig.16(c), the number of invalid ants is mostly at 1, 2,



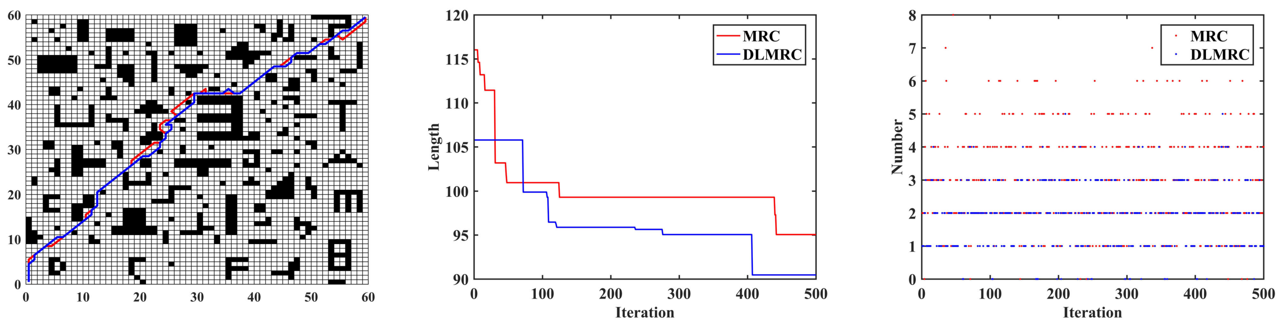
(a) The optimal path. (b) Path iteration diagram. (c) Number of invalid ants.

FIGURE 15. Performance comparison between MRC and DLMRC algorithms.



(a) The optimal path. (b) Path iteration diagram. (c) Number of invalid ants.

FIGURE 16. Performance comparison between MRC and DLMRC algorithms.



(a) The optimal path. (b) Path iteration diagram. (c) Number of invalid ants.

FIGURE 17. Performance comparison between MRC and DLMRC algorithms.

and 3 when the DLMRC algorithm is used, and the number of invalid ants is mostly at 4, 5 when the MRC algorithm is used. This is also the case with Fig.17(c). This shows that the use of the deadlock rollback strategy proposed in this paper is effective.

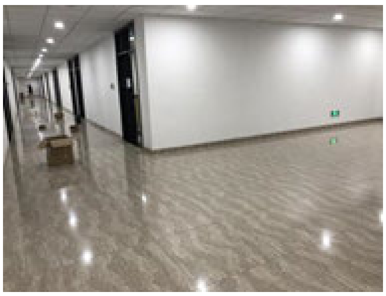
The simulation results of the two algorithms show that the DLMRC algorithm can significantly reduce the number of invalid ants in these three maps, find better paths and converge faster. DLMRC algorithm has higher accuracy, efficiency, and a faster convergence rate than the MRC algorithm.

The path planning of the two algorithms was repeated for 15 times in the three kinds of maps, and the feasibility and effectiveness of the proposed algorithm were discussed by comparing the optimal path length, the worst path length, the average path length, and the stability of the algorithm (standard deviation, calculated as (19)).

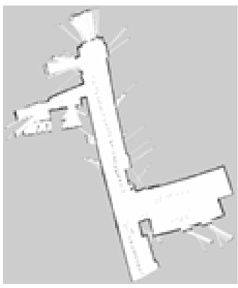
It can be seen from Table.12 that both algorithms can find better solutions in maps. In map 1, both algorithms are easy to find the optimal solution, but the two algorithms differ

TABLE 12. Comparison of the performances of the two algorithms in three maps.

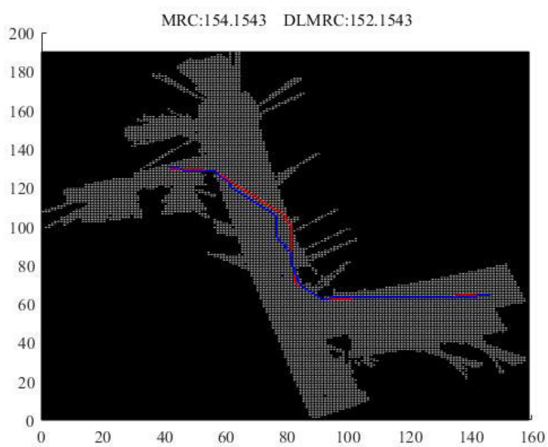
Performance indexes	map1		
	MRC	DLMRC	Performance improvement (%)
Optimal path length	**58.669**	**58.669**	0
Worst path length	63.0122	**59.4975**	5.58
Average path length	60.6462	**58.9586**	2.78
The standard deviation	1.1487	**0.2962**	74.21
Performance indexes	map2		
	MRC	DLMRC	Performance improvement (%)
Optimal path length	75.397	**73.9828**	1.88
Worst path length	85.6396	**78.8112**	7.98
Average path length	82.9082	**77.5599**	6.45
The standard deviation	1.5598	**0.4229**	72.89
Performance indexes	map3		
	MRC	DLMRC	Performance improvement (%)
Optimal path length	95.0538	**90.468**	4.82
Worst path length	104.0244	**99.196**	4.64
Average path length	100.6638	**97.2373**	3.40
The standard deviation	1.4611	**0.2840**	80.56



(a) Real map.

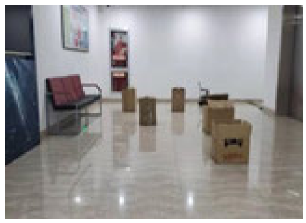


(b) Scanned map.

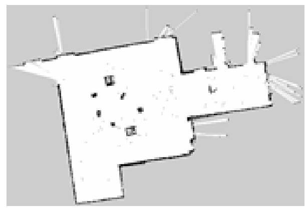


(c) Grid path diagram.

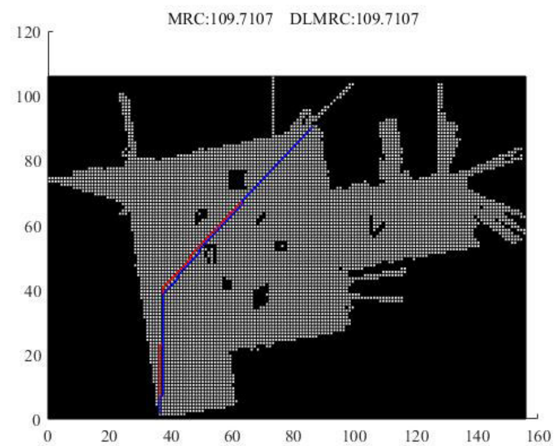
FIGURE 18. Scanning map 1.



(a) Real map.



(b) Scanned map.



(c) Grid path diagram.

FIGURE 19. Scanning map 2.

greatly in the worst path length, average path length, and stability. In map 2 and map 3, the four performance indexes of the DLMRC algorithm are better than MRC algorithm. With the expansion of the map scale, the ability of the DLMRC algorithm to search the optimal path is getting better and better, the DLMRC algorithm has higher stability. Thus, with the increase of the map scale, the performance advantages of the DLMRC algorithm gradually emerge.

2) PRACTICAL APPLICATION OF MRCACO IN PATH PLANNING

According to the simulation experiment, the MRC algorithm and the DLMRC algorithm are feasible in the path planning problem, next we will use the robot to scan two real maps. Fig.18(a) and 19(a) are the actual built environment diagrams. Fig.18(b) and 19(b) are a PGM version of the reality map scanned by the turtlebot2 robot, white is the feasible area, black is the obstacle area, gray is the unfeasible unknown area, some slender white protrusions are errors caused by radar scanning. Fig.18(c) and 19(c) shows the converted grid diagram, using two algorithms to find the optimal path on the grid diagram, blue is the path sought by DLMRC, and red is the path sought by MRC.

In Fig.18(c), the optimal path found by MRC is 154.1543, and the optimal path found by DLMRC is 152.1543. In Fig.19(c), the optimal paths found by MRC and DLMRC are both 109.7107, although the paths have the same length, they are different. This shows that both MRC and DLMRC can better solve the problem of path planning, and DLMRC is more effective than MRC.

Based on the above experiments, it can be seen that the MRCACO algorithm and the improved algorithm can be used in the path planning problem, and have certain effects, with feasibility and effectiveness.

VI. CONCLUSION

This paper presents dynamic multi-role adaptive collaborative ant colony optimization (MRCACO). Firstly, an adaptive dynamic complementary algorithm (ADCA) is proposed, there is a dynamic grouping mechanism in ADCA, so that it can balance diversity and convergence, and the three subpopulations ACS, MMAS, and ADCA form heterogeneous multi-colony. Then, a multi-role adaptive collaborative mechanism is proposed, there are two communication strategies under this mechanism: elite attribute learning strategy and pheromone balancing strategy. Elite attribute learning strategy is used to highlight the role of elite attributes and improve the comprehensive performance of the population. Pheromone balancing strategy is used to make the algorithm jump out of the locally optimal. Finally, by selecting several TSP instances for simulation, it is found that the algorithm proposed in this paper can provide higher accuracy of the solution.

The proposed algorithm MRCACO is applied to the path planning problem, making it useful and meaningful. In the study of path planning, it is necessary to improve the existing algorithm to make it applicable, so the DLMRC algorithm is proposed. Firstly, the improvement of heuristic function in this paper is the same as that in most literature, highlighting the role of the shorter path. Besides, this paper gives solutions to two kinds of deadlock problems, and puts forward two slightly different deadlock rollback strategies to reduce the generation of ineffective ants. Through the simulation of maps with obstacles of different sizes, it is found that the MRC and DLMRC algorithms proposed in this paper are feasible to solve the path planning, and the DLMRC algorithm can improve the path optimization efficiency and accuracy.

The MRCACO algorithm proposed in this paper also has certain limitations: when dealing with large-scale TSP problems, the accuracy of the solution needs to be further improved. For example, TSP instances in the thousands or even tens of thousands are better handled by the current popular clustering algorithm.

Future research directions are as follows: 1. When the ant colony algorithm is applied to the path planning problem, the dynamic obstacles are added to the actual map to study. 2. Apply the improved ant colony algorithm to more practical problems than just path planning. For example, the routing problem of communication network, vehicle scheduling problem, and so on.

REFERENCES

[1] Y. Zhong, J. Lin, L. Wang, and H. Zhang, "Discrete comprehensive learning particle swarm optimization algorithm with metropolis acceptance criterion for traveling salesman problem," *Swarm Evol. Comput.*, vol. 42, pp. 77–88, Oct. 2018.

[2] G. Li and W. Chou, "Path planning for mobile robot using self-adaptive learning particle swarm optimization," *Sci. China Inf. Sci.*, vol. 61, no. 5, pp. 263–280, May 2018.

[3] M. Elhoseny, A. Tharwat, and A. E. Hassanien, "Bezier curve based path planning in a dynamic field using modified genetic algorithm," *J. Comput. Sci.*, vol. 25, pp. 339–350, Mar. 2018.

[4] 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.

[5] M. Dorigo and L. M. Gambardella, "Ant colony system: A cooperative learning approach to the traveling salesman problem," *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 53–66, Apr. 1997.

[6] M. Mahi, Ö. K. Baykan, and H. Kodaz, "A new hybrid method based on particle swarm optimization, ant colony optimization and 3-opt algorithms for traveling salesman problem," *Appl. Soft Comput.*, vol. 30, pp. 484–490, May 2015.

[7] M. Mahi, O. K. Baykan, and H. Kodaz, "MAX–MIN ant system," *Future Gener. Comput. Syst.*, vol. 16, no. 8, pp. 889–914, 2000.

[8] M. Mandloi and V. Bhatia, "A low-complexity hybrid algorithm based on particle swarm and ant colony optimization for large-MIMO detection," *Expert Syst. Appl.*, vol. 50, pp. 66–74, May 2016.

[9] Y. Ge and A. Wang, "Spatial scheduling strategy for irregular curved blocks based on the modified genetic ant colony algorithm (MGACA) in shipbuilding," *Int. J. Prod. Res.*, vol. 56, no. 9, pp. 3099–3115, May 2018.

[10] M. Xu, X. You, and S. Liu, "A novel heuristic communication heterogeneous dual population ant colony optimization algorithm," *IEEE Access*, vol. 5, pp. 18506–18515, 2017.

[11] X.-L. He, P. Zhang, M. Ma, J. Lin, and X. Huang, "Dual population ant colony algorithm based on heterogeneous ant colonies," *Comput. Eng. Appl.*, vol. 45, no. 27, pp. 36–38, 2009.

[12] Y.-C. Li and Y. Peng, "Improved artificial bee colony algorithm based on information entropy," *Control Decis.*, vol. 30, no. 6, pp. 1121–1125, Aug. 2015.

[13] K. Yang, X. You, and S. Liu, "Self-adaptive double-population ant colony algorithm with entropy," *Comput. Eng. Appl.*, vol. 55, no. 19, pp. 66–73, 2019.

[14] X. Sun, K. Zhang, M. Ma, and H. Su, "Multi-population ant colony algorithm for virtual machine deployment," *IEEE Access*, vol. 5, pp. 27014–27022, 2017.

[15] J. L. Bai, H. N. Chen, and Y. B. Hu, "Ant colony algorithm based on negative feedback and its application on robot path planning," *Comput. Integr. Manuf. Syst.*, vol. 25, no. 7, pp. 1767–1774, 2019.

[16] J. Liu, J. Yang, H. Liu, X. Tian, and M. Gao, "An improved ant colony algorithm for robot path planning," *Soft Comput.*, vol. 21, no. 19, pp. 5829–5839, Oct. 2017.

[17] W. Zhang, X. Gong, G. Han, and Y. Zhao, "An improved ant colony algorithm for path planning in one scenic area with many spots," *IEEE Access*, vol. 5, pp. 13260–13269, 2017.
[18] Y. N. Wang and H. W. Ge, "Improved simulated annealing genetic algorithm for solving TSP problem," *Comput. Eng. Appl.*, vol. 46, no. 5, pp. 44–85, 2010.
[19] R. Martí, A. Martínez-Gavara, J. Sánchez-Oro, and A. Duarte, "Tabu search for the dynamic bipartite drawing problem," *Comput. Oper. Res.*, vol. 91, pp. 1–12, Mar. 2018.
[20] L. Zhang, X. Qin, Q. Li, H. Peng, and Y. Hou, "Single bitmap block truncation coding of color images using hill climbing algorithm," 2018, *arXiv:1807.04960*. [Online]. Available: http://arxiv.org/abs/1807.04960
[21] N. Hansen, S. D. Müller, and P. Koumoutsakos, "Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES)," *Evol. Comput.*, vol. 11, no. 1, pp. 1–18, Mar. 2003.
[22] M. H. Lokman, I. Musirin, S. I. Suliman, H. Suyono, R. N. Hasanah, S. A. S. Mustafa, and M. Zellagui, "Multi-verse optimization based evolutionary programming technique for power scheduling in loss minimization scheme," *IAES Int. J. Artif. Intell.*, vol. 8, no. 3, pp. 292–298, 2019.
[23] D. Zhang, X. You, and S. Liu, "Dynamic grouping ant colony algorithm combined with cat swarm optimization," *J. Frontiers Comput. Sci. Technol.*, vol. 14, no. 5, pp. 880–891, 2020.
[24] Y. Xue, J. Jiang, B. Zhao, and T. Ma, "A self-adaptive artificial bee colony algorithm based on global best for global optimization," *Soft Comput.*, vol. 22, no. 9, pp. 2935–2952, May 2018.
[25] X. Li, L. Sun, J. Li, and H. Piao, "An improved fruit fly optimization algorithm and its application in heat exchange fouling ultrasonic detection," *Math. Problems Eng.*, vol. 2018, pp. 1–9, Jan. 2018.
[26] Q.-K. Pan, H.-Y. Sang, J.-H. Duan, and L. Gao, "An improved fruit fly optimization algorithm for continuous function optimization problems," *Knowl.-Based Syst.*, vol. 62, pp. 69–83, May 2014.
[27] S. Gao, M. Zhou, Y. Wang, J. Cheng, H. Yachi, and J. Wang, "Dendritic neuron model with effective learning algorithms for classification, approximation, and prediction," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 2, pp. 601–614, Feb. 2019.
[28] D. Zhang, X. You, S. Liu, and K. Yang, "Multi-colony ant colony optimization based on generalized Jaccard similarity recommendation strategy," *IEEE Access*, vol. 7, pp. 157303–157317, 2019.
[29] H. Zhu, X. You, and S. Liu, "Multiple ant colony optimization based on Pearson correlation coefficient," *IEEE Access*, vol. 7, pp. 61628–61638, 2019.
[30] J. Chen, X.-M. You, S. Liu, and J. Li, "Entropy-based dynamic heterogeneous ant colony optimization," *IEEE Access*, vol. 7, pp. 56317–56328, 2019.
[31] L. Zhongqiang, Y. Xiaoming, and L. Sheng, "Ant colony algorithm for heuristic dynamic pheromone update strategy," *Comput. Eng. Appl.*, vol. 54, no. 20, pp. 20–27, 2018.
[32] Ş. Gülcü, M. Mahi, Ö. K. Baykan, and H. Kodaz, "A parallel cooperative hybrid method based on ant colony optimization and 3-Opt algorithm for solving traveling salesman problem," *Soft Comput.*, vol. 22, no. 5, pp. 1669–1685, Mar. 2018.
[33] M. A. H. Akhand, S. I. Ayon, S. A. Shahriyar, N. Siddique, and H. Adeli, "Discrete spider monkey optimization for travelling salesman problem," *Appl. Soft Comput.*, vol. 86, Jan. 2020, Art. no. 105887.
[34] L. Meng, X. You, S. Liu, and S. Li, "Multi-colony ant algorithm using both generative adversarial nets and adaptive stagnation avoidance strategy," *IEEE Access*, vol. 8, pp. 53250–53260, 2020.

DEHUI ZHANG was born in Hefei, Anhui, China, in 1995. She received the bachelor's degree from Suzhou University, Suzhou City, Anhui. She is currently pursuing the M.S. degree with the Shanghai University of Engineering Science. Her research interests include intelligent algorithm, path planning of mobile robot, and embedded systems. She has won various scholarships and honors. She has published journal articles: *Dynamic grouping Ant Colony Algorithm combined with Cat Swarm Optimization*, published in the *Journal of Frontiers of Computer Science and Technology* and *Multi-Colony Ant Colony Optimization Based on Generalized Jaccard Similarity Recommendation Strategy*, published in IEEE ACCESS.

XIAOMING YOU was born in Huaihua, Jiangsu, China, in 1963. She received the Ph.D. degree in computer science from the East China University of Science and Technology, in 2007. She is currently a Professor and the M.S. Supervisor with the Shanghai University of Engineering Science. Her research interests include swarm intelligent systems, distributed parallel processing, and evolutionary computing.

SHENG LIU was born in Daye, Hubei, China, in 1966. He received the Ph.D. degree in computer science from the East China University of Science and Technology, in 2008. He is currently a Professor and the M.S. Supervisor with the Shanghai University of Engineering Science. His research interests include quantum inspired evolutionary computation, distributed parallel processing, and evolutionary computing.

HAN PAN was born in Anqing, Anhui, China, in 1994. He is currently pursuing the M.S. degree with the Shanghai University of Engineering Science. His research interests include intelligent algorithm, path planning of mobile robot, and embedded systems.