Received September 11, 2020, accepted September 24, 2020, date of publication October 2, 2020, date of current version October 13, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3028467

Dynamic Path Planning Based on Improved Ant Colony Algorithm in Traffic Congestion

CHUNJIANG WU, SHIJIE ZHOU, (Member, IEEE), AND LICAI XIAO

School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu 610054, China

Corresponding author: Shijie Zhou (sjzhou@uestc.edu.cn)

This work was supported in part by the Sichuan Science and Technology Program under Grant 2018GZDZX0033.

ABSTRACT Congestion road condition is an important factor that must be considered in urban traffic path planning, while most path planning algorithms only consider the distance factor, which is not suitable for the current complex urban traffic congestion road condition. In order to solve the above problems, this article proposes a dynamic path planning method based on improved ant colony algorithm in congested traffic. The method quantifies the main attributes of urban road length, number of lanes, incoming and outgoing traffic flow, and introduces the road factor used for replacing the distance parameters of particle swarm optimization and ant colony algorithm. In the method, the particle swarm algorithm can effectively optimize the parameters of the ant colony algorithm, and significantly improve the efficiency of ant colony algorithm, such that it is more applicable for dynamic path planning application to greatly reduce the congestion rate of path planning. In addition, this article selects some intersections in the Beijing area to carry out the dynamic path planning experiment based on the improved ant colony algorithm under congested road conditions. The experimental results show that, compared with the ant colony algorithm based on distance parameter, the proposed dynamic path planning method can effectively reduce the average congestion rate ranging from 9.73% to 13.63%.

INDEX TERMS Traffic congestion, road condition factor, ant colony algorithm, particle swarm optimization, path planning.

I. INTRODUCTION

With the rapid development of the economy and the rapid growth of urban car ownership, traffic congestion has become a serious problem faced by all large and medium-sized cities. Relieving traffic congestion and rationally planning travel routes have become hot research topics in the field of urban transportation [1]–[3].

In the application of urban traffic path planning, the existing algorithms can just find the shortest path between two points in the traffic road network while rarely considering the impact of the traffic environment. However, in the practical urban traffic, a path may be the shortest path but it is not necessarily the optimal one. This is because in the shortest path, there are often some traffic congestion issues to be considered, such as heavy traffic, high traffic density, and slow vehicle. These factors may seriously affect travel efficiency, further affect whether the selected path is the shortest path. At present, as a common problem for various large and medium-sized cities, congested road conditions in traffic congestion have become an important factor that must be considered in urban traffic path planning [4], [5].

The path planning algorithms for urban traffic mainly include Dijkstra [6], Floyd [7], A* [8], D* [9] and genetic algorithm [10]. With the development of urban traffic infrastructure, the growth of urban car ownership, the highly complicated factor of the situation of traffic roads and the rapidly increasing computational complexity of path planning, most of the above-mentioned path planning algorithms seem not to be appropriate for the urban traffic. This is due to the fact that the Dijkstra algorithm and Floyd algorithm have the issues of prohibitive computation efficiency and heavy space complexity, the A* algorithm is greatly suffered from the influence of the selected evaluation function, the D* algorithm also has a problem of high computation complexity, and the genetic algorithm has a hard problem of selecting genetic operators. As the capability of fast computing speed and good optimization, the ACO (Ant Colony Algorithm) [11]–[13] and PSO (Particle Swarm Optimization) [14]–[16]

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

are generally deemed to be more suitable for path planning than other algorithms, which has been popular and applied in the path planning. However, the existing ACO algorithms use the positive feedback and heuristically probabilistic search method. Such a method makes it difficult for ACO algorithm to fall into a local optimum. That is, it is easy to find a globally optimal solution, whereas the convergence of ACO algorithm is slower when the search space tends to be larger. Besides, although the existing PSO algorithms have some desirable characteristics, such as few parameters, collaborative search, and fast convergence, the local search ability of PSO algorithm is much limited, which leads not to ensuring that the global optimal solution can be found. Furthermore, most existing ACO and PSO algorithms are distance-based path planning algorithms, which are not suitable for the current complicated urban traffic congestion road conditions due to the neglect of practical road condition factor.

Aiming to considering the practical condition factor into dynamic path planning under congested road conditions, our main contributions are as follows:

- This article conducts a comprehensive evaluation of the main attributes of urban roads, models traffic conditions concerning the urban traffic congestion index, and proposes the concept of road conditions factors to replace the distance parameters in traditional path planning algorithms, which can more accurately describe the cost from one intersection to another intersection.
- This article proposes a dynamic path planning method based on improved ant colony algorithm in congested roads. This method combines the advantages of particle swarm optimization (PSO) and ant colony optimization (ACO), and replaces the distance parameter in the two algorithms with the road condition factor, which is more suitable for dynamic path planning in congested roads.
- In the practical experimental simulations, we choose some intersections in Beijing to conduct dynamic path planning experiments, which results show that our proposed method can improve the effectiveness and accuracy of path planning under congested road conditions since it considers the dynamic changes of traffic conditions and traffic congestion. Also, under congested road conditions, compared with the ACO algorithm based on distance parameter, the dynamic path planning method based on the improved ant colony algorithm proposed in this article can reduce the average congestion rate ranging from 9.73% to 13.63%.

Organization: In Section II, the related works including ant colony algorithm, particle swarm optimization and combining algorithms are introduced. Section III presents the basic knowledge of the used algorithms. Section IV presents the algorithm improvement containing the basic idea and road condition factor. In Section V, the simulation experiment is shown to our practicability of our scheme. Section VI summarizes the conclusion of this manuscript.

II. RELATED WORK

Urban traffic path planning is mainly to quickly find the shortest path between two points in the traffic network, which can be divided into static path planning and dynamic path planning.

Static path planning algorithms mainly include Dijkstra [6], Floyd [7], A* [8], etc. Dijkstra algorithm is a typical single-source shortest path algorithm, which requires that there is no negative edge weight in the graph. Floyd algorithm is to find the shortest path between any two points, which can correctly deal with the shortest path problem of directed graph or negative weight. The A* algorithm combines the merits of Dijkstra and BFS (Breadth First Search) in the algorithm, and can find a shortest path faster.

Dynamic path planning algorithms mainly include D* [9], genetic Algorithm [10], ACO(Ant Colony Algorithm) [11]–[13], PSO (Particle Swarm Optimization) [14]–[16], etc. D* algorithm is a heuristic path search algorithm, which can carry out dynamic search and is suitable for situations where the surrounding environment is unknown or dynamic changes exist in the surrounding environment. Genetic algorithm is a method to find the optimal solution by simulating the natural evolution process, which is suitable for solving complex optimization problems and has strong robustness. The ant colony algorithm was proposed by Italian scholars-Dorigo, Maniezzo and Colorni by simulating ant colony foraging behavior. It is a population-based simulation evolution algorithm. The basic idea of the algorithm is derived from the shortest path principle of ants foraging in nature. The particle swarm algorithm was proposed by Dr. Eberhart and Dr. Kennedy in 1995, and it is a swarm intelligence algorithm designed to simulate the predation behavior of birds. Its basic idea is to share information of individuals in the group, resulting that the movement of the entire group evolves from disorder to order.

With the development of urban traffic infrastructure, the growth of urban car ownership, the highly complicated factor of the situation of traffic roads and the rapidly increasing computational complexity of path planning, most of the above-mentioned path planning algorithms seem not to be appropriate for the urban traffic. This is due to the fact that the Dijkstra algorithm and Floyd algorithm have the issues of prohibitive computation efficiency and heavy space complexity, the A* algorithm is greatly suffered from the influence of the selected evaluation function, the D* algorithm also has a problem of high computation complexity, and the genetic algorithm has a hard problem of selecting genetic operators. As the capability of fast computing speed and good optimization, the ACO (Ant Colony Algorithm) and PSO (Particle Swarm Optimization) are generally deemed to be more suitable for path planning than other algorithms, which has been popular and applied in the path planning.

Although the ant colony algorithm can find the global optimal solution, it is prone to stagnation and has a high time complexity. Moreover, the ant colony algorithm is a

TABLE 1. Performance comparisons among similar schemes.

Scheme	Dynamic	Fast solution	Less space occupation	Easy optimization	Global optimal solution	Traffic path planning
Dijkstra [6]	×	×	✓	✓	✓	✓
Flord [7]	×	×	✓	✓	✓	✓
A* [8]	✓	×	×	×	✓	×
D* [9]	×	✓	×	×	✓	×
ACO [11]–[13], [17]–[20]	✓	×	×	×	✓	✓
PSO [14]–[16], [21], [22]	✓	✓	✓	✓	×	✓
Ours	✓	✓	✓	✓	✓	✓

typical probabilistic algorithm. Several important parameters in the algorithm usually need to be determined through repeated experiments, which makes the optimization of the algorithm difficult [17]–[20]. Although the particle swarm optimization algorithm is easy to fall into the local optimum, it has a low time complexity [21], [22]. Therefore, combining the ant colony algorithm with the particle swarm algorithm to form an ACO-PSO combined algorithm is a commonly chosen way in the field of urban traffic path planning research [23], [24].

In terms of the combined research of particle swarm optimization and ant colony algorithm, Min *et al.* used *PSO* to optimize *ACO*'s α and β parameters [25]. Cai *et al.* also utilized *PSO* to optimize *ACO*'s β, q_0 and ρ [26]. Sun *et al.* used *PSO* to optimize *ACO*'s α, β, ρ and γ [27] Based on PSO, Xie *et al.* proposed the optimal combination of *ACO* parameters [28]. Shi applied *ACO-PSO* in the path planning problem [29]. Gigras *et al.* proposed the hybrid *ACO-PSO* technology for path planning [30]. Pal *et al.* improved the hybrid *ACO-PSO* [31]. Ouyang *et al.* proposed an improved PSO-ACO algorithm to solve the large scale TSP problem [32]. Shuang *et al.* studied the hybrid *PSO-ACO* algorithm [33]. These results solved the problem of randomization of the parameters in the *ACO* algorithm. The *PSO* algorithm was used to determine and optimize the different parameters of the *ACO* algorithm, which improved the efficiency of the *ACO* algorithm.

Although the above articles have proposed a combination of particle swarm optimization and ant colony algorithm to solve the problem of path planning, they are all distance-based path planning algorithms, which are not suitable for the current complicated urban traffic congestion road conditions. Besides, in terms of traffic congestion discrimination, there are three algorithms: coil based traffic congestion discrimination, traffic video based traffic congestion discrimination and vehicle GPS data based traffic congestion discrimination. However, studies on urban traffic path planning and traffic congestion discrimination are independent at present, and few algorithms take traffic congestion into account in urban traffic path planning. The detailed performance comparison among similar schemes could be found in TABLE 1.

III. PRELIMINARIES

In the process of ant searching, there are two mathematical models introduced as follows: transfer probability model and pheromone update model.

A. TRANSFER PROBABILITY MODEL

The transfer probability model describes the probability of ants moving from point i to point j at time t, which is used to determine the direction of ants' next movement, as shown in formulas (1) and (2).

$$p_{ij}^k = \begin{cases} \dfrac{[\tau_{ij}(t)]^\alpha * \eta_{ij}(t)]^\beta}{\sum_{s\in J_k(i)}[\tau_{is}(t)]^\alpha * \eta_{is}(t)]^\beta}, & j \in J_k(i) \\ 0, & j \notin J_k(i) \end{cases} \tag{1}$$

$$\eta_{ij}(t) = \frac{1}{d_{ij}} \tag{2}$$

Here, $\tau_{ij}(t)$ represents the pheromone on the road (i, j) at time t. α is the pheromone factor $\eta_{ij}(t)$ represents expectation from point i to j. β is the heuristic function factor. J_k represents set of points the ant k heads for. d_{ij} represents the distance between points i and j.

The pheromone model describes how the pheromone of all paths is updated, as shown in formulas (3), (4), (5).

$$\tau_{ij}(t+n) = (1-\rho) * \tau_{ij}(t) + \Delta\tau_{ij} \tag{3}$$

$$\Delta\tau_{ij} = \sum_{k=1}^{m} \Delta\tau_{ij}^k \tag{4}$$

$$\Delta\tau_{ij}^k = \begin{cases} \dfrac{Q}{L_k}, & \textit{if the ant passed } (i,j) \\ 0, & \textit{otherwise} \end{cases} \tag{5}$$

Here, ρ represents the volatilization rate of pheromone from t to $t+n$. $\Delta\tau_{ij}$ represents the increment of pheromone concentration on the path (i, j) in this iteration. $\Delta\tau_{ij}^k$ represents the pheromone concentration left by ant k on path (i, j) in this iteration. Q is a constant. L_k represents the length of the path that ant k has traveled during this tour.

The steps of the ant colony algorithm are as follows:

1) Initialization: Set parameters, $t = 0, \tau_{ij}(t) = C$, $\Delta\tau_{ij} = 0$.
2) Ant placement: m ants are randomly placed at n points. For each ant, J_k is a set of all points except the starting point of ant k.
3) Path selection: Calculate the probabilities of ant k from i to j ($j \in J_k(i)$) according to the transition probability formula, select a point as the next destination and remove the point from J_k.
4) Pheromone update: When ant k walks the city and returns to the starting point, it obtains the current

optimal path and updates the pheromone matrix according to the pheromone update formula.

5) Exit: If the number of iterations is greater than threshold or the best path has not changed for a long time, the best path is found. Otherwise return to 3).

B. TRANSFER PROBABILITY MODEL

The particle swarm algorithm involves two models: velocity update model and position update model.

Each particle has two attributes, namely velocity $V_i = (v_{i1}, v_{i2}, \ldots, v_{id})$ and position $X_i = (x_{i1}, x_{i2}, \ldots, x_{id})$, and the best position experienced $pbest_i = (pbest_{i1}, pbest_{i2}, \ldots, pbest_{id})$.

The entire particle swarm records the best locations $gbest_i = (gbest_{i1}, gbest_{i2}, \ldots, gbest_{id})$ that the entire population has ever experienced. Then, each particle will update its speed and position according to the speed update formula (6) and position update formula (7).

$$v_i(t+1) = wv_i(t) + c_1r_1(pbest_i - x_i(t)) + c_2r_2(gbest_i - x_i(t)) \quad (6)$$

$$x_i(t+1) = x_i(t) + v_i(t) \quad (7)$$

Here, $v_i(t)$ represents the flight speed of particle i at the iteration t. $x_i(t)$ represents the position of particle i at the iteration t; w is a weight to adjusts the search range of the solution space. c_1 and c_2 are the acceleration constant to adjusts the maximum learning step. r_1 and r_2 are two random functions. $pbest_1$ represents the best position of the particle i and $gbest_1$ indicates the best position of the population.

The steps of the particle swarm algorithm are as follows:

1) Initialize a group of particles (group size is m), including random position, velocity, $pbest$ and $gbest$;
2) For each particle, update its velocity and position according to the velocity update formula and position update formula, and then calculate its adaptive value;
3) For each particle, compare its adaptive value with the best position $pbest$ it has experienced, and if it is better, set it as the current optimal position $pbest$;
4) For each particle, compare its adaptive value with the best position $gbest$ experienced globally, and if it is better, set it as $gbest$;
5) If the end condition is not satisfied (condition is usually threshold of adaptation value or threshold of maximum number of iterations), return to 2). Otherwise output $gbest$.

IV. ALGORITHM DESIGN

A. BASIC IDEA

In the existing urban traffic road network modeling, the physical distance between two intersections is usually used as a measure of the distance between the two intersections. But in the practical urban traffic environment, the road conditions are much complicated. Urban traffic conditions not only include the length of the road, but also include various attributes such as the number of lanes, entering traffic, exiting traffic, and traffic accidents. The greater the urban traffic flow is and the more complex the traffic conditions are, the greater the likelihood of traffic congestion is. If the path planning is based solely on the physical length of the road, the complex and changeable road conditions of urban traffic are simplified and the optimal solution cannot be found easily.

The basic idea of the dynamic path planning method proposed in this article under congested road conditions mainly includes the following three points:

1) Comprehensive evaluation of the main attributes of urban roads is a criterion for measuring the distance between two intersections.
2) Introduce the comprehensive evaluation value of the main attributes of urban roads into the PSO algorithm, which is used to quickly train three parameters α, β, and ρ of the ant colony algorithm, where α represents the importance of each point. β represents the importance of heuristic information. ρ is the volatilization rate of the pheromone within a certain period.
3) Introduce the comprehensive evaluation value of the main attributes of urban roads into the ACO algorithm, which is used to calculate transition probability and update pheromone.

This method fully considers that under urban traffic congestion road conditions, the length of the road is not the only criterion for measuring the distance of the path. At the same time, the proposed method takes advantage of the $ACO-PSO$ combined algorithm, resulting in a quick convergence and global optimal solution.

Remark: Three important parameters are involved in our designed scheme, such as α, β, ρ. These three parameters directly affect the efficiency of the designed algorithm. The fundamental reason leading to this is that these three parameters have a large adjustable range. Hence, it is necessary to reduce the adjustable range as much as possible. As an alternative tool, PSO can significantly reduce the range of this parameter, and can quickly optimize the solution speed compared to other algorithms such as the F-Race. The functions of these parameters in an ACO algorithm are described as follows: (1) For α, it reflects the intensity of pheromone factors. The larger the value, the greater the possibility that the ants will choose the path they have walked before. In other words, the randomness of the search will be weakened. When the value is much larger, the search of the ant colony will fall into the local optimum prematurely; On the contrary, when the value is much smaller, the randomness of the search seems to be enhanced and the algorithm convergence speed slows down. (2) For β, it reflects the intensity of a prior and deterministic factors. The larger the value, the greater the probability that the ants will choose the local shortest path at a certain local point. In other words, the randomness of the algorithm is weakened, thus it is easy to fall into the local optimum; while the smaller the value, it will cause the ant colony to fall into pure random search, thus resulting in that it is difficult to find optimal solution. (3) For ρ, it is directly related to the global search ability of the ant colony

TABLE 2. Traffic Congested Index.

Congestion condition	Unblocked	Basically smooth	Light congestion	Light congestion	Severe congestion
Index value	TCI $\in [0, 0.2]$	TCI $\in [0.2, 0.4]$	TCI $\in [0.4, 0.6]$	TCI $\in [0.6, 0.8]$	TCI $\in [0.8, 1]$

algorithm and its convergence speed. When it is relatively larger, because the positive feedback of information dominates, it is too likely that the previously searched path will be selected again, and the randomness of the search is weakened; on the contrary, when it is much smaller, the role of positive feedback of information is relatively weak. The randomness of search process will be enhanced, so the convergence speed of ant colony algorithm is very slow.

B. ROAD CONDITION FACTOR

In this article, we first comprehensively evaluate the main attributes of urban roads as a measure of the distance between two intersections. We build an urban traffic congestion model. The model integrates the main attributes, such as urban road length, the number of lanes, inbound traffic, outbound traffic, etc, to jointly construct a road condition factor R. The road condition factor is used to replace the length of the road. The specific modeling and calculation methods are as follows.

1) Each intersection is a point and each road is an edge.
2) Define the congestion coefficient of road i at time t as $C_i(t)$ shown in formula (10);

$$C_i(t) = \begin{cases} \dfrac{(f_i(t-1) + fin_i(t) - fout_i(t)) * L}{l_i * d_i}, \\ \quad if\ f_i(t-1) + fin_i(t) > fout_i(t) \\ 0,\ if\ f_i(t-1) + fin_i(t) \leq fout_i(t) \end{cases} \tag{8}$$

Here, $f_i(t-1)$ represents the number of vehicles on road i at the time $t-1$. $fin_i(t)$ represents the traffic flow on road i at time t. $fout_i(t)$ represents the traffic flow on road i at time t. The l_i represents the number of lanes on road i. d_i represents the length of road i. L represents the average length of each vehicle. If $f_i(t-1)+fin_i(t) - fout_i(t)$ is negative, then $C_i(t)$ is equal to zero.

3) Define the road condition factor of road i at time t as $R_i(t)$, as shown in formula (11).

$$R_i(t) = d_i * (1 + C_i(t)) \tag{9}$$

That is, the road condition factor of road i at time t is shown in formula (12).

$$R_i(t) = d_i * (1 + \frac{(f_i(t-1) + fin_i(t) - fout_i(t)) * L}{l_i * d_i}) \tag{10}$$

According to the "Urban Road Traffic Operation Evaluation Index System (Beijing Local Standard)" [34], the main measure of traffic operation level is the road traffic operation index TPI (Traffic Performance Index), which ranges from 0 to 10. According to the road traffic operation index, we divide the urban road congestion into five grades, namely unobstructed, basically unblocked, mild congestion, moderate congestion and severe congestion. The urban road congestion index TCI (Traffic Congested Index) is defined as in TABLE 2.

C. ALGORITHM STEPS

According to the basic ideas proposed in section 3.1, we design a dynamic path planning method based on an improved ant colony algorithm under congested road conditions. The steps are as follows.

1) Initialize the map information, including the number of intersections n and the number of roads m;
2) Set $t = 0$, and initialize $R(0) = \{R_1(0), R_2(2), \ldots, R_m(0)\}$;
3) Enter the starting and ending points $\{start, end\}$, initialize R_{max}, ΔR_{max}, $ar = 0$. Here, R_{max} represents the maximum value of the road condition factor. C is a constant and the value range is [0, 1], ΔR_{max} represents the maximum change of the road condition factor, and ΔR is calculated as shown in formula (13). Ar indicates the length of the path that has been traversed.

$$\Delta R_i = R_i(t+1) - R(t) \tag{11}$$

4) Introduce $R(0)$ into the PSO algorithm to replace the distance between two particles, and calculate the three parameters α, β, and ρ of the ACO algorithm to get the $best_{tour}$;
5) Introduce $R_i(t)$ into the ACO algorithm to replace d_i;
6) At time $t + 1$, calculate the values of R_i and ΔR_i. If $\Delta R_i > \Delta R_{max}$ or $R_i > R_{max}$, then use the ACO algorithm to get a new route new_{tour};
7) If $new_{tour} < best_{tour} - ar$, then $best_{tour} = new_{tour} + ar$, $ar = ar + R_i(t)$, $t = t + 1$;
8) If the path planning reaches the endpoint, output $best_{tour}$. Otherwise, return to step 5);

Remark: (1) Our ACO algorithm itself has certain limitations. (2) Both the algorithm of PSO to optimize the ACO's parameter and the algorithm dynamic planning path increase the complexity of the calculation to a certain extent. Correspondingly, (1) the limitations of ACO algorithm: Due to the large adjustment range of the ACO algorithm parameter ($0 \leq \alpha \leq 5$, $0 \leq \beta \leq 5$, $0 \leq \rho \leq 1$), inappropriate parameter selection will lead to the slow convergence, easy stagnation, and even longer searching time. (2) Using PSO to optimize ACO parameters indeed can greatly reduce the influence of the limitations of the ACO algorithm itself, but this optimization process itself requires calculation time and also increases the complexity of the calculation. Moreover, dynamic planning path method indeed improves the accuracy of the optimal solution, whereas it is a real-time planning route. During this process, the route is calculated multiple times. This process also requires calculation time as well as increases the computation complexity.

Besides, it is noted that in our scheme the cost effectiveness of our ACO algorithm is linear with the number of ants and intersections, and the times of iteration. When the iteration times reaches a certain value, the cost effectiveness of our scheme will tend to be stable. Hence, the computation complexity can be expressed as $\mathbb{O}(n * (n-1)^2 * m * T/2)$, where n denotes the amount of intersections, m represents the number of ants and T stands for the iteration times. In our experiment, the runtime of our experimental result refers to the time consumed from a random start-point to the end-point or the time required to generate the best path within the number of iterations. From the expression of computation complexity, it is easy to observe that when the number of ants and the number of intersections are constant, the computation complexity and runtime are positively related with the iteration times. In other words, the greater the number of iterations, the greater the computational complexity and the more time required.

V. SIMULATION EXPERIMENT

In this article, two simulation experiments are designed to verify the validity of a dynamic path planning method based on improved ant colony algorithm proposed in Section 3 on congested roads.The first kind of simulation experiment is *ACO* parameter optimization experiment based on *PSO* algorithm. It is mainly used to verify the effectiveness of *PSO* algorithm for *ACO* algorithm parameter optimization through comparative analysis of *ACO* algorithm parameter randomization and optimization experiment.The second simulation experiment is to select some intersections in Beijing area for dynamic path planning experiment. The *ACO* algorithm based on distance parameter and road condition factor is compared and analyzed to verify the effectiveness of the dynamic path planning method proposed in this article under congested traffic conditions.

We run our experiments with a Lenove server that owns 500GB storage space of hard disk and implements on Windows 10 operating system under Intel (R) Core (TM) i7-4710HQ CPU @2.50 GHz and 8GB RAM. In the experimental simulations for the ACO parameter randomization and parameter optimization experiments conducted in this article, the number of ants was set to 10, and five classical data sets of the TSPLIB database TSPLIB, eIL51, Berlin52, ST70, KROD100 and LIN105, were selected as test data sets [35].

A. ACO PARAMETER OPTIMIZATION EXPERIMENT BASED ON PSO ALGORITHM

In this article, two *ACO* parameter randomization experiments and one parameter optimization experiment are designed for comparison and verification. In the *ACO* parameter randomization experiment, the values of α, β, and ρ are randomly selected within the setting range, and the values of the three parameters will change with each iteration. In the *ACO* parameter optimization experiment, the values of the three parameters α, β, and ρ will be determined by the *PSO* algorithm before the iteration, and will not change during the iteration process. Because the ACO algorithm has a wide range of parameter values, and the parameter values affect the speed of convergence, in order to find the parameter setting with the fastest convergence, the α, β, and ρ parameters of ACO algorithm need to be optimized.

TABLE 3. The parameter randomization experiment 1.

Dataset	Eil51	Berlin52	St70	Krod100	Lin105
Average iteration number	169	183	176	201	198
Average best solution	485	8368	782	25439	16824
Best Know solution	426	7542	675	21294	14379
Deviation rate	14.08%	10.96%	15.96%	19.47%	17.01%

TABLE 4. The parameter randomization experiment 2.

Dataset	Eil51	Berlin52	St70	Krod100	Lin105
Average iteration number	123	132	126	159	139
Average best solution	467	8014	756	24333	15843
Best Know solution	426	7542	675	21294	14379
Deviation rate	9.78%	6.26%	12.01%	14.27%	10.1%

There are three measures in *ACO* parameter randomization experiments, including average iteration number $iteration_{aver}$, average best solution $result_{aver}$ and deviation rate $deviation_{rate}$. Here, the calculation of the deviation rate is shown in formula (12).

$$deviation_{rate} = \frac{|result_{aver} - result_{best}|}{result_{best}} * 100 \qquad (12)$$

In the first experiment of *ACO* parameter randomization, according to the conclusion obtained in the article by Jiang et al's work [36], the parameters are $0.1 \leq \alpha \leq 0.3$, $3 \leq \beta \leq 6$ and $0.7 \leq \rho \leq 0.9$. The accuracy of the three parameters are 0.1. In each experiment, the values of α, β and ρ are randomly selected within the setting range. When the maximum number of iterations reaches 1000 or the optimal solution does not change with 200 iterations, the experiment exits. The results of *ACO* parameter randomization experiment 1 are shown in TABLE 3.

In the second experiment of ACO parameter randomization, according to the conclusion obtained in Dorigo *et al*'s [12], [13] and Jiang *et al.*' s work [36], the parameters are $0.5 \leq \alpha \leq 1.5$, $1 \leq \beta \leq 5$ and $0.5 \leq \rho \leq 1$. The accuracy of the three parameters is 0.1. In each experiment, the values of α, β and ρ are randomly selected within the setting range. When the maximum number of iterations reaches 1000 or the optimal solution does not change with 200 iterations, the experiment exits. The results of the *ACO* parameter randomization experiment 2 are shown in TABLE 4.

In the *ACO* parameter optimization experiment, the *PSO* algorithm is used to train the three parameters of the *ACO* algorithm α, β and ρ for the different data sets within the value range. According to the conclusion from [12], [13] and [36], the range of parameters are $0.5 \leq \alpha \leq 1.5$,

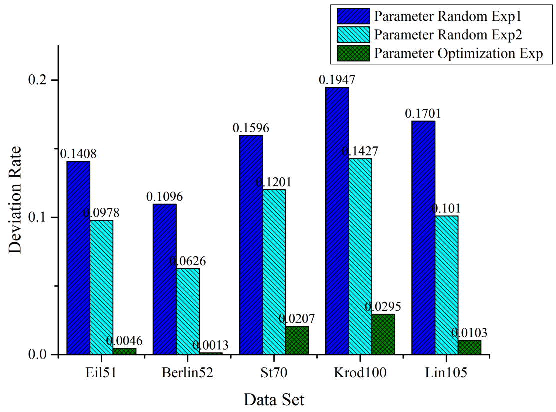


FIGURE 1. **The ACO algorithm parameter randomization and parameter optimization comparison experiment.**

$1 \leq \beta \leq 5$ and $0.5 \leq \rho \leq 1$. The accuracy of the three parameters is 0.1. In each experiment, the values of the three parameters α, β and ρ are determined by the *PSO* algorithm before the iteration, and it will not change during the experiment. When the maximum number of iterations reaches 1000 or the optimal solution does not change with 200 iterations, the experiment exits. The results of the *ACO* parameter optimization experiment are shown in TABLE 6.

The comparison results of *ACO* algorithm parameter randomization experiment 1, experiment 2, and parameter optimization experiment are shown in FIGURE 1.

From *ACO* algorithm parameter randomization experiment 1 and 2 shown in TABLE 3 and TABLE 4, it can be seen that the setting of parameters α, β and ρ has great influence on the experimental results. In the same number of iterations, there is a large gap between the average best solution and the known best solution, which requires multiple iterations to converge. Comparing with the results of *ACO* algorithm parameter randomization experiment 1 and experiment 2, it can be seen that the average optimal solution obtained is closer to the known best solution than experiment 1, because the experimental 2 parameter range is smaller than experiment 1 parameter range.

The disadvantage of *ACO* algorithm is that it is easy to produce stagnation phenomenon and the search time is long. For different path planning requirements, repeated experiments and debugging are needed to obtain the optimal α, β and ρ values and the best path planning results, but this will waste a lot of time and lead to low efficiency of the algorithm. In order to improve the effectiveness of *ACO* algorithm, *PSO* algorithm is used to optimize the α, β and ρ parameters of *ACO* algorithm.

From the comparison of *ACO* algorithm parameter optimization experiment and parameter randomization experiment shown in TABLE 4 and TABLE 5, it can be seen that under the same conditions, after *PSO* algorithm optimized parameters, the *ACO* algorithm used in path planning obtained the average best solution is better than the *ACO* algorithm experiment with parameter randomization, and the

TABLE 5. **The parameter optimization experiment.**

Dataset	Eil51	Berlin52	St70	Krod100	Lin105
Average iteration number	123	124	121	149	117
Average best solution	428	7552	689	21923	14528
Best Know solution	426	7542	675	21294	14379
Deviation rate	0.46%	0.13%	2.07%	2.95%	1.03%

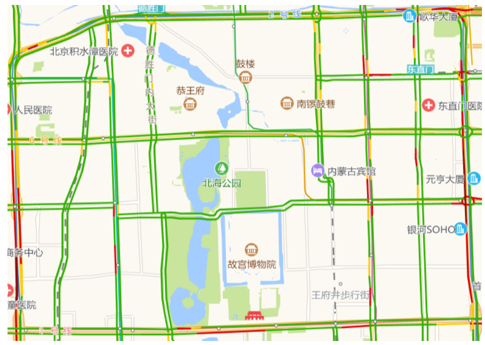


FIGURE 2. **Congested road network in Beijing.**

deviation rate is smaller.Therefore, it can be concluded that the *ACO* algorithm after *PSO* optimization has better effect and faster convergence speed than the ordinary *ACO* algorithm in path planning.

From the comparative analysis of the above three experiments, it can be seen that using the *PSO* algorithm to optimize the α, β and ρ parameters of the *ACO* algorithm can effectively improve the effectiveness and accuracy of the *ACO* algorithm shown in FIGURE 1. Therefore, the congestion proposed in this article In the dynamic path planning method based on the improved ant colony algorithm under road conditions, the *PSO* algorithm is first used to optimize the parameters of the *ACO* algorithm.

B. DYNAMIC PATH PLANNING EXPERIMENT BASED ON IMPROVED ANT COLONY ALGORITHM UNDER CONGESTED ROAD CONDITIONS

In order to verify the effectiveness of the proposed dynamic path planning method based on improved ant colony algorithm in real congestion environment, this article uses real traffic network and traffic flow data for simulation and comparison experiments.

According to the "2019Q3 China Major City Traffic Analysis Report" [37] released by Gaode Map, among the major cities in China, Beijing's road network has been the highest percentage of congested road mileage during peak hours, reaching 8.55%. The average speed in Beijing during peak hours is 25.65km/h. So we select some intersections in Beijing for dynamic path planning experiments based on improved ant colony algorithm under congested road conditions. The specific selection area is shown in FIGURE 2.

According to the map-related information provided by Gaode, the intersections of this area are regarded as points and the connectivity between the intersections is regarded as edges. According to the relative position and connection between the intersections, we give a graph in FIGURE 3.

TABLE 6. **The dynamic path planning experiment based on improved ant colony algorithm under congested road conditions (congestion coefficient follows uniform distribution).**

n	$tour\_init_{aver}$	$tour\_static_{aver}$	$tour\_aco_{aver}$	$tour\_opt_{aver}$	CR_static	CR_aco	CR_opt	CR_promote
100	1784	1957	1982	1824	9.69%	11.09%	2.24%	7.97%
200	1739	1904	1959	1786	9.48%	12.65%	2.70%	8.83%
500	1709	1862	1943	1748	8.95%	13.69%	2.28%	10.03%
1000	1690	1831	1934	1726	8.34%	14.43%	2.13%	10.75%
1406	1684	1820	1929	1715	8.07%	14.54%	1.84%	11.09%
average	1721	1874	1949	1759	8.91%	13.28%	2.23%	9.73%

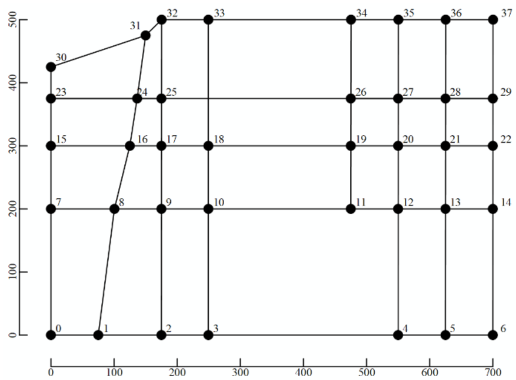


FIGURE 3. **The modeling of congested road networks in Beijing.**

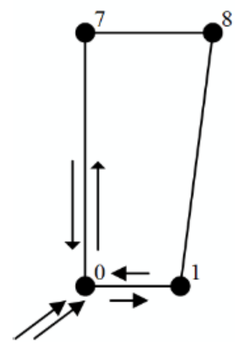


FIGURE 4. **The traffic generation at edge intersections.**

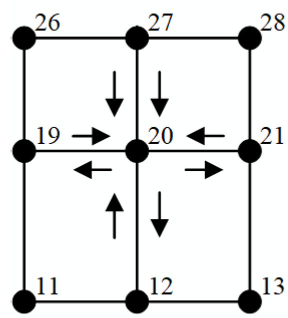


FIGURE 5. **The traffic generation at non-edge intersections.**

There are 38 intersections and 62 roads in this area. The X-axis range is [0,700] and the Y-axis range is [0,500]. The scale of the entire map is 1:10 meters. Hence, n in our experiment refers to the number of random selections of two intersections from 38 intersections as the experimental test of dynamic path planning. For example, $n = 500$ means 500 paths are randomly selected from the number of samples $A_{38}^2 = 1406$ for our experimental test, here one path refers to a road path from one start-point to one destination.

In this article, the intersections selected in the road network are divided into edge intersections and non-edge intersections. Edge intersections refer to intersections with other sources of traffic in the actual road network, including intersections 0, 1, 2, 3, 4, 5, 6, 7, 14, 15, 22, 23, 29, 30, 31, 32, 33, 34, 35, 36, and 37, the remaining intersections are non-edge intersections.

In the experiment, the traffic flow is randomly generated at each edge intersection, and the traffic flow at each intersection in the road network is affected by the traffic volume of the surrounding connected intersections. Taking the edge intersection 0 as an example, the inbound traffic volume at time t is equal to the sum of the traffic volume at intersection 1, intersection 7, and external inbound intersection (the ratio of inbound and outbound traffic flow is less than or equal to 1). The outbound traffic flow of at time t will go to intersection 1, intersection 7, and external intersection, according to a random ratio as shown in FIGURE 4. Taking the non-edge intersection 20 as an example, the inbound traffic at time t is equal to the sum of the traffic volume at intersection 12, intersection 19, intersection 21, and intersection 27. The outbound traffic at time t will go to intersection 12, intersection 19, intersection 21, and intersection 27, according to a random ratio (the ratio of inbound and outbound traffic flow is less than or equal to 1) as shown in FIGURE 5.

In the dynamic path planning experiment based on the improved ant colony algorithm under congested road conditions, firstly define the initial path *tour_init*, static path *tour_static*, dynamic path *tour_aco*, and dynamic path *tour_opt*. The four definitions are shown in Table 6.

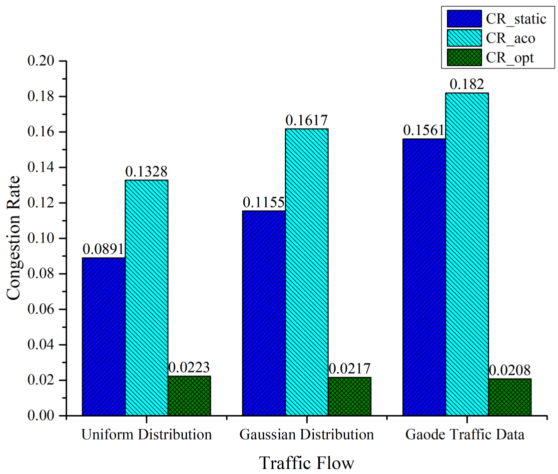


FIGURE 6. Comparative experiment of dynamic path planning based on improved ant colony algorithm under congested road conditions.

The initial path *tour_init* represents the sum of the road condition factors, which is calculated by the *PSO* algorithm during the initial path planning. The static path *tour_static* represents the sum of the road condition factors, which is obtained in the actual operation process. The dynamic path *tour_aco* represents the sum of the road condition factors, which is obtained by *ACO* algorithm whose parameters are optimized by *PSO*. The dynamic path *tour_opt* represents the sum of road condition factors, which is obtained by the dynamic path planning method based on an improved ant colony algorithm.

The comparison results of dynamic path planning experiments based on improved ant colony algorithm under congested road conditions are shown in FIGURE 6.

In the experiment, firstly the *ACO* algorithm's three parameters are obtained by the *PSO* algorithm. The parameters are $\alpha = 1$, $\beta = 2.5$, and $\rho = 0.8$. In each experiment, the number of ants is 10, different starting and ending points are randomly chosen and the number of iterations Q is 100. Each experiment is executed n times, and the sum of the road condition factors of the initial path *tour_init*, static path *tour_static*, dynamic path *tour_aco* and dynamic path *tour_opt* is obtained by averaging. The calculation formulas are shown as (13), (14), (15), and (16).

$$\text{tour\_init}_{\text{aver}} = \frac{\sum_{i=0}^{n} \text{tour\_init}(i)}{n} \tag{13}$$

$$\text{tour\_static}_{\text{aver}} = \frac{\sum_{i=0}^{n} \text{tour\_static}(i)}{n} \tag{14}$$

$$\text{tour\_aco}_{\text{aver}} = \frac{\sum_{i=0}^{n} \text{tour\_aco}(i)}{n} \tag{15}$$

$$t\text{our\_opt}_{\text{aver}} = \frac{\sum_{i=0}^{n} \text{tour\_opt}(i)}{n} \tag{16}$$

The evaluation indicators of dynamic path planning experiments based on an improved ant colony algorithm under congested road conditions are *CR_static*, *CR_aco*, *CR_opt* and $CR_p romote$. *CR_static* is a static path planning congestion rate. *CR_aco* is path planning congestion rate when parameters are optimized by the *ACO* algorithm. *CR_opt* is the congestion rate based on the improved ant colony algorithm. *CR_promote* is the congestion promotion rate. The calculation formulas are shown as (17), (18), (19) and (20).

$$CR\text{\_static} = \frac{|\text{tour\_static}_{\text{static}} - \text{tour\_init}_{\text{aver}}|}{\text{tour\_init}_{\text{aver}}} \tag{17}$$

$$CR\text{\_aco} = \frac{|\text{tour\_aco}_{aver} - \text{tour\_init}_{\text{aver}}|}{\text{tour\_init}_{\text{aver}}} \tag{18}$$

$$CR\text{\_opt} = \frac{|\text{tour\_opt}_{aver} - \text{tour\_init}_{\text{aver}}|}{\text{tour\_init}_{\text{aver}}} \tag{19}$$

$$CR\text{\_promote} = \frac{|\text{tour\_aco}_{aver} - \text{tour\_opt}_{\text{aver}}|}{\text{tour\_aco}_{\text{aver}}} \tag{20}$$

This article studies dynamic path planning under congested road conditions. Therefore, if the entire planning path is a clear path (ie. initial planning path = dynamic planning path, $C_i(t) \in [0, 4)$), it is considered as an invalid experiment result. There are three experiments in this article. In the first two experiments, the congestion coefficients of each road segment in the road network respectively follow the uniform distribution and normal distribution of [0, 1] and the congestion coefficients will change with time. In the third experiment, the congestion coefficient of each road segment in the road network was subject to the traffic situation which is provided by Gaode Maps during the peak working hours of the region at the working day. The experimental results are shown in TABLE 5, TABLE 7, and TABLE 8.

It can be seen from the experimental results in TABLE 6 that when the traffic congestion coefficient obeys the average distribution, PSO algorithm used for initial path planning will result in a congestion rate of 8.07% ∼ 9.69%; If taking the distance between the two points as the parameter of the ACO algorithm for dynamic path planning, a congestion rate of 11.09%∼14.54% is generated by using the ACO algorithm; If taking the road condition factor between the two points as the parameter, the dynamic path planning method of ant colony algorithm finally produced only 1.84%∼2.70% congestion rate. Compared with the ACO algorithm based on distance parameter, the congestion rate can be decreased by 7.97%∼11.09%.

It can easily be observed from the experimental results in TABLE 7 that when the traffic congestion coefficient obeys the normal distribution, PSO algorithm used for initial path planning will result in a congestion rate of 10.16% ∼ 12.16%; If also taking the distance between the two points as the parameter of the ACO algorithm for dynamic path planning, a congestion rate of 13.60%∼17.15% is generated by using the ACO algorithm; If taking the road condition factor between the two points as the parameter, the dynamic path planning method of ant colony algorithm finally produced only 1.95%∼2.48% congestion rate. Compared with the ACO algorithm based on distance parameter, the congestion rate can be decreased by 10.23%∼12.52%.

TABLE 7. The dynamic path planning experiment based on improved ant colony algorithm under congested road conditions (congestion coefficient follows normal distribution).

n	$tour\_init_{aver}$	$tour\_static_{aver}$	$tour\_aco_{aver}$	$tour\_opt_{aver}$	CR_static	CR_aco	CR_opt	CR_promote
100	1712	1886	1945	1746	10.16%	13.60%	1.95%	10.23%
200	1691	1880	1964	1728	11.17%	16.14%	2.18%	12.01%
500	1698	1898	1985	1732	11.77%	16.90%	2.00%	12.74%
1000	1698	1907	1988	1736	12.30%	17.07%	2.23%	12.67%
1406	1690	1899	1980	1732	12.36%	17.15%	2.48%	12.52%
average	1697	1894	1972	1734	11.55%	16.17%	2.17%	12.03%

TABLE 8. Dynamic path planning experiment based on improved ant colony algorithm under congested road conditions(congestion coefficient follows the traffic situation of Gaode map).

n	$tour\_init_{aver}$	$tour\_static_{aver}$	$tour\_aco_{aver}$	$tour\_opt_{aver}$	CR_static	CR_aco	CR_opt	CR_promote
100	1699	1950	2024	1727	14.77%	19.12%	1.64%	14.67%
200	1742	1998	2070	1800	14.69%	18.82%	3.32%	13.04%
500	1753	2037	2039	1791	16.20%	16.31%	2.16%	12.16%
1000	1758	2042	2080	1784	16.15%	18.31%	1.47%	14.23%
1406	1723	2003	2041	1754	16.25%	18.45%	1.79%	14.06%
average	1735	2006	2050	1771	15.61%	18.20%	2.08%	13.63%

It can be easily found from the experimental results in TABLE 8 that when the traffic congestion coefficient obeys the traffic situation of the Gaode map, PSO algorithm used for initial path planning will result in a congestion rate of 14.69%∼16.25%; If also taking the distance between the two points as the parameter of the ACO algorithm for dynamic path planning, a congestion rate of 16.31%∼19.12% is generated by using the ACO algorithm; If taking the road condition factor between the two points as the parameter, the dynamic path planning method of ant colony algorithm finally produced only 1.47%∼3.32% congestion rate. Compared with the ACO algorithm based on distance parameter, the congestion rate can be decreased by 12.16%∼14.67%.

As depicted in Figure 6, it can be summarized that the proposed dynamic path planning method based on improved ant colony algorithm under congested road conditions can make up for the existing path planning algorithm that does not consider traffic conditions and traffic congestion dynamic changes, and can improve congested road conditions.

To summarize, through the comparative analysis of the above three experiments, it can be seen that the use of distance parameter-based ACO algorithm for dynamic path planning produces a higher congestion rate, which indicates that in actual road conditions, the shortest path is not necessarily the optimal one. Moreover, we can easily conclude that the dynamic path planning method based on the improved ant colony algorithm proposed in this article under congested road conditions can dynamically avoid congested sections and improve the accuracy of obtaining the optimal solution. Also, it is suitable for dynamic path planning under congested road conditions. Moreover, compared with the *ACO* algorithm based on distance parameters, the average congestion rate can be reduced by about 9.73% to 13.63%.

VI. CONCLUSION

In this article, various attributes in a complex traffic environment, including road length, number of lanes, inbound traffic and outbound traffic, are quantified as road condition factors. In congested road conditions, the road condition factor can measure the cost of driving more accurately than the distance between intersections. In this article, the particle swarm optimization algorithm is used to optimize the parameters of the ant colony algorithm, so that the ant colony algorithm can converge faster and query more efficiently. This article also introduces the road condition factor into the parameter-optimized ant colony algorithm for dynamic path planning. The road condition factor is used to replace the road length in the traditional ant colony algorithm. The experimental results show that when the ant colony algorithm's parameters are optimized by particle swarm optimization algorithm, the ant colony algorithm converge globally faster and the trade off between the accuracy and efficiency of the ant colony algorithm is good. Moreover, the experimental results show that the proposed method can effectively improve the effectiveness and accuracy of path planning under congested road conditions and significantly reduce the congestion rate.

REFERENCES

[1] J. Yang, A. Purevjav, and S. Li, "The marginal cost of traffic congestion and road pricing: Evidence from a natural experiment in Beijing," *Amer. Econ. J., Econ. Policy*, vol. 12, no. 1, pp. 418–453, 2020.

[2] J. Sun, H. Xiong, H. Zhang, and L. Peng, "Mobile access and flexible search over encrypted cloud data in heterogeneous systems," *Inf. Sci.*, vol. 507, pp. 1–15, Jan. 2020.

[3] C. Lee, Y. Kim, S. Jin, D. Kim, R. Maciejewski, D. Ebert, and S. Ko, "A visual analytics system for exploring, monitoring, and forecasting road traffic congestion," *IEEE Trans. Vis. Comput. Graphics*, vol. 26, no. 11, pp. 3133–3146, Nov. 2020, doi: 10.1109/TVCG.2019.2922597.

[4] H. Huang, A. V. Savkin, and C. Huang, "Reliable path planning for drone delivery using a stochastic time-dependent public transportation network," *IEEE Trans. Intell. Transp. Syst.*, early access, Apr. 7, 2020, doi: 10.1109/TITS.2020.2983491.

[5] J. Sun, H. Xiong, S. Zhang, X. Liu, J. Yuan, and H. Deng, "A secure flexible and tampering-resistant data sharing system for vehicular social networks," *IEEE Trans. Veh. Technol.*, early access, Aug. 11, 2020, doi: 10.1109/TVT.2020.3015916.

[6] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Math.*, vol. 1, no. 1, pp. 269–271, Dec. 1959.

[7] B. Roy, "Transitivité et Connexité," *Comptes Rendus Hebdomadaires Des Seances De L Academie Des Sciences*, vol. 249, no. 2, pp. 216–218, 1959.

[8] B. Seet, G. Liu, B. Lee, C. H. Foh, K. J. Wong, and K. K. Lee, "A-STAR: A mobile ad hoc routing strategy for metropolis vehicular communications," in *Proc. Int. Conf. Res. Netw.*, in Lecture Notes in Computer Science, vol. 3042. Berlin, Germany: Springer, 2004, pp. 989–999.

[9] A. Stentz, "The focussed d^* algorithm for real-time replanning," *IJCAI*, vol. 95, pp. 1652–1659, Aug. 1995.

[10] J. J. Grefenstette, "Genetic algorithms and machine learning," in *Proc. 6th Annu. Conf. Comput. Learn. Theory (COLT)*, 1993, pp. 3–4.

[11] 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.

[12] M. Dorigo and G. D. Caro, "Ant colony optimization: A new meta-heuristic," in *Proc. Congr. Evol. Comput.*, vol. 2, Jul. 1999, pp. 1470–1477.

[13] M. Dorigo, M. Birattari, and T. Stutzle, "Ant colony optimization," *IEEE Comput. Intell. Mag.*, vol. 1, no. 4, pp. 28–39, Nov. 2006.

[14] K. E. Parsopoulos and M. N. Vrahatis, "Particle swarm optimization method in multiobjective problems," in *Proc. ACM Symp. Appl. Comput. (SAC)*, 2002, pp. 603–607.

[15] J. Kennedy and R. Eberhart, "Particle swarm optimization (PSO)," in *Proc. IEEE Int. Conf. Neural Netw.*, Perth, WA, Australia, Nov./Dec. 1995, pp. 1942–1948.

[16] R. Eberhart and K. James, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Netw.*, Perth, WA, Australia, vol. 4, Dec. 1995, pp. 1942–1948.

[17] M. Dorigo and L. M. Gambardella, "Ant colonies for the travelling salesman problem," *Biosystems*, vol. 43, no. 2, pp. 73–81, Jul. 1997.

[18] W. Zhangqi, Z. Xiaoguang, and H. Qingyao, "Mobile robot path planning based on parameter optimization ant colony algorithm," *Procedia Eng.*, vol. 15, no. 12, pp. 2738–2741, Aug. 2011.

[19] 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.

[20] R. Gan, Q. Guo, H. Chang, and Y. Yi, "Improved ant colony optimization algorithm for the traveling salesman problems," *J. Syst. Eng. Electron.*, vol. 21, no. 2, pp. 329–333, Apr. 2010.

[21] X. Li, P. Tian, J. Hua, and N. Zhong, "A hybrid discrete particle swarm optimization for the traveling salesman problem," in *Proc. Asia–Pacific Conf. Simulated Evol. Learn.* Berlin, Germany: Springer, 2016, pp. 181–188.

[22] A. K. M. F. Ahmed and J. U. Sun, "An improved particle swarm optimization algorithm for the travelling salesman problem," *Adv. Sci. Lett.*, vol. 22, no. 11, pp. 3318–3322, Nov. 2016.

[23] Y. Zhang and J. Su, "Solving TSP problem based on hybrid algorithm of ant colony and particle swarm optimization," *China Sci. Technol. Inf.*, vol. 2, no. 2, pp. 260–261, 2008.

[24] 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.

[25] K. Min, H. Ge, and Y. Zhang, "A hybrid algorithm of ant colony and particle swarm optimization on solving TSP problem," *J. Jilin Univ.*, vol. 24, no. 4, pp. 402–405, 2006.

[26] B. Cai and D. Liu, "An application of ant colony algorithm based on particle swarm optimization in TSP," *J. Comput. Simul.*, vol. 26, no. 8, pp. 89–91, 2009.

[27] K. Sun, H. Wu, and H. Wang, "Hybrid algorithm of ant colony and particle swarm to solve TSP problem," *J. Comput. Eng. Appl.*, vol. 48, no. 34, pp. 60–63.

[28] X. Xie and P. Wu, "Research on the optimal combination of ACO parameters based on PSO," in *Proc. Int. Conf. Netw. Digit. Soc.*, vol. 1, May 2010, pp. 94–97.

[29] C.-X. Shi, B. Ying-Yong, L. Zi-Guang, and T. Jun, "Solving path planning problem by an ACO-PSO hybrid algorithm," in *Proc. Intell. Syst. Knowl. Eng. (ISKE)*. Paris, France: Atlantis Press, 2007, pp. 1–4.

[30] Y. Gigras, K. Choudhary, and K. Gupta, "A hybrid ACO-PSO technique for path planning," in *Proc. 2nd Int. Conf. Comput. Sustain. Global Develop. (INDIACom)*, Mar. 2015, pp. 1616–1621.

[31] D. Pal, P. Verma, D. Gautam, and P. Indait, "Improved optimization technique using hybrid ACO-PSO," in *Proc. 2nd Int. Conf. Next Gener. Comput. Technol. (NGCT)*, Oct. 2016, pp. 277–282.

[32] A. J. Ouyang and Y. Q. Zhou, "An improved PSO-ACO algorithm for solving large-scale TSP," *Adv. Mater. Res.*, vols. 143–144, pp. 1154–1158, Oct. 2010.

[33] B. Shuang, J. Chen, and Z. Li, "Study on hybrid PS-ACO algorithm," *Appl. Intell.*, vol. 34, no. 1, pp. 64–73, 2011.

[34] [Online]. Available: http://www.jianbiaoku.com/webarbs/book/57558/1062191.shtml

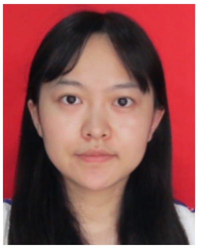
[35] [Online]. Available: http://elib.zib.de/pub/mp-testdata/tsp/tsplib/tsp/index.html

[36] Y. Jiang, J. Zhang, and S. Zhong, "Parameter analysis of ant colony algorithm," *J. Comput. Eng. Appl.*, vol. 20, no. 3, pp. 31–36, Sep. 2007.

[37] [Online]. Available: https://report.amap.com/download.do

CHUNJIANG WU is currently pursuing the Ph.D. degree with the School of Information and Software Engineering, UESTC. His research interests include intelligent transportation system (ITS), peer-to-peer networks, sensor networks, and big data.

SHIJIE ZHOU (Member, IEEE) is currently a Full Professor and the Dean of the School of Information and Software Engineering, UESTC. His research interests include communication and security in computer networks, peer-to-peer networks, sensor networks, cloud security, and big data. He has published many top journals and conferences, such as the IEEE Transactions on Parallel and Distributed Systems, the IEEE Internet of Things Journal, the IEEE Transactions on Industrial Informatics, SIGMOD, the IEEE Transactions on Biomedical Engineering, and GLOBECOM.



LICAI XIAO is currently pursuing the master's degree with the School of Information and Software Engineering, UESTC. Her research interests include intelligent transportation systems (ITSs) and sensor networks.