Research on intelligent city parking guidance method based on ant colony algorithm

Feng Gao

School of Artificial Intelligence,
Chongqing University of Arts and Sciences,
Chongqing 402160, China
and
Lyceum of the Philippines University,
Batangas Campus,
Batangas City, Philippines
Email: gaof619@163.com

Aifeng Chen*

College of Computer Science,
Sichuan University,
Chengdu 610065, China
Email: aifengchen@21cn.com
*Corresponding author

Abstract: In order to get the most satisfactory parking space at the fastest speed, an intelligent urban parking guidance method based on ant colony algorithm is proposed. The main factors affecting the selection of parking spaces in parking lots are analysed, including walking distance, driving distance, walking time, driving time and so on. Each factor is set as multiple attributes of berth, and the optimal berth selection model of smart city is established. Ant colony algorithm is used to solve the model, obtain the optimal parking space, and realise intelligent guidance of intelligent city parking. The simulation results show that the proposed method is feasible and effective.

Keywords: ant colony algorithm; ACO; simulation; parking; intelligent guidance.

Reference to this paper should be made as follows: Gao, F. and Chen, A. (2022) 'Research on intelligent city parking guidance method based on ant colony algorithm', *Int. J. Innovative Computing and Applications*, Vol. 13, No. 1, pp.1–10.

Biographical notes: Feng Gao completed his Master's in Software and Theory from the Chongqing University, in 2011. He is currently working toward his PhD at the Lyceum of the Philippines University. He is a senior experimenter of the School of Artificial Intelligence, Chongqing University of Arts and Sciences. His research direction addresses computer science and technology, large data, artificial intelligence and virtual reality.

Aifeng Chen received her PhD in Computer Science from the College of Computer Science, Sichuan University, in 2007. She is a Full Professor in the College of Computer Science, Sichuan University. Her research interests include computer science and its application in engineering, intelligent computation and AI.

1 Introduction

With the development of the city and the increasing number of parking lots, some operators began to install parking guidance system in the parking lot. At present, most cities have installed parking guidance system in newly built parking lots. But in the actual operation process, there are some difficulties, such as users are not satisfied with the parking spaces provided by the system, and research for new parking spaces, which makes the whole system unable to play its role. The main reason is that the above system randomly provides parking space for users without considering the satisfaction of users. Therefore, the parking guidance system also needs to solve the problem of optimal parking space selection, so that users can be satisfied with the selected parking space and rely on the system, so as to increase the reliability and utilisation of the whole system.

At this stage, most of the research on parking guidance system focuses on the technical implementation level. This system is only a part of the whole parking intelligent management system. It mainly studies the parking guidance system and related workflow. Zhao et al. (2014) presents a feasible parking guidance method. The parking planning problem is transformed into a linear assignment problem. We take vehicles as jobs and parking spaces as agents. The distance between the vehicle and the parking space is considered as the cost of the agent's work. Then, an algorithm is designed to solve the parking planning problem. However, this method is too complex to be widely used. Ni et al. (2018) presents an intelligent parking navigation system (P-SPAN) with efficient navigation result retrieval function. P-SPAN enables cloud computing to guide vehicles to empty parking spaces at their destination based on real-time parking information without revealing any personal information about drivers. This method has good effect, but its anti-jamming ability is poor in practical application. Jeong and Park (2017) study the characteristics of parking guidance system. Four hundred seventy parking lot management system applications were identified and analysed. The characteristics of parking guidance management system are summarised. On this basis, the intelligent parking ecosystem is introduced in Lin et al. (2017a). By identifying their functions and existing problems, a comprehensive and thoughtful classification is proposed, and three macro-themes of information collection, system deployment and service dissemination are carried out. However, the method lacks scientific verification and guidance.

In view of the existing research results at this stage, most of the studies only focus on micro-behaviour, and the research on parking guidance strategy is relatively few. Based on the traditional methods and the above research results, an intelligent city parking guidance method based on ant colony algorithm (ACO) is proposed from the perspective of parkers.

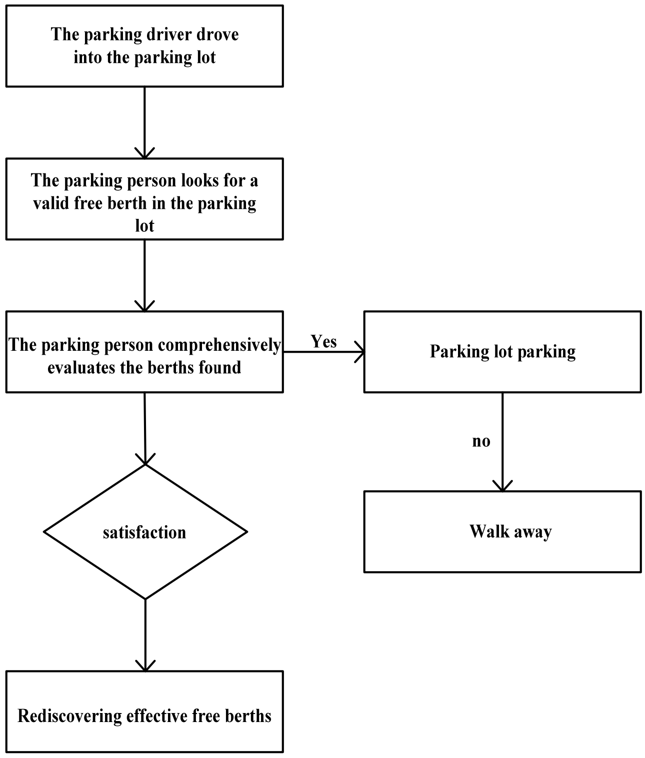
2 Analysis of intelligent guidance method for parking in smart city

2.1 Establishment of optimal berth selection model

At present, the existing intelligent city parking guidance system focuses on indoor guidance, but there are still some shortcomings in the specific parking guidance after vehicles enter the parking lot. Usually, it will cause users to park blindly. Long-term cumulative, will make the driver stay in the parking lot for a long time, and have a negative impact on the traffic order of the whole parking lot. But the intelligent city parking guidance system based on ACO can effectively reduce the occurrence of the above situation. The Central Management Office of the whole system will use data acquisition module to transmit real-time vehicle operation information to the system, through the acquisition module to achieve the selection of the best parking space, and through calculation to obtain the shortest parking path, to achieve intelligent city parking guidance (Piao, 2016). The research in this section is mainly from the point of view of parkers, considering all kinds of impacts caused by berth selection, and establishing the optimal berth selection model of smart city. Under the environment of the internet of things, vehicles will be guided dynamically to obtain the optimal guiding path.

Intelligent city parking belongs to micro parking behaviour, which is the final parking result obtained by parkers after detailed analysis and calculation in the parking lot. In the absence of guidance, Figure 1 is used to give a specific parking flowchart.

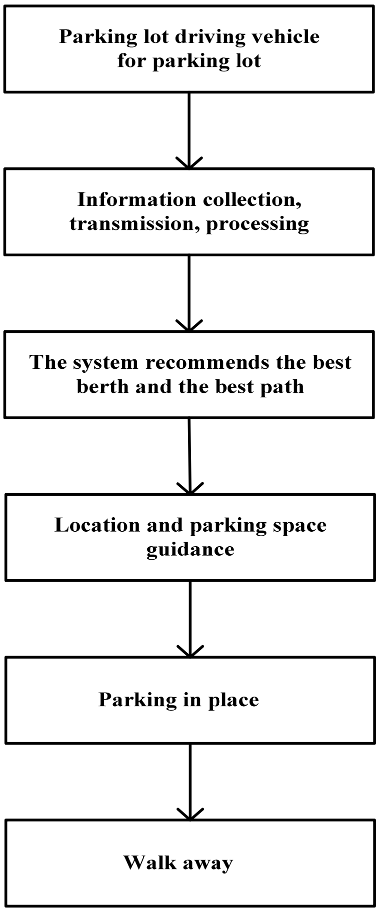
Figure 1 Parking flowchart without onsite guidance in parking lot



With the continuous development of the city, parking lots are expanding. If parkers cannot make quick judgements in a relatively short time, they need to continue to search for idle parking spaces in the parking lot. The above-mentioned operation process increases parking time for parkers, but also affects the traffic order of the whole parking lot (Li et al., 2018; Qi et al., 2017).

The detailed structure diagram of the berth guidance system in the internet of things environment is given by using Figure 2. After the driver enters the parking lot, the system needs to collect the relevant information of the vehicle by using the corresponding acquisition module. The relevant technology is adopted to transmit the vehicle information to the information processing centre. The system will provide the corresponding berth for the user according to the relevant information of the vehicle and make use of the voice of the system. Tip system and other publishing vehicle parking information (Liu et al., 2018).

Figure 2 Parking flow guided in parking lot



The analysis of the flowchart shows that the driver can effectively reduce the parking time under the guidance of the system, on the premise that the driver must completely follow the guidance method given by the system. However, in most cases, some drivers will be dissatisfied with the guidance scheme of the system and find a new berth, so the whole system will lose its significance (Gong et al., 2016). In order to provide drivers with the most satisfactory berth, it is necessary to analyse the different influencing factors from the driver's point of view, so as to obtain the best guidance scheme.

The following is a comprehensive analysis of several key factors from the perspective of parkers:

1 Driving time

The total travel time of the vehicle is from the driver's entry into the parking lot to the final parking position. Among them, the final parking position of the driver's vehicle and the driving path of the vehicle in the parking lot will have an impact on the driving time of the driver (Wang et al., 2017; Yuan et al., 2018). In practice, most drivers want to reach the target position at the fastest speed.

2 Walking time

That is, the time required for the driver to walk from the entrance to the exit of the parking lot, in which the driver needs to consider the following two factors: the final parking position of the vehicle and the length of the driver's walking path. Relevant survey results show that some parkers prefer to use longer driving time to reduce the driving time.

3 Driving distance

That is, the driving distance from the driver entering the parking lot to the final parking position, where the length of the whole path is related to the target parking space and the final parking position. Usually, the driver wants to reach the parking position (Lin et al., 2017b) with the shortest distance.

4 Walking distance

That is, the length of the path from the parking lot to the exit, usually the shortest distance the driver wants to walk.

5 Storage difficulty

The difficulty of parking in storage mainly refers to the difficulty of parking in storage. Usually, the type of parking is related to parking location, parking type and driver's proficiency. Usually, the hardest way to enter a warehouse is a zigzag berth, while the hardest way to enter a three-dimensional slanted berth is the smallest. The types of berths are given in detail in Figure 3.

6 Safety

The safety of parking is closely related to the size, orientation and monitoring equipment of parking space. Usually, the closer the berth is to the monitoring location, the higher the safety factor of the berth; conversely, it indicates that the berth may have potential safety hazards.

7 Shading

Outdoor parking lot in summer and noon, parkers first choose the shade location to park vehicles (Zhao et al., 2018).

If there are m alternatives in the multi-attribute problem, the alternatives are recorded as $A = \{a_1, a_2, \cdots, a_m\}$, which contains n decision-making indicators. The set of indicators is $F = \{f_1, f_2, \cdots, f_n\}$. The decision-making value of the j^{th} decision-making indicator f_j is expressed by x_{ij}, and the decision-making matrix composed of n indicators of m alternatives is given by formula (1):

$$X = \left(x_{ij}\right)_{m\times n} = \begin{bmatrix} x_{11}, x_{12}, \ldots, x_{1n} \\ x_{21}, x_{22}, \ldots, x_{2n} \\ \vdots \\ x_{1n}, x_{2n}, \ldots, x_{mn} \end{bmatrix} \tag{1}$$

The standardisation of decision-making indicators is mainly to convert the change indicators of different dimensions into non-dimensional standardised indicators (Huang and Xiao, 2018; Ardalan et al., 2020). The following details are given:

1 Linear proportional transformation method (Jia et al., 2018):

$$y_{ij} = \frac{x_{ij}}{\max_{1\le i\le m} x_{ij}}, \max_{1\le i\le m} x_{ij} \ne 0, i \in I_1 \tag{2}$$

$$y_{ij} = \frac{\min_{1\le i\le m} x_{ij}}{x_{ij}}, x_{ij} \ne 0, i \in I_2 \tag{3}$$

The linear proportional standardisation matrix is given by using the following formulas:

$$Y = (y_{ij})_{m\times n} = \begin{bmatrix} y_{11}, y_{12}, y_{13}, \cdots, y_{1n} \\ y_{21}, y_{22}, y_{23}, \cdots, y_{2n} \\ \vdots \\ y_{m1}, y_{m2}, y_{m3}, \cdots, y_{mn} \end{bmatrix} \tag{4}$$

2 Range transformation method:

$$y_{ij} = \frac{x_{ij} - \min_{1\le i\le m} x_{ij}}{\max_{1\le i\le m} x_{ij} - \min_{1\le i\le m} x_{ij}}, i \in I_1 \tag{5}$$

$$y_{ij} = \frac{\max_{1\le i\le m} x_{ij} - x_{ij}}{\max_{1\le i\le m} x_{ij} - \min_{1\le i\le m} x_{ij}}, i \in I_2 \tag{6}$$

The normalised matrix of range transformation is given by using the following formula, which is called matrix:

$$Y = (y_{ij})_{m\times n} = \begin{bmatrix} y_{11}, y_{12}, y_{13}, \cdots, y_{1m} \\ y_{21}, y_{22}, y_{23}, \cdots, y_{2m} \\ \vdots \\ y_{m1}, y_{m2}, y_{m3}, \cdots, y_{mn} \end{bmatrix} \tag{7}$$

In order to ensure the validity of the whole calculation method, it is necessary to ensure the differences among the distribution coefficients. The following theoretical studies on adding distance function to the algorithm show the above differences.

If the subjective weights (Xiong et al., 2017) calculated by subjective and objective weighting methods are w_i' and w_i'', $d(w_i', w_i'')$ is used to represent the distance function between the two, then there are:

$$d(w_i', w_i'') = \left[\sum_{i=1}^{n} (w_i', w_i'')^2\right] \tag{8}$$

If the combination weight is set to w_i', i.e., w_i is a linear weight of w_i' and w_i'', then:

$$w_i = \alpha w_i' + \beta w'' \tag{9}$$

Among them, α represents the distribution coefficient of subjective weight and β represents the distribution coefficient of objective weight.

In order to maintain the difference between the above two different coefficients, then:

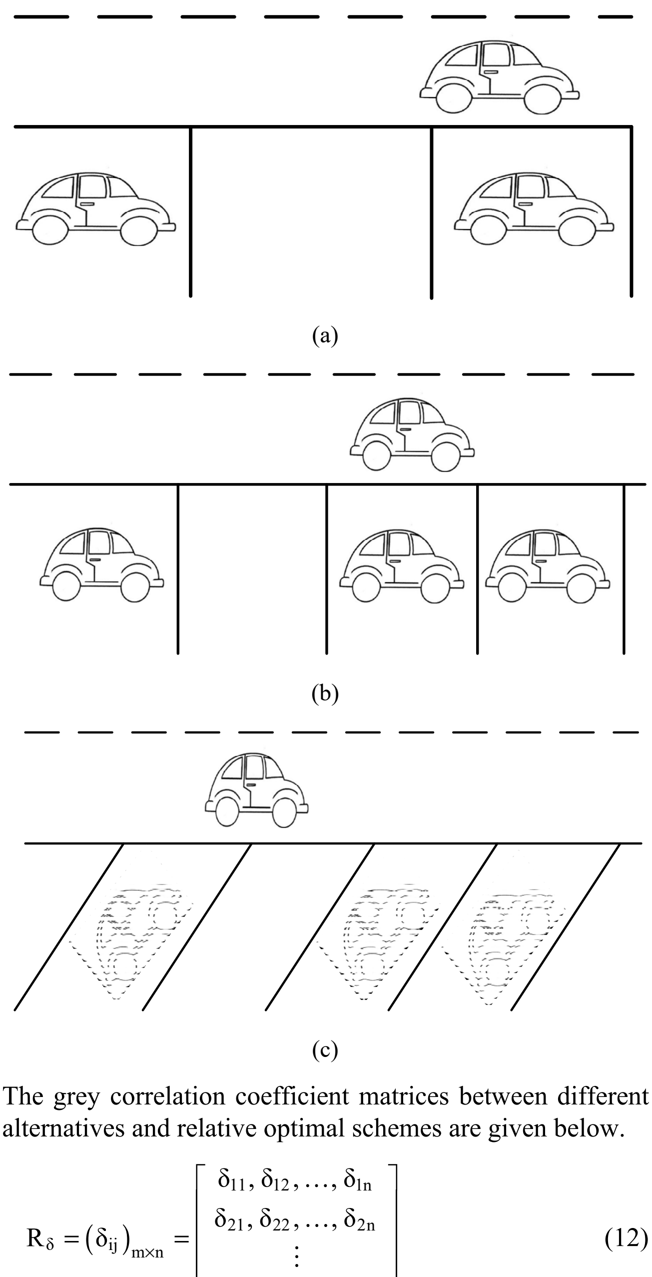
$$d(w_i', w_i'')^2 = (\alpha - \beta)^2 \tag{10}$$

If the relative optimal scheme is set to be $\alpha_0 = (x_{01}, x_{02}, \cdots, x_{0n})$, and the decision values of the j^{th} decision-making index f_i of the alternative and the optimal scheme are set to be x_{ij} and x_{0j}, respectively, the relationship between the different decision-making values is given by using the following formula:

$$\delta_{ij} = \frac{\min_i \min_j |x_{oj} - x_{ij}| + \rho \max_i \max_j |x_{oj} - x_{ij}|}{|x_{oj} - x_{ij}| + \rho \max_i \max_j |x_{oj} - x_{ij}|} \tag{11}$$

In the above formula, ρ represents the resolution coefficient, and its value range is within the range of [0, 1].

Figure 3 Types of berths, (a) one-shaped berth (b) multi-shaped berths (c) oblique berth



The grey correlation coefficient matrices between different alternatives and relative optimal schemes are given below.

$$R_\delta = (\delta_{ij})_{m\times n} = \begin{bmatrix} \delta_{11}, \delta_{12}, \ldots, \delta_{1n} \\ \delta_{21}, \delta_{22}, \ldots, \delta_{2n} \\ \vdots \\ \delta_{m1}, \delta_{m2}, \ldots, \delta_{mn} \end{bmatrix} \tag{12}$$

The difference between different indexes is analysed, and the correlation matrix is weighted. The combination weight vector is set as $w = (w_1, w_2, \cdots, w_n)$. The grey correlation decision matrix after weighting is given below.

$$Z = \left(z_{ij}\right)_{m\times n} = R_\delta W = \begin{bmatrix} \delta_{11}w_1, \delta_{12}w_2, \ldots, \delta_{1n}w_n \\ \delta_{21}w_1, \delta_{22}w_2, \ldots, \delta_{2n}w_n \\ \vdots \\ \delta_{m1}w_1, \delta_{m2}w_2, \ldots, \delta_{mn}w_n \end{bmatrix} \tag{13}$$

If all the attribute values of different alternatives are set as the optimal values of alternatives, the scheme is called ideal solution, and vice versa, the negative ideal solution. At present, the existing schemes do not have the above two solutions. The following needs to compare the actual scheme with the ideal solution of grey correlation. If there is an optimal solution (Jin et al., 2018), then it is set as the optimal scheme, that is:

$$V^+ = \left[\max R_{ij}w_j\right] = \left[z_1^+, z_2^+, z_1^+, \cdots, z_n^+\right] \tag{14}$$

$$V^- = \left[\min R_{ij}w_j\right] = \left[z_1^-, z_2^-, z_1^-, \cdots, z_n^-\right] \tag{15}$$

Computing the Euclidean distance spaces of alternatives and grey relational ideal solutions and negative ideal solutions:

$$D_1^+ = \sqrt{\sum\nolimits_{j=1}^{n}\left(z_{ij} - z_j^+\right)}, i = 1, 2, \cdots, m \tag{16}$$

$$D_1^- = \sqrt{\sum\nolimits_{j=1}^{n}\left(z_{ij} - z_j^-\right)}, i = 1, 2, \cdots, m \tag{17}$$

Compute the relative closeness of alternatives and grey relational ideal solutions:

$$C_i = \frac{D_i^-}{D_i^+ + D_i^-}, i = 1, 2, \cdots, m \tag{18}$$

When $z_{ij} = z_j^+, C_i = 1,$ the alternative approaches the ideal solution; if $z_{ij} = z_j^-, C_i = 0,$ the alternative approaches the negative ideal solution completely.

1 All the important information such as driving distance and walking distance can be queried in the system. In the parking peak stage of the parking lot, a large number of vehicles will enter the parking lot in a certain period of time. If any path of the parking lot is blocked, the driving time of the vehicle will increase. Therefore, it is necessary to use the following functions to calculate the dynamic travel time of the vehicle in detail (Ge et al., 2016), then:

$$t_{ij} = t_{0ij}\left[1 + \alpha\left(\frac{q_{ij}}{C_{ij}}\right)\right]^\beta \tag{19}$$

In the above formula, t_{ij} represents the dynamic travel time of vehicles on the road section (i, j), t_{0ij} represents the free flow time on the road section (i, j), q_{ij} represents the total number of vehicles encountered before reaching the final berth, C_{ij} represents the capacity of the road section, α and β represent the coefficients that are not calibrated, and need to be calibrated by fitting a large number of measured data.

2 Walking time is closely related to walking distance, main body age and driver's physical fitness. The following formulas can be used to calculate walking time:

$$t_w = \frac{d_w}{v_w} \tag{20}$$

In the above formula, t_w represents the walking distance of an arbitrary path, d_w represents the total length of the path and v_w represents the walking speed of an arbitrary parking user.

Dijkstra algorithm is needed to solve the shortest path problem, but the use of Dijkstra algorithm will increase the amount of calculation of the whole algorithm, the most important reason is that the algorithm needs to search the remaining nodes again, and the efficiency of the whole algorithm is not guaranteed (Li et al., 2016; Yan et al., 2016). In the whole calculation process of the model, the shortest path needs to be calculated without considering other nodes. On the basis of traditional methods, Dijkstra algorithm needs to be improved comprehensively. The termination condition of Dijkstra algorithm is set as follows: assuming the target node appears, the whole algorithm stops computing; on the contrary, it continues to calculate.

The selection of berth is a decision-making problem. There are many factors to be considered in the whole process, and there are certain correlations among different factors. Some of them cannot be described in detail by quantitative way. In order to better solve the above problems, the problem needs to be transformed. The most effective method is the analytic hierarchy process (AHP), which can describe the relationship between different variables in detail, describe the whole decision-maker's thinking through less information, and the calculation process of the whole method is very concise and rapid. Therefore, the following needs to use AHP to subjectively empower different decision-making indicators.

There are many different situations when the driver enters the parking lot for parking. The following needs to obtain different weight vectors within the prescribed range and check their consistency:

$$CI = \frac{\lambda - n}{n - 1} \tag{21}$$

$$CR = \frac{CI}{RI} \tag{22}$$

Among them, CI represents consistency index, CR represents consistency ratio, RI represents arbitrary consistency index, λ represents the maximum eigenvalue of matrix and n represents the number of decision indicators.

The matrix is standardised by the following formula:

$$P_{ij} = \frac{y_{ij}}{\sum\nolimits_{i=1}^{m} y_{ij}} \tag{23}$$

The following formulas are used to calculate the entropy value and the coefficient of difference of index j.

$$e_j = -k\sum_{i=1}^{m} p_{ij} \tag{24}$$

$$g_j = 1 - e_j \tag{25}$$

Different weight indexes w_j are determined by the following formulas:

$$w_j = \frac{g_j}{\sum_{i=1}^{m} g_j} \tag{26}$$

The following decision matrix is constructed for alternatives:

$$X = \left(x_{ij}\right)_{6\times3} = \begin{bmatrix} 39, 20, 9 \\ 59, 11, 7 \\ 63, 20, 3 \\ 69, 30, 7 \\ 46, 24, 4 \\ 37, 12, 4 \end{bmatrix} \tag{27}$$

The following comparison matrices are constructed by AHP:

$$A = \begin{bmatrix} 1, \frac{1}{3}, \frac{1}{2} \\ 3, 1, 2 \\ 2, \frac{1}{2}, 1 \end{bmatrix} \tag{28}$$

Normalisation of matrices in formula (27) includes:

$$R_{6\times3} = \left(r_{ij}\right)_{6\times3} = \begin{bmatrix} 0.211, 0.143, 0.091 \\ 0.139, 0.260, 0.116 \\ 0.130, 0.143, 0.271 \\ 0.119, 0.096, 0.116 \\ 0.179, 0.119, 0.230 \\ 0.222, 0.293, 0.203 \end{bmatrix} \tag{29}$$

The weighted coefficient matrix is given below.

$$Z_{6\times3} = \left(\delta_{ij} w_j\right)_{6\times3} = \begin{bmatrix} 0.191, 0.298, 0.232 \\ 0.091, 0.466, 0.255 \\ 0.082, 0.298, 0.319 \\ 0.072, 0.213, 0.255 \\ 0.138, 0.257, 0.300 \\ 0.215, 0.439, 0.300 \end{bmatrix} \tag{30}$$

On the basis of the above, the optimal berth selection model of smart city is established.

$$\log\left(\delta_{ij} w_j\right) = {k\sum_{i=1}^{m} p_{ij}} \Big/ {CR}. \tag{31}$$

2.2 *Intelligent guidance method for smart city parking based on ACOs*

Because the parking guidance problem is multi-path optimisation, it needs a better algorithm to solve the problem. ACO is a probabilistic algorithm used to find optimal paths. ACO is a kind of distributed computation, which can express the mathematical model as follows: in the process of foraging, each ant uses the concentration of pheromone left on the path and the probability of node state transition to achieve the search function of the shortest path. At any time, the probability of ant k turning from node i to node j can be calculated by using the following formula (Wang et al., 2018):

$$p_{ij}^{k} = \begin{cases} \dfrac{\left[\tau_{ij}(t)\right]^{\beta} \cdot \left[\eta_{ij}(t)\right]^{\beta}}{\sum_{s\in i}\left[\tau_{ij}(t)\right]^{\beta} \cdot \left[\eta_{ij}(t)\right]^{\beta}}, & j \in \text{allowed} \\ 0, & \text{otherwise} \end{cases} \tag{32}$$

In the formulas above, p_{ij}^{k} represents the probability that ant k will transfer from node i to node j in t-time period, and allowed represents the set of nodes selected by ant allowed in the next step, which does not contain obstacle nodes.

After completing the first cycle update, all the ants that have completed the path search task need to update all the path pheromones that the ants have searched. Detailed update rules are given below.

$$\tau_{ij}(t+1) = (1-\rho)\cdot\tau_{ij}(t) + \Delta\tau_{ij}(t, t+1) \tag{33}$$

$$\Delta\tau_{ij}(t, t+1) = \sum_{ij}^{k}(t, t+1) \tag{34}$$

$$\Delta\tau_{ij}^{k}(t, t+1) = \begin{cases} \dfrac{Q}{L_k}, & \text{suppose that the } k \text{ ant passes} \\ & \text{through this cycle (i, j)} \\ 0, & \text{otherwise} \end{cases} \tag{35}$$

In the formula above, $\Delta\tau_{ij}^{k}$ represents the information increment of ants on the path (i, j) and ρ represents the volatilisation coefficient of pheromones.

In order to solve the shortcomings of traditional optimisation methods such as slow convergence speed and premature phenomena, the ACO is improved by adding local pheromone and global updating strategy.

In general, the reciprocal of the Euclidean distance between node i and next node j is used to set the heuristic function factor η_{ij} = 1 / d(i, j). However, the heuristic function at this stage only considers the distance relationship between two nodes, which results in blindness in the process of path search (Xiang and Pan, 2016). Therefore, the Euclidean distance between node j and target node j is set as a heuristic factor function.

$$d(j, g) = \sqrt{\left(x_j - x_g\right)^2 + \left(y_j - y_g\right)^2} \tag{35}$$

$$\eta_{jg} = 1 / d(j, g) \tag{36}$$

In the above formula, (x_i, y_i), (x_g, y_g) coordinate values of two different nodes j and g, respectively, and d(j, g) represents the Euclidean distance of the node (j, g).

The improved distance heuristic function factor is introduced into the node state and substituted into the node state transition function.

$$p_{ij}^{k}(t)=\begin{cases}\dfrac{\left[\tau_{ij}(t)^{\alpha}\cdot\left[1/d(j,g)^{\beta}\right]\right]}{\sum_{s\in i}\tau_{is}(t)\cdot\left[1/d(j,g)\right]^{\beta}}\\ 0,\ \text{otherwise}\end{cases} \tag{37}$$

Through the factor analysis of the modified distance heuristic function, it can be seen that the expected value between different nodes on the search path and the target node will increase with the decrease of the distance between them. If the selected node j around node i is closer to the target node g, the expected value between the two nodes will be larger. Combining with the relevant calculation formula, the improved heuristic function factor is introduced into the corresponding calculation formula, which can effectively promote the ant to select the node j closer to the target node, and effectively enhance the global search ability of the whole algorithm as well the convergence rate (Gao et al., 2017).

In the process of path searching, ants will determine the specific transfer direction by the concentration of pheromones on the path, which will further enhance the global search ability of the whole algorithm and improve the convergence performance of the algorithm. The detailed calculation steps of the whole algorithm are given below:

1 Pheromone local update

Local pheromone updating can promote ants to select unused grid edges, and can effectively avoid ants convergence to the same path. In the process of path searching, ants need to update the pheromone on the path by using the following formula for each step:

$$\tau_{ij}(t+1)=(1-\rho)\tau_{ij}(t)+\varepsilon\tau_0 \tag{38}$$

In the formula above, τ_0 represents the pheromone under the initial condition; ρ represents the volatilisation coefficient of the pheromone, whose range of values is within [0, 1], and ε is an arbitrary number.

2 Pheromone global update

The main purpose of pheromone global update is to ensure that ants are more instructive in path search, so as to enhance the global search ability of the whole algorithm and improve the convergence performance of the whole algorithm. When all ants complete the iteration path search, they need to select the shortest one of all paths and bring it into the corresponding formula to optimise the model and obtain the optimal solution:

$$\tau_{ij}(t,t+1)=(1-\rho)\tau_{ij}(t)+\rho\Delta\tau_{ij}(t,t+1) \tag{39}$$

$$\tau_{ij}(t,t+1)=\begin{cases}K/L_{im},\ \text{if } (i,j)\\ 0,\qquad \text{otherwise}\end{cases} \tag{40}$$

In the formula above, $\Delta\tau_{ij}$ represents the pheromone increment of ants on paths (i, j) in the global update process, and L_{im} represents the shortest path length of iteration at this stage.

The detailed calculation steps of the whole algorithm are given below.

1 On the platform of MATLAB, the corresponding model is constructed by using the grid method.

2 Initialise the parameters, set the starting point of ants to S, the end point to E, and the number of ants to m.

3 The ant algorithm searches iteratively and places all ants at the starting point.

4 The state transition probability of ants from one node to another is calculated according to the corresponding formulas, and the next node is determined by the roulette method.

5 After determining the location of the next node, the pheromone of the ant's path is updated by using the corresponding formula.

6 Judge whether all ants reach the end point, if so, turn to the next step; otherwise, return to Step 4.

7 Statistical ant can search all the optimal paths, select the shortest one, and update all pheromones on the whole path through the corresponding functional formula.

8 Judge the number of iterations NC ≤ NC – max, if it is, then jump to Step 3; conversely, the whole calculation ends and the final calculation results are output.

3 Simulation experiment

The experiment selects the actual environment of a parking lot in a city as an example. The intelligent parking ecosystem proposed in Lin et al. (2017a) was compared with the method proposed in this paper. On the basis of selecting the optimal parking space by the method in this paper, the application time of the method is taken as the experimental index for comparative test analysis. By comparing the path iteration effect of the two methods, the application performance of the different methods was evaluated with the experimental index. As shown in Figure 4, there are only five effective parking spaces in this parking lot at a certain time. The results of selecting the optimal parking spaces using the proposed method are given in Table 1.

The following compares the running time of different intelligent city parking guidance methods with the traditional methods. The specific results are shown in Figure 5.

Figure 4 Distribution of free and effective parking space in smart city parking (see online version for colours)

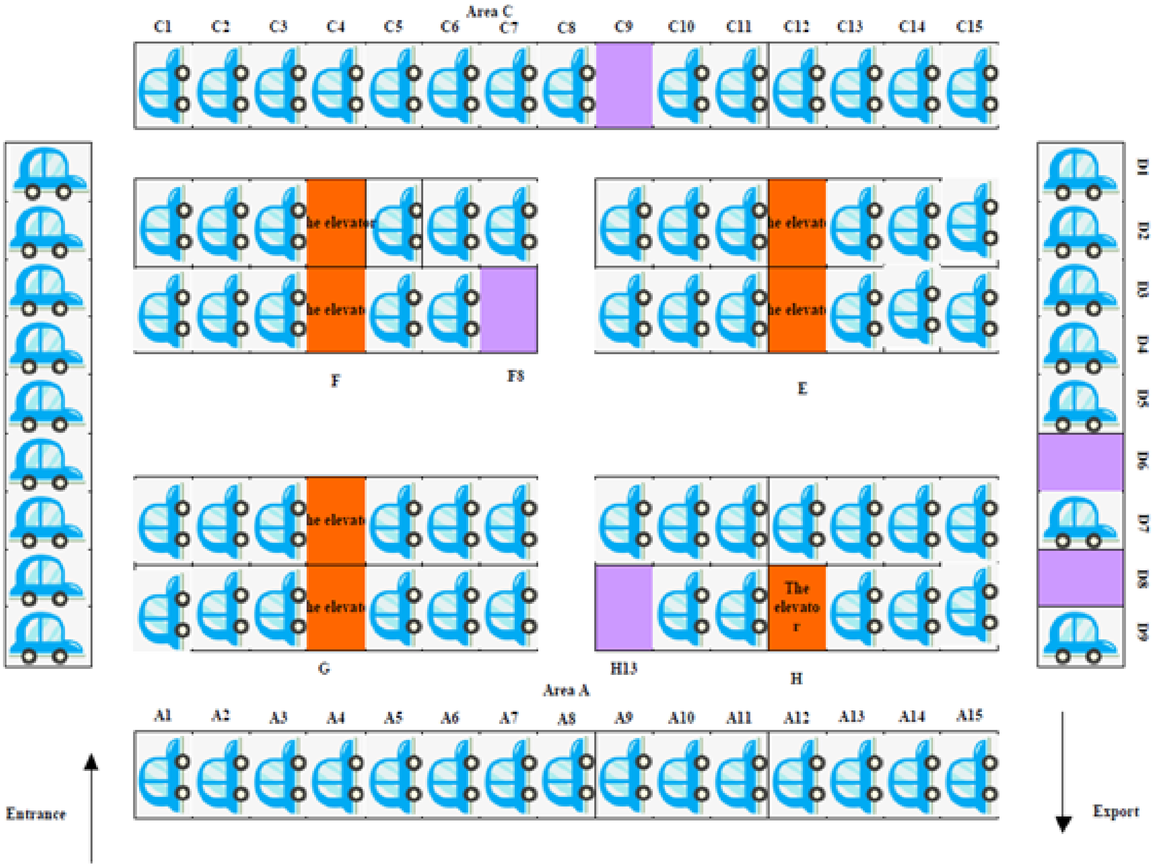


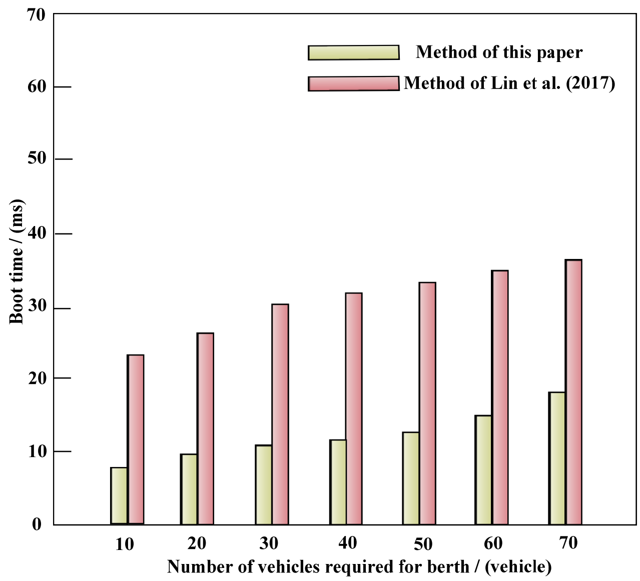
Table 1 Specific attributes of valid parking spaces in parking lots

Effective parking space	*Walking distance/(m)*	*Distance/(m)*	*Occupancy of vehicles on both sides of parking space*
C9	4	41	Vehicles parked on both sides
D6	18	44	Vehicles parked on both sides
D8	14	40	Vehicles parked on both sides
F8	16	26	Vehicles parked on the left
H13	8	28	Vehicles parked on the right

Through the analysis of Figure 5, we can see that compared with the traditional method, the boot time of the proposed method is relatively short. The main reason is that the proposed method uses ACO to optimise the model and obtain the optimal berth path, which can effectively reduce the boot time and improve the computational efficiency of the whole method.

In order to verify the comprehensive effectiveness of the proposed method, the trajectory iteration of the proposed method is compared with that of the traditional method in the environment of Figure 4. The specific comparison results are shown in Figures 6 and 7.

Figure 5 Comparison results of different methods of vehicle berth guidance time (see online version for colours)



By analysing Figures 6 and 7, we can see that the search speed of the proposed method is relatively faster than that of the traditional method, which fully proves the superiority of the proposed method.

Figure 6 Path trajectory iteration diagram of Lin et al. (2017a) method

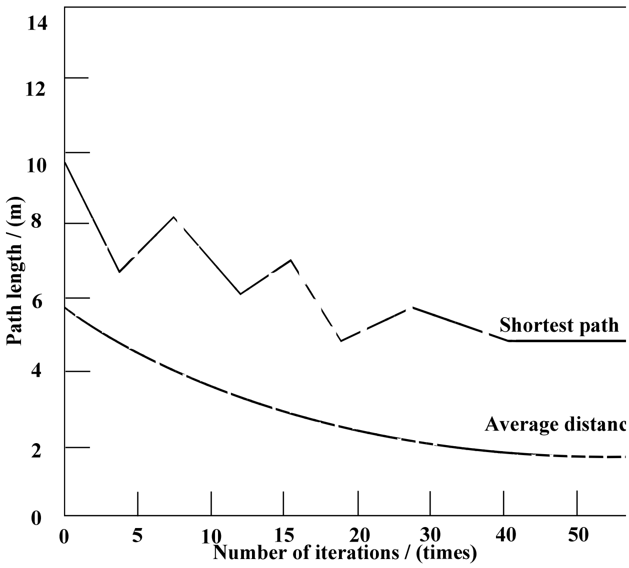
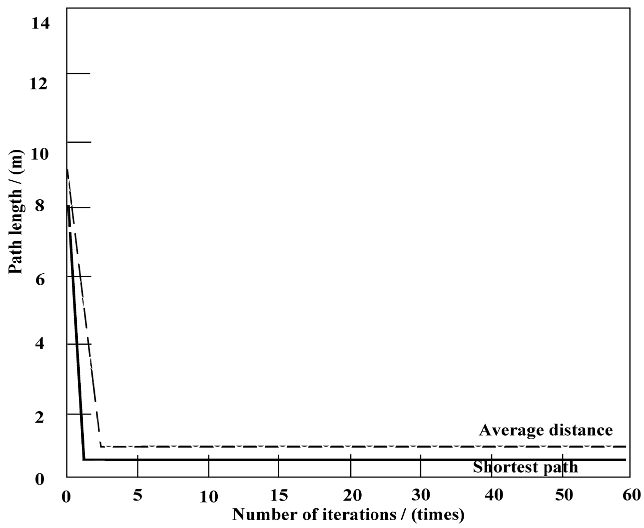


Figure 7 Path trajectory iteration diagram of the proposed method



4 Conclusions

In view of the shortcomings of the traditional methods, this paper improves them, proposes an intelligent city parking guidance method based on ACO, studies the main technologies of the research methods in depth, and obtains relatively research results. However, there are some shortcomings in the research methods at this stage. The following detailed issues need to be further studied in the future are given:

1 The proposed method has carried out in-depth analysis and research for the corresponding research theory, but there will be some shortcomings in the daily application process, which will be improved in the future.

2 In the established model, due to the lack of sufficient experimental data to determine, the future stage will carry out a comprehensive study on this aspect.

3 The current model solving algorithm cannot meet the current development needs. How to solve the model more quickly is the focus of the next research.

Acknowledgements

This work was supported by the Science and Technology Research Program of Chongqing Municipal Education Commission (Grant No. KJ1711278), the Industrial Technology Development Project of Chongqing Development and Reform Commission, China (Grant No. 2018148208), Chongqing Research Program of Basic Research and Frontier Technology, China (Grant No. cstc2015jcyjA40026, cstc2016jcyjA0568).

References

Ardalan, S., Goliaei, S. and Isazadeh, A. (2020) 'Evaluating non-deterministic signal machine relative complexity: a case study on dominating set problem', *International Journal of Innovative Computing and Applications*, Vol. 11, No. 1, pp.1–8.

Gao, S., Lv, J.F., Chen, C.B. et al. (2017) 'Vehicle speed sliding mode control method based on feedback linearization', *Computer Simulation*, Vol. 34, No. 6, pp.135–138.

Ge, Y.F., Chen, T., Kong, X.Y. et al. (2016) 'Application of improved ant colony algorithm in urban car navigation', *Control Engineering*, Vol. 23, No. 1, pp.133–137.

Gong, Y., Zhang, J., Li, S.J. et al. (2016) 'Step-by-step induction algorithm for frequent changes in congestion of random road networks', *Transportation Systems Engineering and Information*, Vol. 16, No. 3, pp.67–72.

Huang, Z.D. and Xiao, R.B. (2018) 'Dual neighbourhood discrete artificial bee colony algorithm for the convex polygons packing problem with constraints of equilibrium', *International Journal of Innovative Computing and Applications*, Vol. 9, No. 1, pp.1–12

Jeong, B.Y. and Park, M.H. (2017) 'Risk assessment of parking lot management based on occupational injuries data', *Human Factors & Ergonomics in Manufacturing & Service Industries*, Vol. 27, No. 5, pp.147–155.

Jia, Q., Zhang, X.C., Yuan, Y.W. et al. (2018) 'On-line trajectory planning method for tractor automatic driving system', *Journal of Agricultural Machinery*, Vol. 49, No. 4, pp.36–44.

Jin, C., Zhang, Y. and Wang, C. (2018) 'Distributed multi-agent ant colony algorithm with time window vehicle routing problem', *Computer Application Research*, Vol. 35, No. 3, pp.666–670.

Li, X., Zhao, Z.F., Liu, Y. et al. (2018) 'A single-lane improved ns model with induced vehicle speed under the roadway', *Road Traffic Science and Technology*, Vol. 35, No. 2, pp.101–108.

Li, Y.Q., Zhang, L.Y., Guo, C.S. et al. (2016) 'Application of improved ant colony algorithm in emergency VRP problem', *Mathematics Practice and Cognition*, Vol. 42, No. 9, pp.91–95.

Lin, T., Rivano, H. and Mouël, F.L. (2017a) 'A survey of smart parking solutions', *IEEE Transactions on Intelligent Transportation Systems*, Vol. 31, No. 99, pp.1–25.

Lin, X.N., Gu, Y. and Shen, H. (2017b) 'Trajectory planning and tracking control of vehicle autonomous rapid change lane', *Journal of Lanzhou University of Technology*, Vol. 43, No. 6, pp.108–112.

Liu, F.S., Kang, Y., Zhang, X.Y. et al. (2018) 'Effect of initial temperature on turbulent combustion characteristics induced by hydrogen-air premixing', *Internal Combustion Engine Engineering*, Vol. 39, No. 1, pp.23–28.

Ni, J., Zhang, K., Yong, Y. et al. (2018) 'Privacy-preserving smart parking navigation supporting efficient driving guidance retrieval', *IEEE Transactions on Vehicular Technology*, Vol. 18, No. 99, p.1.

Piao, F.B. (2016) 'Simulation and system optimization of induced venting jet field for large vehicle cabins', *Ship Engineering*, Vol. 12, Nos. 19–22, p.33

Qi, B.L., Zhao, F. and Sun, M. (2017) 'Parking induction parking system based on Bluetooth communication', *Computer Engineering and Design*, Vol. 38, No. 2, pp.557–561.

Wang, C., Zhu, M. and Wang, M. (2018) 'Evaluation of sector dynamic traffic capacity based on controller's cognitive load and improved ant colony algorithm', *Journal of Computer Applications*, Vol. 58, No. 13, pp.258–263.

Wang, Q.Q., Zhao, Y., Lu, X.G. et al. (2017) 'Research progress of laser induced breakdown spectroscopy and Raman spectroscopy in the detection of dangerous substances', *Spectroscopy and Spectral Analysis*, Vol. 37, No. 8, pp.2430–2434.

Xiang, T. and Pan, D.Z. (2016) 'Clustering algorithm for solving demand-separable vehicle routing problems', *Computer Applications*, Vol. 36, No. 11, pp.3141–3145.

Xiong, G.M., Lu, H., Guo, K.H. et al. (2017) 'Research on the prediction method of tracked vehicle trajectory based on real-time estimation of sliding parameters', *Journal of Ordnance Engineering*, Vol. 38, No. 3, pp.600–607.

Yan, K., Li, A.G. and Guo, J. (2016) 'Single distribution center path optimization method based on hybrid algorithm', *Journal of Surveying and Mapping Science and Technology*, Vol. 33, No. 6, pp.650–653.

Yuan, N., Shi, X. and Zhao, X.M. (2018) 'Vehicle trajectory planning method based on improved artificial fish swarm algorithm', *Computer Application*, Vol. 38, No. 10, pp.284–289+301.

Zhao, S.S., Yu, H. and Li, G.Q. (2018) 'Trajectory planning for steering robots for automotive experiments', *Chinese Journal of Construction Machinery*, Vol. 16, No. 2, pp.153–157.

Zhao, X., Zhao, K. and Hai, F. (2014) 'An algorithm of parking planning for smart parking system', *Proceeding of the 11th World Congress on Intelligent Control and Automation*, Shenyang, China, 29 June–4 July.