Real-time Allocation of Shared Parking Spaces Based on Deep Reinforcement Learning

Minghai Yuan, Chenxi Zhang, Kaiwen Zhou, Fengque Pei*

College of Mechanical and Electrical Engineering, Hohai University, China
ymhai@hhu.edu.cn, 2624176205@qq.com, 1134357020@qq.com, fq_pei@163.com

Abstract

Aiming at the parking space heterogeneity problem in shared parking space management, a multi-objective optimization model for parking space allocation is constructed with the optimization objectives of reducing the average walking distance of users and improving the utilization rate of parking spaces, a real-time allocation method for shared parking spaces based on deep reinforcement learning is proposed, which includes a state space for heterogeneous regions, an action space based on policy selection and a reward function with variable coefficients. To accurately evaluate the model performance, dynamic programming is used to derive the theoretical optimal values. Simulation results show that the improved algorithm not only improves the training success rate, but also increases the Agent performance by at least 12.63% and maintains the advantage for different sizes of parking demand, reducing the user walking distance by 53.58% and improving the parking utilization by 6.67% on average, and keeping the response time less than 0.2 seconds.

Keywords: Shared parking, Deep reinforcement learning, Space allocation model

1 Introduction

According to the data of the Ministry of Public Security Traffic Administration, by 2021, the number of motor vehicles in China will reach 395 million, but the number of parking spaces is only 131 million. Facing such a huge gap, it is difficult to solve it by increasing the number of parking spaces, and it is more important to improve the utilization of existing parking spaces. Therefore, with the support of technologies such as Internet of Things and big data [1], a shared parking platform [2-4] is established to share the private parking spaces in the community that are idle due to residents going to work and other reasons during free hours for those in demand to alleviate parking demand. However, compared with professional parking lots, communities cover a larger area, parking spaces are sparsely distributed, and each shared parking space has a different sharing period, thus giving rise to the problem of optimizing the resource allocation of shared parking spaces.

1.1 Related Work

In recent years, a large number of studies on parking space management have focused on two aspects: parking space pricing strategy and parking resource allocation strategy.

Around the aspect of parking space pricing, literature [5] designed an auction-based uniform pricing strategy, which effectively improved the platform revenue; for the dynamic pricing problem [6], literature [7] constructed a multi-period noncooperative two-tier model to describe the interactive competition among parking agencies and designed a non-myopic approximate dynamic programming (ADP) approach to solve it; while literature [8] used a distributed framework to construct an adaptive pricing strategy based on virtual voting.

In terms of parking space allocation, many scholars construct shared parking space allocation models with optimization objectives such as space utilization [9], social welfare [10], parking walking distance [11] and platform revenue [12], and solve them using multiple classes of intelligent algorithms, including genetic algorithms [13-14], ant colony algorithms [15] and particle swarm algorithms [16]. In addition, to ensure the robustness of parking space allocation, the literature [17] constructs a many-to-many structured recurrent neural network to achieve accurate prediction of parking space status; while the literature [18] constructs a time-of-day based parking probability function for random factors such as untimely parking.

Deep reinforcement learning, as a branch of machine learning, is widely recognized as a powerful way to solve Markov Decision Process (MDP). In recent years, many scholars have used reinforcement learning to allocate parking resources. In literature [19], the parking waiting model is described as a Markov queue in continuous time; in literature [20], a two-layer online optimization model based on reinforcement learning is constructed to reduce users' parking waiting time; in literature [21], a deep reinforcement learning model is constructed to estimate the parking status with high efficiency and accuracy.

1.2 Contribution

In this paper, with the background of shared parking and with the help of deep reinforcement learning, we research the mechanism of shared parking space allocation based on real-time demand and space heterogeneity. The contribution of

*Corresponding Author: Minghai Yuan; E-mail: ymhai@hhu.edu.cn
DOI: 10.53106/160792642023012401004

this paper can be summarized as follows:

1) A multi-objective shared parking space real-time allocation model is constructed and transformed into a Markov decision process.

2) A two-stage variable coefficient DQN algorithm is proposed for model training, and a dynamic planning-based evaluation strategy is designed.

3) The performance of the proposed algorithm and model are validated, which shows that the model can effectively reduce walking distance of users and improve the utilization of parking spaces.

The rest of this paper is organized as follows. Section 2 builds a multi-objective optimization mathematical model for the parking space allocation problem. Section 3 transforms the mathematical model into a Markov decision process and designs an improved algorithm based on DQN for model training. Section 4, simulation and verification. Section 5, conclusion.

2 Problem Formulation

Compared with professional parking lots, shared parking lots have two significant problems: the shared parking spaces are more sparsely distributed and the spatial heterogeneity problem is more prominent, leading to the walking distance after parking becoming an important factor affecting user satisfaction; each parking space is shared for different time periods, leading to the need for the allocation model to focus on temporal sequencing. The model considered in this paper is based on real-time allocation, which means that parking requests and parking space allocation are made in real time. Based on this, the parking space scheduling platform is able to optimize the parking space allocation scheme in real time according to the different demands of each parking order, and improve the average user satisfaction and the utilization rate of shared parking spaces.

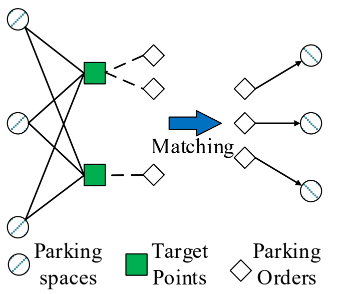

Figure 1. Parking space matching

More specifically, as shown in Figure 1, for a shared parking lot with n spaces and p target points, each space is shared during its respective sharing period $(T_i^{st} \sim T_i^{ed})$, and the space allocation system accepts a parking order every time interval T_{order}. The information of each order includes the demanded parking length l_k and the parking target point h_k. The system matches spaces according to an intelligent policy so that the average walking distance of all users is reduced and the space Utilization rate. When the user's parking length demand cannot be met, the system can match multiple parking spaces for the user in different time slots to meet the user's needs as much as possible. The research problem is based on the following assumptions: parking spaces are free at the start of sharing; parking is prohibited beyond the sharing period; parking orders are generated at a fixed interval; and all users strictly comply with the platform scheduling for parking.

Therefore, this paper takes minimizing the average walking distance of users and maximizing the utilization rate of parking spaces as the optimization objectives, as shown in Equations (1) and (2), where b_k is whether to assign a parking space to an order, d_{ij} is the distance from the parking space i to the target point j, and $l_k^{'}$ is the actual parking duration. Specifically, the constructed multi-objective shared parking real-time allocation model is as follows:

$$f_1 = \min \frac{\sum_{k=1}^{m} b_k d_{u_k h_k}}{\sum_{k=1}^{m} b_k} . \tag{1}$$

$$f_2 = \max \frac{\sum_{k=1}^{m} b_k l_k^{'}}{\sum_{i \in I} (T_i^{ed} - T_i^{st})} . \tag{2}$$

$$s.t. B_k = \{ i \mid i \in I, e_i = 0 \} . \tag{3}$$

$$e_i, b_k \in \{0,1\}, h_k \in J, u_k \in B_k . \tag{4}$$

$$b_k = (B_k \neq \phi) \wedge (l_k + T_{now} \leq \max_{i \in I} T_i^{ed}) . \tag{5}$$

$$l_k' \leq \min\{l_k, T_{u_k}^{ed} - T_{now}\} + L . \tag{6}$$

Equation (3) represents the set of currently assignable parking spaces for order k, where e_i is the state of the parking space i; Equation (4) restricts the values of e_i, b_k, the target point and the parking position; Equation (5) indicates whether the parking space can be assigned to order k, depending on whether the available parking space is non-zero and whether the parking duration exceeds the maximum shared period; Equation (6) indicates that the actual parking duration cannot exceed the reserved duration and the shared duration of the parking space, but the parking overtime is allowed not to exceed the specified duration L.

3 Problem Solution

In this section, the real-time allocation model of shared parking spaces is transformed into a Markov decision process; a DQN-based training method and a dynamic programming-based evaluation method are designed.

3.1 Construction of Markov Decision Process

In a dynamic shared parking environment, the Agent makes reasonable scheduling decisions based on real-time parking space information and order information, and since the future state depends only on the current state and the current policy, the original problem can be described as a Markov decision process. An MDP can consist of a five-tuple shown in Equation (7), where $P_{ss'}^{a}$ is the state transfer matrix; S is the state space to describe the shared parking state; $A(s)$ is the action space to select parking spaces; R_s^a is the reward function to evaluate the Agent's behavior; λ is the discount factor to measure the importance between long-term reward and immediate reward.

$$< S, A(s), P_{ss'}^{a}, R_s^a, \lambda \mid s, s' \in S, a \in A(s) > \quad . \tag{7}$$

To reduce the dimensionality of the state space and action space and make the algorithm easy to converge, the parking lot is divided into multiple heterogeneous regions based on the location information of the shared parking spaces, and the space selection is transformed into region selection. The literature [22] shows that most drivers can tolerate walking distances of less than 200 m after stopping. Therefore, within 200 m is defined as the dominant region, and for each target point, it is divided into three types of regions (A, B, and C) according to the distance from the target point [0, 50], [50, 100], and [100, 200] m. Other parking spaces are classified as D regions. As shown in Figure 2, the zoning status of a shared parking lot with the number of target points of 3 is divided into 10 zones.

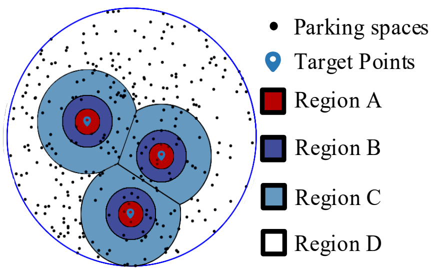

Figure 2. Area division

3.1.1 State Space

The observations in the state space include the following three items: the number of remaining spaces in each region, which serves as the basis for the Agent to evaluate the current space information; the information of the current order, including the target point h_k and the demanded parking time l_k, which serves as the basis for the Agent to filter the executable actions and to measure the reward; and the current time T_{now}. The specific description is shown in Equation (8), where c_x is the number of free parking spaces in region x.

$$S = \prod_{i \in X} c_i \times (h_k \times l_k) \times T_{now} \, . \tag{8}$$

3.1.2 Action Space

In order to minimize the size of the action space, the action is divided into two steps: selecting a region and selecting a strategy. The action space is as follows, where X denotes the selection of a region and U denotes the parking space selection strategy.

$$A = X \times U \quad . \tag{9}$$

U contains the following three strategies, where G_x is the set of available parking spaces in region x.

a) The shortest walking distance:

$$u_k = arg \min_{i \in G_x} d_{ih_k} \quad . \tag{10}$$

b) The least remaining sharing time:

$$u_k = arg \min_{i \in G_x} T_i \quad . \tag{11}$$

c) Heterogeneous prominence strategy(most deviated spaces from other target points):

$$u_k = arg \min_{i \in G_x} \left(d_{ih_k} - p^{-1} \cdot \sum_{j \in J} d_{ij} \right) \quad . \tag{12}$$

In addition, to ensure the legitimacy of the action space, a mapping from A to A_s needs to be established by filtering the action space according to each state S. The constraints of G_x are shown in Equation (13), and the legal state space A is shown in Equation (14).

$$G_x = M_x \cap B_k \quad . \tag{13}$$

$$A_s = A \cap \{(x, \delta) \mid c_x \neq 0, \delta \in U\} \quad . \tag{14}$$

3.1.3 Reward

The design of the payoff function takes the objective function as the main reference basis. For the multi-objective optimization problem studied in this paper, two payoff functions are designed for comparison.

a) Linear weighting strategy. By assigning corresponding weights to different objectives, it is transformed into a single objective $f = \omega_1 f_1 + \omega_2 f_2$, The corresponding payoff function:

$$r = -\omega_1 d_{u_k h_k} + \omega_2 l_k \quad . \tag{15}$$

b) Two-stage variable coefficient strategy. Considering the problem that the weighting strategy is not easy to determine reasonable weights and leads to unstable training, variable coefficients and stages are used to make the training stable and efficient. In the first stage, no variable coefficients are introduced so that the Agent is initially trained with f_1 as the main target; in the second stage, variable coefficients are introduced and the target f_2 is introduced into the training.

$$r = (\varphi - 1) \cdot d_{u_k h_k} = (\beta \frac{T_{now} - T_i^{st}}{T_i^{ed} - T_i^{st}} - 1) \cdot d_{u_k h_k} \quad \textbf{(16)}$$

The variable coefficient φ introduced is positively correlated with time and is used to guide the Agent's understanding of the current time T_{now} as an observation in the environment, thus enabling the pursuit of parking space utilization, where β is taken to be between 0.05 and 0.1.

3.2 Model Training Framework

The constructed DQN training framework is shown in Figure 3. Each time, order information, current time, and the number of free parking spaces in each area are input as the current state; the Agent selects the optimal action based on the current state, and gets the next state and reward by interacting with the environment; the sample (s_t, a_t, r_t, s_{t+1}) is sent to the experience pool [23] for disruption and storage; and then the samples are periodically taken out for model training. The three-layer feedforward network used in this paper describes the Q-value network of the Agent, as shown in Figure 4.

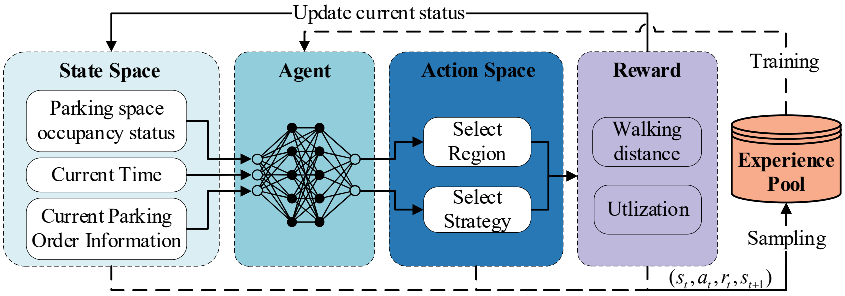

Figure 3. Deep reinforcement learning training framework

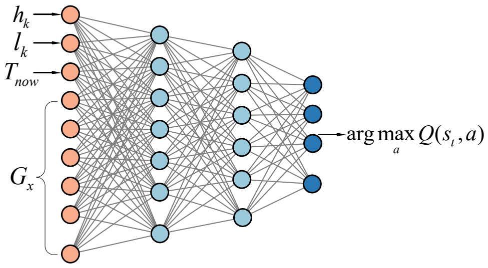

Figure 4. Q-value network

Algorithm 1. Improved DQN

Input: batch size b, discount factor λ, learning rate α, exploration factor ϵ
Output: network parameters θ^*

Initialize action-value function Q with random weights θ
Initialize target action-value function $\hat{Q}$ with weights $\theta^- = \theta$
for iteration = 1 to N **do**
  Initialize state $c_x = |M_x|$, $T_{now} = \min T_i^{st}$
  while $T_{now} \leq max\ T_i$ **do**
    Input s_t, output Q value of each action, filter illegal actions ▷ Eq.: 14
    With probability ϵ select a random action, otherwise select $a_t = \arg\max_a Q^*(s_t, a)$
    Execute a_t, get the new state s_{t+1} and reward r_t
    Put (s_t, a_t, r_t, s_{t+1}) into the experience pool
    if step % learn frequency = 0 **then**
      Get b samples from the experience pool as a mini batch
      Calculate the target Q value ▷ Eq.: 17
      Calculate the training loss with respect to Q_{target} ▷ Eq.: 18
      Perform a gradient descent to update network parameters ▷ Eq.: 19
      Modify exploration factor $\epsilon = max\{\epsilon_{min}, \epsilon - \Delta_\epsilon\}$
    end if
    if learn episode > Number of training sessions in stage 1 **then**
      Convert r in the experience pool, replace the reward function ▷ Eq.: 16
    end if
    Every C learning steps set $\hat{\theta} \leftarrow \theta$
  end while
end for

The designed state space, action space, and reward function are integrated into the DQN algorithm to obtain the specific learning steps, and the pseudo-code is shown in Algorithm 1.

Where STEP11-14 is the key step of model training, calculating the target Q-value as in Equation (17); calculating the mean squared loss with the predicted Q-value as in Equation (18); using the mini-batch gradient descent method for direction propagation, for a batch of samples of number b, the update formula of the Q-value network parameters is obtained after deriving $L_i(\theta)$ as in Equation (19), where α is the learning rate.

$$Q_{target} = r_t + \gamma \max_a Q(s_{t+1}, a \mid \hat{\theta}) \ . \tag{17}$$

$$L_i(\theta) = \frac{1}{2} E_{(s_t,a_t,r_t,s_{t+1})\sim U(D)} [Q_{target} - Q(s_t, a_t \mid \theta)]^2 \ . \tag{18}$$

$$\theta_{k+1} = \theta_k - \alpha \frac{1}{b} \sum_{i=1}^{b} [Q_{target} - Q(s_t, a_t \mid \theta)] \nabla_\theta Q(s_t, a_t \mid \theta) \ . \tag{19}$$

The innovations involved in the improved algorithm include the following:

Step 6, in order to ensure the legitimacy of Agent actions, a mask layer is added after the output of the value function to filter illegal actions according to the mapping relationship of Equation (13) and (14).

Step 15, a variable $\epsilon - greedy$ strategy is adopted: ϵ is large in the early stage of training to encourage the Agent to explore; ϵ decays gradually in the later stage of training to accelerate convergence

Step 18, using a two-stage variable coefficient reward function: gradually introducing multiple optimization targets for training through the variable coefficient payoff function shown in Equation (16); and using phased training to ensure stability.

3.3 Evaluation Strategies

To evaluate the effectiveness of the improved DQN algorithm, a theoretical optimum is used for comparison. The theoretical optimal solution is derived using dynamic programming iterations with known information about all orders in a day, including the start time and departure time of each user from the parking space. This means that the state transfer matrix $P^a_{ss'}$ is completely deterministic, so Q^* can be reached by greedy selection for iteration. The iterative formula is given in Eq. (20), where Q_k is the Q-value function for the k-th iteration

$$Q_{k+1} = R^a_s + \gamma \sum_{s' \in S} P^a_{ss'} \max_{a'} Q_k \ . \tag{20}$$

4 Simulation Analysis

4.1 Parameter Setting

Taking a university as an example (the spatial distribution is shown in Figure 5), three target points are established; parking spaces of different densities (300 by default) and order generation cycles of different lengths (72s by default) are generated according to the simulation requirements; the sharing start and end times of each parking space obey a uniform distribution from 7:00 to 9:00 and from 15:00 to 17:00, respectively; the demanded parking duration of each order obeys a normal distribution; the maximum parking timeout is 0.5 h.

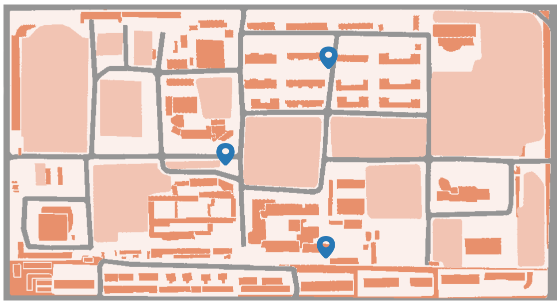

Figure 5. Spatial distribution map of a university

The training hyperparameters of the improved DQN algorithm are shown in Table 1. The discount factor is calculated using the empirical formula $\lambda = 0.1^{1/t}$, and the number of steps in a single round in this problem is about 300, resulting in $\lambda \approx 0.99$; the learning rate is linearly decayed, $\alpha = \alpha \cdot \gamma$, the initial value is set to 10^{-3}, and every learning 1000 times reduced to 0.9; exploration factor $\epsilon = max\{\epsilon_{min}, \epsilon - \Delta\epsilon\}$, initial value is 0.1, reduced $\Delta\epsilon = 10^{-6}$ after each sampling. The fully connected network is set to have an input layer of size 13, two implicit layers of sizes 64 and 32, and an output layer of size 20.

Table 1. Simulation hyperparameters

Hyperparameters	Value
Maximum number of training sessions (N)	2000
Update period of the target network (C)	50
Learning frequency	20
Discount factor (λ)	0.99
Initial value of learning rate (α)	10^{-3}
Number of training sessions in stage 1	500
Experience pool size (E)	3000
Batch size (b)	32
Initial value of ϵ	0.1

4.2 Algorithm Comparison

The algorithm comparison includes three DQN-based car space matching algorithms and a theoretical optimal algorithm for evaluating intelligent decisions: the improved DQN algorithm, which uses segmented training with the training process shown in Algorithm 1; Single-objective algorithms, trained only for the objective of walking distance; the linear weighting algorithm, which uses a linear weighting method to transform the multi-objective problem into a single-objective problem and uses the payoff function shown in Equation (15) for model training; and the theoretical optimal algorithm, which uses dynamic programming to calculate the optimal value, based on Equation (20). Figure 6 and Figure 7 show the test results for the average distance traveled and space utilization, and Figure 8 shows the training process.

It can be seen from Figure 6: After training using the single-objective algorithm, the results of the Agent car space assignment are highly fluctuating; the Agent trained with the linear weighting algorithm has a more stable performance, and the average walking distance is controlled between 151 The average distance traveled is between 151 m and 184 m, with a mean value of 166.37 m. The average distance traveled by the Agent trained with the improved DQN algorithm is between 138 m and 153 m, with a mean value of 145.35 m. Compared with the first two groups, the performance is improved by at least 12.63%, which is the closest to the theoretical optimal value and the training results are relatively stable. As seen in Figure 7, the improved DQN algorithm is higher than the other algorithms in terms of space utilization, and the difference with the theoretical optimal value is less than 2.5%.

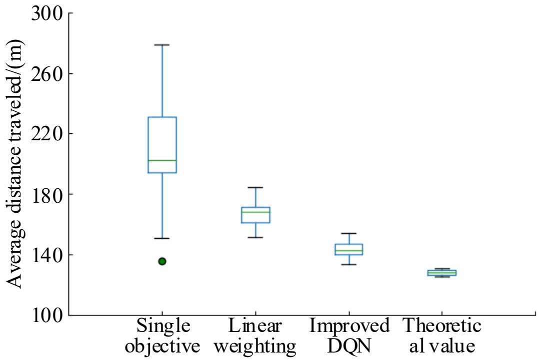

Figure 6. Comparison of walking distance

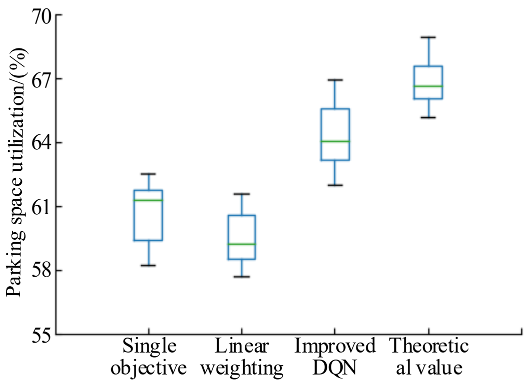

Figure 7. Comparison of parking space utilization

It can be seen from Figure 8: When the linear weighting algorithm was used for training, it took more than 1000 rounds of training to converge, which is significantly slower than the other two groups, and the final performance is not as good as the other groups; the single-objective algorithm is used to perform well in this training, but comparing with Figure 6, it can be seen that the final performance gap of each training is larger due to the small exploration range in the pre-training period, which relies too much on the selection of initial values. The analysis can be obtained that the improved DQN algorithm with phased processing makes the Agent have stronger search ability in the early stage and can learn the allocation strategy quickly, and focus on the utilization rate of the parking space and converge quickly after the introduction of variable parameters in the later stage, so that the Agent learns incrementally throughout the training process and the convergence speed and model performance are stable.

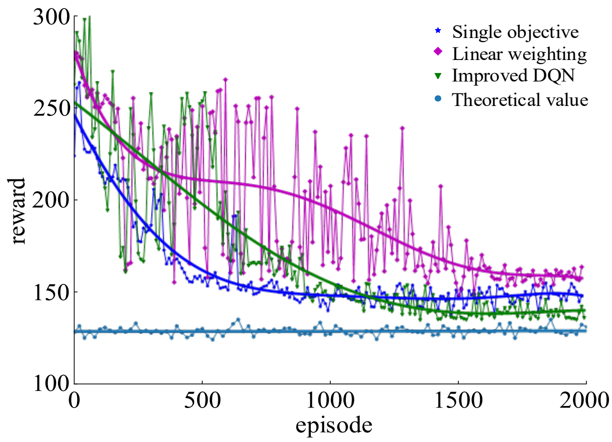

Figure 8. Comparison of training process

4.3 Model Performance Comparison

This section focuses on verifying the superiority and effectiveness of the intelligent parking space allocation model compared with greedy allocation strategy of the «first-come, first-served» and the theoretical optimal value for different number of parking spaces and different order generation cycles. The greedy allocation strategy scheme assigns the closest parking space to the target point each time.

Firstly, we test the influence of the number of parking spaces on the allocation of parking spaces by considering six cases with 100, 200, 300, 400, 500 and 600 parking spaces, and test each case 20 times. Figure 9(a) and Figure 9(b) show the average walking distance and space utilization rate under different number of parking spaces, and it can be seen that as the number of shared parking spaces increases, the average walking distance after using real-time intelligent allocation decreases significantly, and the average difference with the theoretical optimal value is less than 6.25%. Compared with the greedy allocation strategy, the average walking distance is reduced by 53.58% and the average parking space utilization rate is increased by 6.67%; and the smaller the number of parking spaces, the more significant the advantage of intelligent allocation in terms of parking space utilization. Figure 9(c) shows the percentage of users walking distance less than 200 m. It can be seen that when the number of parking spaces reaches 400, 95.80% of users' walking distance is controlled within 200 m, which basically meets the tolerance distance of most users.

In addition, the effect of the order generation cycle on the parking space allocation was tested for 20 groups at 2 s intervals in the interval of 20 to 140 s. The results are shown in Figure 10. It can be seen that the intelligent allocation is stable in all cases, with an average deviation of less than 7.81%, and the gap with the theoretical optimal value is not

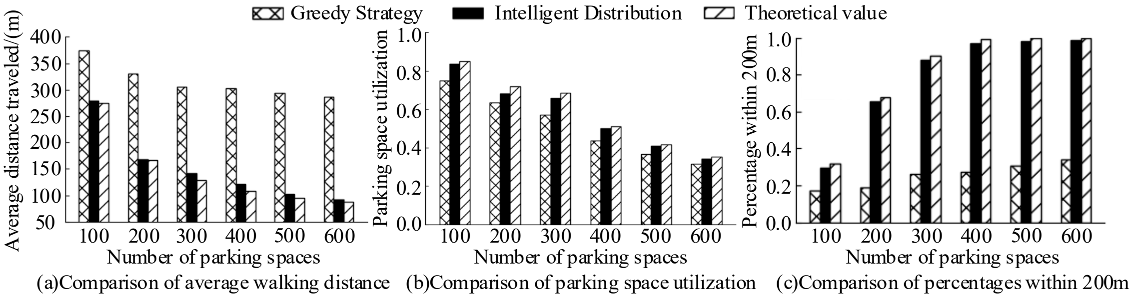

Figure 9. Comparison under different number of parking spaces

higher than 8.66%; with the growth of the order generation period, the gap with the theoretical value slightly widens, but the advantage over the greedy strategy is still obvious.

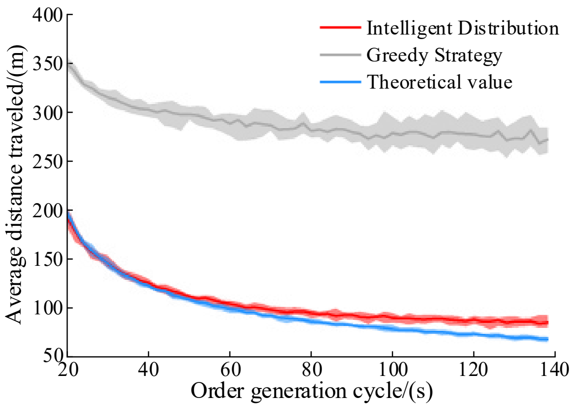

Figure 10. Comparison under different order generation cycles

Finally, the system response time is tested. The response time is defined as the time interval between an order being accepted and being assigned a parking space, which mainly includes the time for the two parts of environment status update and parking space selection. The parameters that determine the size of the research problem include the number of parking spaces, the number of target points, and the order generation cycle, but the order generation cycle does not affect the single-step response time. Therefore, this paper tests the performance of the model response time under different number of parking spaces and number of target points. As shown in Figure 11, the response time increases with the number of parking spaces, and only when the number of parking spaces reaches the extreme case of 3200, the response time starts to increase significantly, but it is still lower than 0.2 seconds, which can meet the normal parking demand. As shown in Figure 12, the number of target points has no significant effect on the response time, and the model can maintain a fast response. From the results, it can be seen that although the response speed of the intelligent allocation model proposed in this paper is slightly slower than that of the greedy strategy, it can still meet the demand for parking space allocation of different sizes.

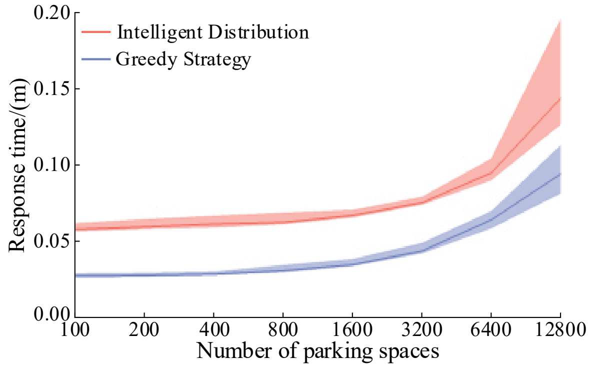

Figure 11. Response time under different number of parking spaces

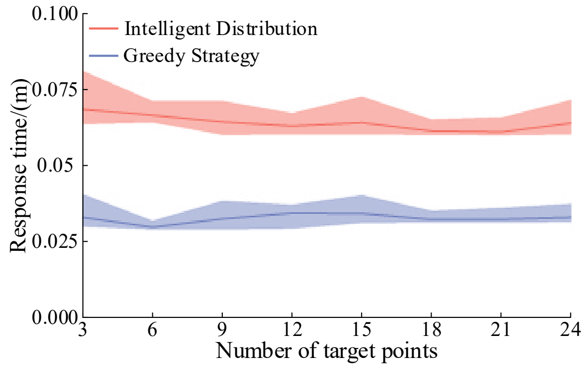

Figure 12. Response time under different number of target points

5 Conclusion

To solve this multi-objective optimization problem, a deep reinforcement learning-based solution is proposed, which gives a state space for heterogeneous regions, an action space based on policy selection and a reward function with variable coefficients. In order to accurately evaluate the model performance, the theoretical optimal value is calculated using dynamic programming.

Simulation results show that compared with the greedy strategy, the real-time allocation model of shared parking spaces proposed in this paper can reduce the user walking distance by 53.58%, improve the utilization rate of parking spaces by 6.67% and stabilize the deviation within 7.81% under different number of parking spaces and different order acceptance frequency. In addition, the response time of the model is less than 0.2s. This shows that the real-time allocation model proposed in this paper can cope with different sizes of parking scenes, effectively reduce users' walking distance and improve parking space utilization, providing an effective solution to the real-time allocation problem of heterogeneous shared parking spaces.

Acknowledgement

This work was supported by General program of Natural Science Foundation of Jiangsu Province under Grant number BK20201162, Ministry of Education Humanities and Social Sciences Planning Fund Project under Grant number 21YJA630111, the Changzhou Science and Technology Support Plan (Social Development) under Grant number CE20205045.

References

[1] X. Zhao, H. Askari, J. Chen, Nanogenerators for smart cities in the era of 5G and Internet of Things, *Joule*, Vol. 5, No. 6, pp. 1391-1431, June, 2021.

[2] X. T. R. Kong, S. X. Xu, M. Cheng, G. Q. Huang, IoT-Enabled Parking Space Sharing and Allocation Mechanisms, *IEEE Transactions on Automation Science and Engineering*, Vol. 15, No. 4, pp. 1654-1664, October, 2018.

[3] S. Jian, W. Liu, X. Wang, H. Yang, S. T. Waller, On integrating carsharing and parking sharing services, *Transportation Research Part B-Methodological*, Vol. 142, pp. 19-44, December, 2020.

[4] B. Q. Tan, S. X. Xu, R. Zhong, M. Cheng, K. Kang, Sequential auction based parking space sharing and pricing mechanism in the era of sharing economy, *Industrial Management & Data Systems*, Vol. 119, No. 8, pp. 1734-1747, September, 2019.

[5] H. Xiao, M. Xu, H. Yang, Pricing strategies for shared parking management with double auction approach: Differential price vs. uniform price, *Transportation Research Part E-Logistics and Transportation Review*, Vol. 136, Article No. 101899, April, 2020.

[6] J. Hao, J. Chen, Q. Chen, Floating Charge Method Based on Shared Parking, *Sustainability*, Vol. 11, No. 1, Article No. 72, January, 2019.

[7] C. Lei, Y. F. Ouyang, Dynamic pricing and reservation for intelligent urban parking management, *Transportation Research Part C-Emerging Technologies*, Vol. 77, pp. 226-244, April, 2017.

[8] V. Hassija, V. Saxena, V. Chamola, F. R. Yu, A Parking Slot Allocation Framework Based on Virtual Voting and Adaptive Pricing Algorithm, *IEEE Transactions on Vehicular Technology*, Vol. 69, No. 6, pp. 5945-5957, June, 2020.

[9] X. Huang, X. Long, J. Wang, L. He, Research on parking sharing strategies considering user overtime parking, *Plos One*, Vol. 16, No. 5, Article No. e0251807, May, 2021.

[10] D. An, Q. Yang, D. Li, W. Yu, W. Zhao, C.-B. Yan, Where Am I Parking: Incentive Online Parking-Space Sharing Mechanism With Privacy Protection, *IEEE Transactions on Automation Science and Engineering*, Vol. 19, No. 1, pp. 143-162, January, 2022.

[11] M. Duan, D. Wu, H. Liu, Bi-level programming model for resource-shared parking lots allocation, *Transportation Letters-the International Journal of Transportation Research*, Vol. 12, No. 7, pp. 501-511, August, 2020.

[12] Y. Ji, J. Dong, Z. Lai, Q. Feng, Optimal allocation of shared parking spaces for hospital parkers considering parking choice behavior under bounded rationality, *Transportation Letters-the International Journal of Transportation Research*, In Press, March, 2022. https://doi.org/10.1080/19427867.2022.2048226

[13] Y. Cai, J. Chen, C. Zhang, B. Wang, A Parking Space Allocation Method to Make a Shared Parking Strategy for Appertaining Parking Lots of Public Buildings, *Sustainability*, Vol. 11, No. 1, Article No. 120, January, 2019.

[14] P. Wu, F. Chu, N. Saidani, H. Chen, W. Zhou, IoT-based location and quality decision-making in emerging shared parking facilities with competition, *Decision Support Systems*, Vol. 134, Article No. 113301, July, 2020.

[15] E. J. Yao, Z. C. Zhang, J. L. Zhang, F. Xue, Y. K. Luo, A Model and Algorithm for Optimization of the Utilization of Residential Shared Parking Slots, *Journal of Transportation Systems Engineering & Information Technology*, Vol. 17, No. 2, pp. 160-167, 2017.

[16] W. H. Zhang, Y. M. Su, J. Dai, L. Z. Wang, Distributing Model For Shared Parking in the Residential Zones, *Journal of Transportation Systems Engineering & Information Technology*, Vol. 19, No. 1, pp. 89-96, 2019.

[17] S. Y. Chou, A. Dewabharata, F. E. Zulvia, Dynamic Space Allocation Based on Internal Demand for Optimizing Release of Shared Parking, *Sensors*, Vol. 22, No. 1, Article No. 235, January, 2022.

[18] B. Jiang, Z. Fan, Optimal allocation of shared parking slots considering parking unpunctuality under a platform-based management approach, *Transportation Research Part E-Logistics and Transportation Review*, Vol. 142, Article No. 102062, October, 2020.

[19] J. Xiao, Y. Lou, J. Frisby, How likely am I to find parking? - A practical model-based framework for predicting parking availability, *Transportation Research Part B-Methodological*, Vol. 112, pp. 19-39, June, 2018.

[20] Y. Wang, M. Li, X. Lin, F. He, Online operations strategies for automated multistory parking facilities, *Transportation Research Part E-Logistics and Transportation Review*, Vol. 145, Article No. 102135, January, 2021.

[21] X. Huang, P. Li, R. Yu, Y. Wu, K. Xie, S. Xie, FedParking: A Federated Learning Based Parking Space Estimation With Parked Vehicle Assisted Edge Computing, *IEEE Transactions on Vehicular Technology*, Vol. 70, No. 9, pp. 9355-9368, September, 2021.

[22] P. van der Waerden, H. Timmermans, M. de Bruin-Verhoeven, Car drivers' characteristics and the

maximum walking distance between parking facility and final destination, *Journal of Transport and Land Use*, Vol. 10, No. 1, pp. 1-11, January, 2017.
[23] T. de Bruin, J. Kober, K. Tuyls, R. Babuska, Experience Selection in Deep Reinforcement Learning for Control, *Journal of Machine Learning Research*, Vol. 19, pp. 9, pp. 1-56, 2018.

Biographies

Minghai Yuan received the Ph.D. degree in Mechanical Manufacturing and Automation from Nanjing University of Science and Technology in 2008. He is currently an associate professor with the School of Mechanical and Electrical Engineering, Hohai University, China. His main research interests include scheduling optimization and intelligent manufacturing system.

Chenxi Zhang received the B.S. degree in Energy and Power Engineering from Hohai University in 2020. He is currently a master of Hohai University, China. His research interests include machine learning, smart manufacturing and scheduling optimization.

Kaiwen Zhou received the B.S. degree in Mechanical Engineering from Hohai University in 2020. He is currently a master of Hohai University, China. His research interests include machine learning, cloud manufacturing and big data analysis.

Fengque Pei received the Ph.D. degree in Mechanical Engineering from Nanjing University of Science and Technology in 2019. He is currently a lecturer with the School of Mechanical and Electrical Engineering, Hohai University, China. His research interests include digital twin, machine learning and smart manufacturing.