

Research Article

Improved Optimal Path Finding Algorithm of Multistorey Car Park Based on MCP Protocol

Zhendong Liu ¹, Dongyan Li ¹, Xi Chen,¹ Xinrong Lv,¹ Yurong Yang,¹ Ke Bai,¹ Mengying Qin,¹ Zhiqiang He,¹ Xiaofeng Li,¹ and Qionghai Dai²

¹School of Computer Science and Technology, Shandong Jianzhu University, Jinan 250101, China

²Department of Automation, Tsinghua University, Beijing 100084, China

Correspondence should be addressed to Zhendong Liu; liuzd2000@126.com

Received 9 January 2022; Accepted 21 February 2022; Published 9 March 2022

Academic Editor: Xingsi Xue

Copyright © 2022 Zhendong Liu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

To address the urgent issue of car owners wasting a lot of time because they cannot find empty parking spaces in the largely multistorey car park, the paper conceives a solution including algorithm ParkIG. This scheme uses the infrared photoelectric switch to monitor the parking space status. At the same time, we define a special communication protocol named MCP to analyze the parking space data when analyzing the data transmitted by the photoelectric switch. Algorithm ParkIG is used in finding the nearest empty parking space and optimal path planning. Based on the MCP protocol and navigation system, the algorithm ParkIG initially identifies the layer of multistorey where the car is positioned. Our algorithm ParkIG greatly reduces the time and space size of the search domain and update domain when searching for vacant parking spaces in optimal path planning and ends the algorithm running to avoid a lot of computations that are not essential when the algorithm found an optimal path from the entrance to empty of parking space. Simulation experiments show that the performance of our scheme has been significantly improved, such as the time that it takes for the car owner to search for a vacant parking space is decreased by about 25.8% and reduced the space size of search domain and update domain by 70% compared to the algorithm Dijkstra.

1. Introduction

With the rapid development of the economy, private car travel is becoming increasingly common. The increased number of vehicles not only puts a strain on traffic but also causes a slew of problems for car owners [1]. People's travel is increasingly inextricably linked to automobiles, so they will inevitably encounter parking during travel [2, 3]. In office buildings or shopping malls with heavy traffic, especially during the peak parking period, the problem of parking difficulties is more prominent. The main reasons for the "difficult parking" problem are the following: current urban parking spaces are in short supply, and more importantly, people can obtain less valuable parking spot info in the process of finding parking spaces [4]. This means that there is currently short of a parking space information man-

agement platform to provide reasonable parking guidance to vehicle drivers.

In outdoor car parks, there are already many technologies that can find and navigate empty parking spaces and guide car owners to find appropriate empty parking spaces to meet their parking demands [5, 6]. However, for some indoor car parks, there is no suitable solution due to technical limitations. According to the survey, the current popular map navigation can only reach the entrance of the car park and cannot see the empty parking spaces inside the car park [7]. The previous path planning algorithms can only be carried out in the same layer, which is more difficult for multistorey car parks. Indoor navigation technology and outdoor navigation technology are essentially the same, and both require three types of technical support: indoor positioning technology, indoor map, and route planning technology [8].

Input: reserve the first K free parking spaces as candidate empty parking spaces,
Output: The route distance and route of the nearest empty parking space.
Begin

1. Use IPS technology to locate the location information I of the entrance of the parking lot, locate the vehicle's location information I_{car} in real time, and the initial number of layers of the vehicle $N_{layer} \leftarrow 0$;
2. if the vehicle is on the ground and $I = I_{car}$
3. then $N_{layer} \leftarrow N_{layer} + 1$;
4. elseif the vehicle is underground and $I = I_{car}$
5. then $N_{layer} \leftarrow N_{layer} - 1$;
6. end if
7. Load N_{layer} layer information; $P \leftarrow$ Free parking space;
8. $W \leftarrow$ the Manhattan distance between the entrance, intersection, and the candidate's empty parking space.; // The adjacency matrix
9. $P \leftarrow$ {The first k parking spaces with the smallest Euclidean distance from the entrance in P };
10. $S[1 \sim K] \leftarrow 0$; $H[1 \sim K] \leftarrow \emptyset$; // Path distance S , path H of candidate parking spaces
11. for $i=1$ to K do
12. $S_i, H_i \leftarrow$ ParkD (P_i, W);
13. end for
14. $S \leftarrow \min\{S\}$; $I \leftarrow \text{index}(\min\{S\})$; $H \leftarrow H_I$;
15. Return S, H

End

ALGORITHM 1: ParkIG(K).

Input: Candidate vacant parking space P , an adjacency matrix W formed by the Manhattan distance between the entrance, intersection, and the candidate vacant parking space.
Output: The path distance and path from the entrance to the candidate vacant parking space P .
Begin

1. $V \leftarrow$ entrance, n intersections, and candidate parking spaces; //a total of $n+2$ nodes;
2. initialize mark set $T[1 \sim n+1] \leftarrow 0$, adjacent node set $\text{Adj}(V_i)$, path distance set $D(V)$, used to record the set $L(V)$ of the previous node in the path;
3. while $T(n+1) \neq 1$ do
4. $m \leftarrow \{m | T(m) = 0 \text{ and } \min\{D(m)\}\}$; $T(m) \leftarrow 1$;
5. For each $i \in \text{Adj}(m)$ and $T(i) \neq 1$ do
6. $D(i) \leftarrow \min\{D(i), D(m) + W(i, m)\}$;
7. if $D(m) + W(i, m) < D(i)$, $L(i) \leftarrow m$;
8. end for
9. end while
10. $H \leftarrow$ backtracking according to $L(n+1)$ to find the optimal path from the entrance to the candidate parking space;
11. Return $D(n+1), H$

End

ALGORITHM 2: ParkD (P, W).

Many car owners often circle the car park to park, but they never find a suitable empty parking space. This brings great trouble to the car owners. For car park managers, in the case of heavy traffic, they can only direct the car owners to park in person, and even the managers do not know where there are vacant parking spaces [9]. This situation not only greatly discounts the parking efficiency of car owners but also is not conducive to the work of car park managers.

This paper gives a series of solutions to this type of problem. Each parking place has an infrared photoelectric switch to gather parking data, and the MCP protocol is configured for data transfer. Simultaneously, the algorithm ParkIG is built and employed while planning the path from the entry

to the empty parking spot. In addition, our solution contains a set of car park navigation systems that are convenient for users to park and the management of the car park by the administrator. The car owner may utilize the empty parking spot navigation applet to find an empty parking place and park quickly, increasing the car owner's parking efficiency. The car park manager may also observe and manage the parking places in the car park using the multistorey parking lot management system, which makes the managers' jobs easier.

2. Materials and Methods

2.1. Introduction to Algorithm ParkIG. Algorithm Dijkstra is a classic single-source shortest route algorithm. As a greedy

strategy, it is often used in path planning problems [10]. Compared with other path planning algorithms, such as algorithm Floyd, algorithm ant colony, and algorithm A* (A Star) [11–13], algorithm Dijkstra has certain advantages in both time complexity and space complexity. And for car park path planning, we only need to find the route from the outset to the destination and do not care about other paths [14]. Dijkstra is a single-source shortest path algorithm, so it is more suitable for solving car park path planning problems [15]. In addition, the algorithm Dijkstra is used to determine the shortest path, and other conditions such as the intersection of the car park lanes, parking spots, and their occupancy are configured. The empty parking space resources can be used more efficiently and with higher accuracy [16]. However, the search for empty parking spaces in a car park only requires the path and path distance from the outset to the parking space, while the traditional algorithm Dijkstra will compute the route from the outset to all the other dots, especially when the car park is very large and the number of nodes will be many. It will cause many unnecessary calculations and waste a lot of time [17]. Therefore, it is necessary to improve the algorithm Dijkstra to meet the demand of finding the nearest empty parking space in the car park [18, 19].

The traditional algorithm Dijkstra is used in the process of finding empty parking spaces in the car park. There will be a lot of redundant calculations, especially when the car park is relatively large; the time wasted will be very long, which will not improve the parking efficiency of the car owner [20]. Furthermore, the Dijkstra algorithm is unsuitable for multistorey parking garages. The car owner enters from the entrance of the car park, so the empty parking space closest to the entrance can be prioritized given the Euclidean distance. In an environment like a car park, the number of adjacent intersections at each intersection is generally no more than 4. The search domain can be modified to reduce the time complexity. In addition, determining which floor the vehicle is on is a key part of a multistorey car park.

Based on these ideas, this paper proposes the algorithm ParkIG; the detailed introduction of the algorithm is as follows: first, use IPS technology to locate the vehicle and determine which floor it is on. After the vehicle goes on a certain floor, load the parking lot layout of that floor. The entrance of this floor is taken as the starting point, and K parking spaces with Euclidean distance closest to the European-style entrance (K value can be determined according to the actual size of the car park) are selected as the endpoint. For each of the K empty parking spaces, an undirected weight map is formed together with the starting point and intersection, and the weight is the Manhattan distance among each node. When algorithm Dijkstra is applied to the undirected weighted graph, the concepts of the label set and adjacent node set are introduced to reduce the search domain and update domain from n to less than 4, which decreases the cost of time to a certain extent. In the ParkIG algorithm, for empty parking spots and intersections, the midpoint of parking spots is selected to calculate the distance and other data. The entrance, intersection, and candidate empty parking space are defined as nodes, and the

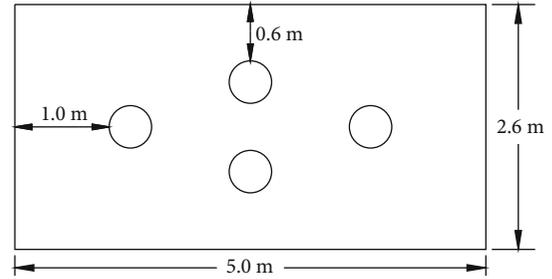


FIGURE 1: Distribution of four infrared photoelectric switches in parking spaces.

collection of adjacent nodes is defined as the collection of adjacent nodes of each node. The tag set is defined to mark whether the nearest route and route distance of the node are determined by the algorithm. If it is “0,” it means it is not determined, and “1” indicates that the path is determined. The path distance set defines the route distance of each node from the outset. The previous node set is used to record the previous node of the currently determined path. The comprehensive algorithm procedures are as follows:

The algorithm ParkIG will call the algorithm ParkD, which adopts the thought of the algorithm Dijkstra and improves it. Algorithm ParkD realizes the optimal path planning from the vehicle to each vacant parking space. As a result, you simply need to use the algorithm ParkD for each candidate’s available parking space to determine the closest distance and optimal path.

The time complexity of the algorithm ParkIG is $O(n^3)$, but due to the particularity of car park routes, the count of adjacent points of nodes is generally not more than 4 in the process of cyclic finding the nearest adjacent points. And because K is a constant, the choice of K is generally not exceeding 10 (depending on the size of the car park). The algorithm ends directly after finding the optimal path from the entrance to the parking space with no vehicles; that is, the maximum count of cycles is n . After adding these constraints, the overall time consumed in the algorithm is no more than $4Kn$, and the overall performance has been greatly improved.

2.2. Car Park Hardware Deployment and MCP Protocol. At present, there are many methods for obtaining the situation of parking spots. For example, the GPP and PGS2 systems use ultrasound, while the Siemens system is a parking sensor placed on the ground [21]. Considering that this system is mainly aimed at large- and medium-sized car parks, and ultrasonic sensors are very sensitive to temperature changes and extreme air, it is one of the wisest methods to monitor vehicles through infrared sensors [22]. Therefore, out of considerations such as funding and accuracy, diffuse reflection infrared photoelectric switches are used at the car park, and four infrared photoelectric switches are arranged in each parking space. Figure 1 depicts the location of the switches in the parking slots. In this figure, the length and width of each parking space are 5.0 meters and 2.6 meters, respectively. The four photoelectric switches are all on the centerline of the parking space. Two photoelectric switches are

TABLE 1: The table of byte descriptions in the MCP protocol.

Type	Province	City	County/district	Community/shopping market	Car park	Data collector number	Parking space data	Check code
Byte count	1	2	3	4-5	6	7	8-15	16

TABLE 2: Parking data table of a single byte in MCP protocol.

The upper 4 bits				The lower 4 bits			
D7	D6	D5	D4	D3	D2	D1	D0
One parking space information				One parking space information			

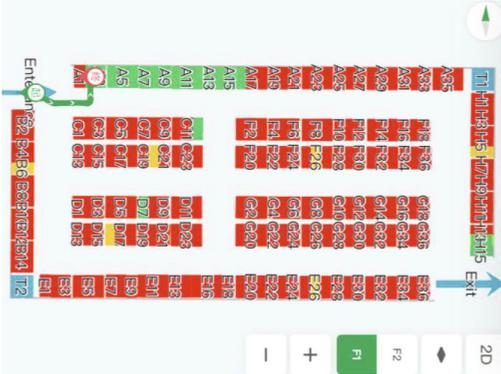


FIGURE 2: Examples of parking recommendations and real-time navigation.

60 cm from the nearest long side, and the remaining two photoelectric switches are 100 cm from the nearest wide side.

The scheme controller uses a single-chip microcomputer to output the parking status information collected by the infrared photoelectric switch. Each photoelectric switch returns one bit of data, and the overwritten data of the photoelectric switch is “0”; otherwise, it is “1.” The communication module adopts a 4G/5G wireless communication module to ensure the data transmission between the car park and the multistorey car park management system [23]. The MCP protocol is based on the distribution of photoelectric switches in the parking spaces. Because there are four infrared photoelectric switches, the management system can know whether the parking space is an empty parking space through the returned data. If it is not an unoccupied parking spot, the management system can determine whether the vehicle parked in the parking space is parked in a standard manner.

As shown in Table 1, the MCP protocol is a new multistorey car park protocol that specifies the following: a total of 16 bytes of data are transmitted, the first six bytes represent provinces, cities, counties, communities, and car parks, the seventh byte represents the data collector number, and the eighth byte to the 15th byte represents the data of parking spaces collected by the collector. For multistorey car parks, each floor has a specific car park number as the basis for judgment. As shown in Table 2, the high and low four digits of each byte represent a parking space, respectively, in the parking data transmitted; that is, the status of 16 parking spaces can be transmitted in one transmission. Such as

Equation (1), if the four-digit data is “1111,” it means an empty parking space, if it is “0000,” it means parking is regulated, and the rest of the data type means parking is not regulated. The last byte is the cumulative checksum.

$$\text{Status} = \begin{cases} \text{Empty parking space,} & \text{the four-digit data} = 1111, \\ \text{Regulate parking,} & \text{the four-digit data} = 0000, \\ \text{Irregular parking,} & \text{others.} \end{cases} \quad (1)$$

2.3. Multistorey Car Park Management System. The parking space management system for the car park is used by the car park administrator, including the two roles of ordinary car park administrator and senior administrator. Ordinary car park administrators can see the data-receiving page, manual entry page, parking space data report statistics page, rent management page, user management page, and personal information modification page. The data-receiving page is used to receive real-time data from the car park. The manual entry page can manually enter data in some fault situations. Parking space data report statistics page can be statistics of a variety of parking data reports, including a statistical table of vehicle entry and exit, a parking information statistical table of parking spaces, a statistical table of vehicle entry and exit time intervals, a statistical table of empty parking spaces within a certain period of time, a statistical table of irregular parking status within a certain period of time, a statistical table of standardized parking status within a certain period of time, a daily report of vehicle entry and exit records, a weekly report of vehicle entry and exit records, a monthly report of vehicle entry and exit records, and an annual report of vehicle entry and exit records. The rent management page is used to manage the rent of temporary parking and long-term parking. The user management page gets used to managing resident users. The personal information modification page is used for modifying personal information and login password. Senior administrators can see the ordinary car park administrator management page and parking space data report statistics page. The ordinary car park administrator management page is used to manage ordinary car park administrators. The parking space data report statistics page has similar functions to ordinary car park administrators. The difference is that senior administrators can manage all car park data.

2.4. Navigation Mini Program for Empty Parking Spaces in Multistorey Car Park. The user terminal includes a page for finding the nearest available parking space as well as a page for real-time navigation. The car owner can view the real-time situation of the parking space in the car park by scanning the QR code of the WeChat applet at the entrance.

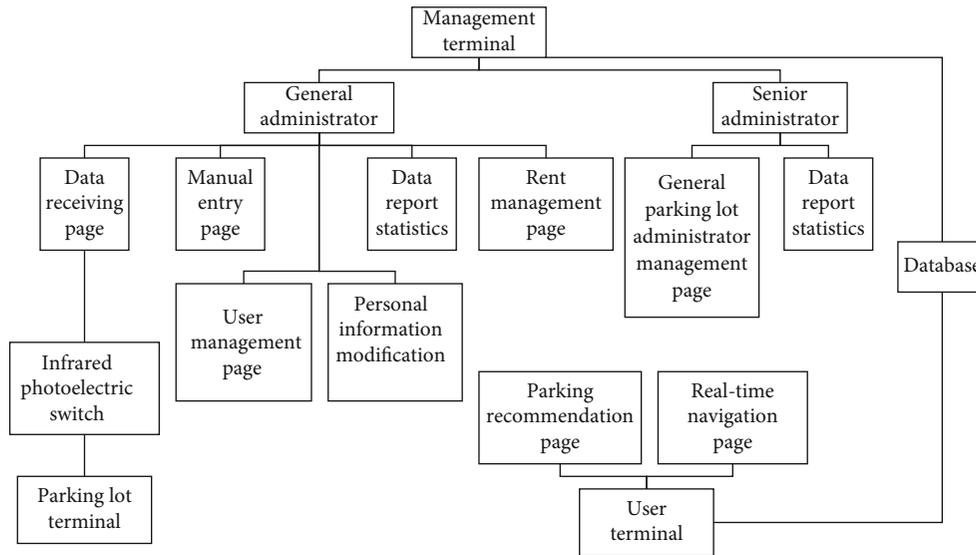


FIGURE 3: The diagram of system architecture.

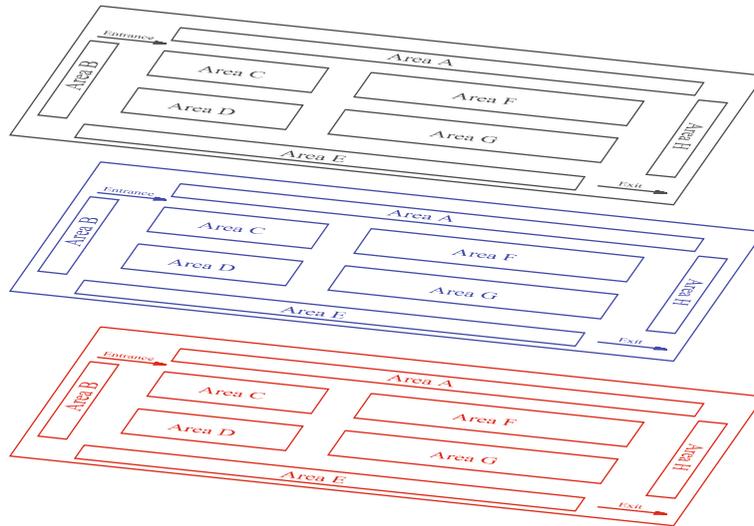


FIGURE 4: Layout of parking spaces in the multistorey car park.

As shown in Figure 2, the system will also recommend the nearest vacant parking space based on the algorithm ParkIG and implement real-time navigation to guide the user to the nearest vacant parking space to park.

2.5. System Architecture Design. The entire system design is mainly divided into three parts: the car park terminal, the parking space management system (management terminal), and the empty parking space navigation applet (user terminal). According to the above algorithm ParkIG, car park hardware deployment, MCP protocol, multistorey car park management system, and navigation miniprogram, the overall architecture of the solution is shown in Figure 3.

3. Results and Discussion

3.1. Environmental Model Building of Multistorey Car Park. Unlike urban roads, the roads of car parks are often not so

complicated to facilitate car owners to park [24, 25]. Figure 4 shows the layout of a common car park. The entrance and exit of the car park are set up separately, and there is only one. The parking spaces are distributed regularly, which is suitable for discussing path planning issues. According to the road environment of the car park, the entrance, intersection, and empty parking spaces are defined as nodes, and the distance between the intersection and the intersection is defined as the weight value to form a weighted undirected graph.

Many factors should be considered when empty parking spaces are recommended in the car park: the factors of the parking spot itself, the distance between the empty parking space and the pedestrian elevator, the distance among the unoccupied parking spot and the outlet, and the distance among the empty parking spot and the entrance [26]. However, priority was given to the factor of path distance from the entrance to the parking spot where no car is parked

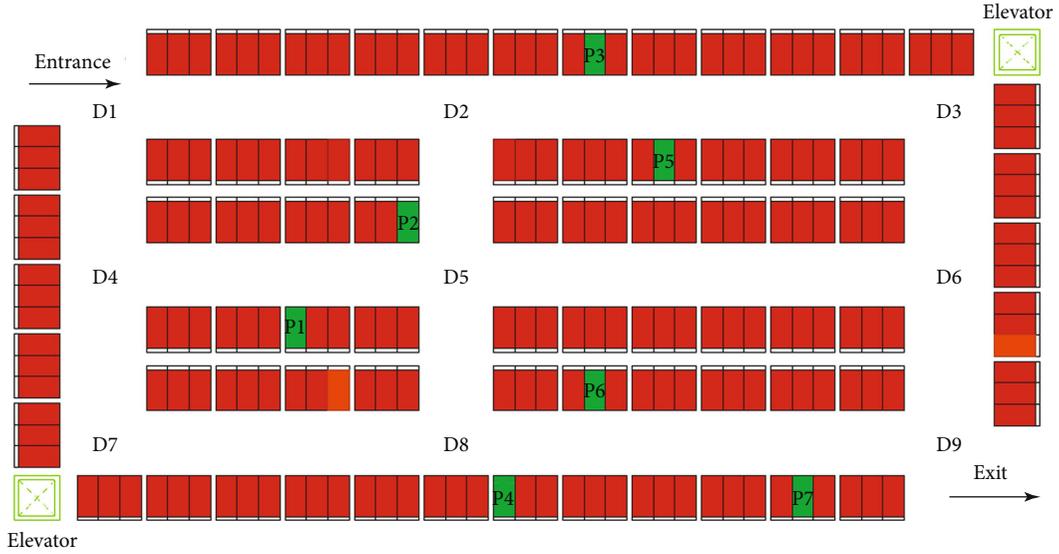


FIGURE 5: Occupancy of parking spaces at a certain time.

TABLE 3: Comparison of two algorithm results.

	Recommended parking space	Path	Path distance (m)	Time-consuming (ms)
Dijkstra	P1	M \rightarrow D1 \rightarrow D4 \rightarrow P1	45.72	88.79
ParkIG	P1	M \rightarrow D1 \rightarrow D4 \rightarrow P1	45.72	55.82

because of the urgent need of car owners to park. Set the empty parking space of the car park as $P(i)$, the length from the entrance to the empty parking space is $S(i)$, and then, the minimum path length is $\min \{S(i)\}$.

3.2. Experimental Environment. In the Windows operating system, PyCharm 2019.3.3 is used to write programs to simulate path planning, and MySQL database is used to store the relative coordinates of the parking space and whether the parking space is empty or not. Navicat Premium 12 was used to facilitate the operation of the database.

3.3. Analysis and Simulation of ParkIG Algorithm. Based on the algorithm ParkIG mentioned above, an example was selected for analysis. The hardware collector collected the car park occupation at a certain time, as shown in Figure 5. In the picture, the red parking spot represents that the parking spot has vehicles parked and the parking is regulated, the green parking space means that there is no parking in the parking space, and the orange parking spot means that the car is parked irregularly in the parking spot.

The algorithm ParkIG starts at the entrance and ends at K candidate parking spaces that may be the optimal empty parking spaces. When $K = 5$ is set here, the results obtained by the two algorithms on the same test data are the same. Both algorithms can successfully find the nearest empty parking space and plan the optimal route. However, the algorithm ParkIG takes less time than the traditional Dijkstra algorithm. As can be seen from Table 3, when the park-

ing space is in this state, the time it takes to run the algorithm has been reduced by about 37%.

As shown in Figure 6, the same experiment was conducted for the other parking spaces in other states, and it was found that the results of the two algorithms were the same each time, and the running time of the algorithm ParkIG was relatively less. The algorithm ParkIG reduces the average time by about 25.8%.

In the same situation, we use the algorithm ParkIG and the algorithm Dijkstra to search for empty parking spaces and route planning and count the size of the update domain and search domain of each node. The results are shown in Figure 7. We have counted the data of 8 nodes. In any case, the size of the update field and search field of the algorithm Dijkstra is 10, but the algorithm ParkIG is between 2 and 5, which reduces invalid operations by 70% on average.

4. Conclusions

This paper proposes a set of complete solutions for the defects of the existing multilevel parking navigation technology. The scheme includes the layout control of the infrared photoelectric switch at the car park, the transmission protocol between the parking lot and the administrator, various statistical reports of the administrator, and the optimal empty parking space recommendation and route guidance based on the algorithm ParkIG. It not only greatly facilitates the car park management and statistics of the car park manager but also greatly facilitates the

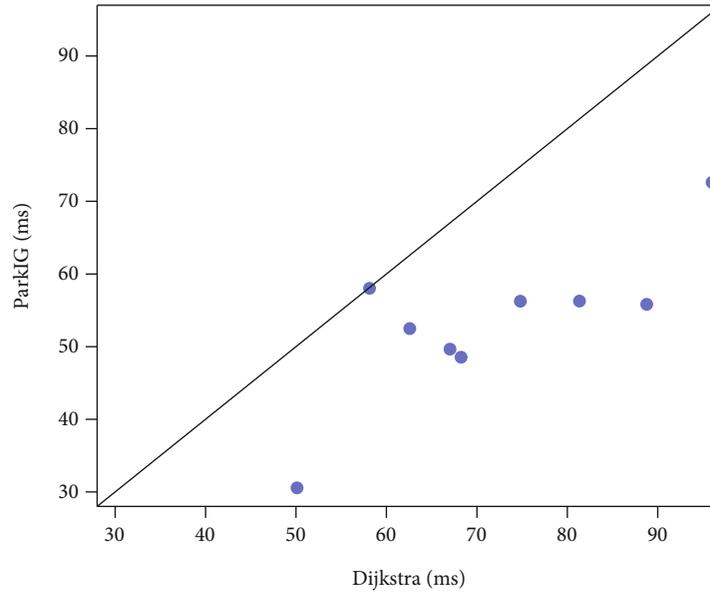


FIGURE 6: Comparison of the time spent by the two algorithms.

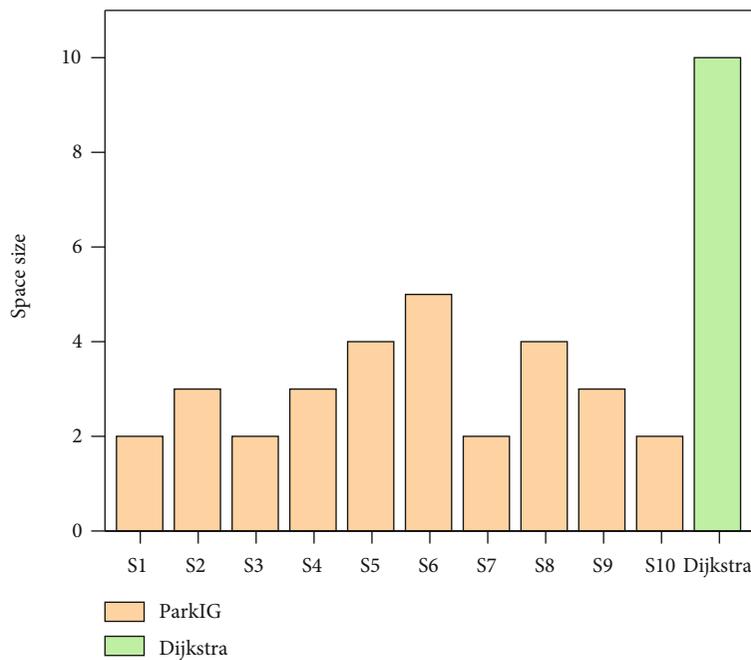


FIGURE 7: Comparison of search domain and update domain space size.

parking of the car owner. We also targeted a simulation for a classic car park, and the simulation results are achieved the expected level.

The main innovation of this paper is the arrangement of a certain regular infrared photoelectric switch to monitor the situation of the parking slots in real-time, and the MCP protocol is proposed to analyze the data transmitted by the hardware. Based on the MCP protocol and real-time parking space data, the algorithm ParkIG is proposed to achieve the search and path planning of empty parking spaces in multi-storey car parks. Thus, the efficiency of route planning is improved to ensure a good parking experience for car

owners, which has high practical application value. In addition, it also visualizes and analyzes the data in the car park, which is convenient for the management of the car park administrator.

Nevertheless, we still need to improve some areas in this paper:

- (1) Because the car owners are recommended for empty parking slots at the entrance and the path is calculated, we cannot predict the changes after the owner enters the car park, nor can we make adjustments based on the real-time situation

- (2) For very large car parks, this article does not provide further proof to ensure the effectiveness of the system, so the system is currently only suitable for small- and medium-sized car parks
- (3) In this paper, the weight calculation method is relatively simple, which needs further improvement and strict comparison with other methods, such as the predictive control method based on the neural network [27]
- (4) Regarding the parking payment module, the automatic payment is not completely realized in this paper [28]. The function of payment can be added to the user miniprogram, so that users do not need to pay at the exit, which greatly improves the user's parking experience [29]. Parking will become increasingly smarter in the future as smart parking solutions are continuously optimized

Data Availability

All data are available within the article.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Authors' Contributions

Zhendong Liu, Dongyan Li, Xi Chen, Xinrong Lv, and Yurong Yang contributed equally to this work.

Acknowledgments

The paper was greatly supported by the NNSF (National Natural Science Foundation of China), with Grant Nos.61672328 and 61672323, and the research is also supported by the Science and Research Plan of Luoyang Branch of Henan Tobacco Company (No. 2020410300270078).

References

- [1] M. Saberi, H. Hamedmoghadam, M. Ashfaq et al., "A simple contagion process describes spreading of traffic jams in urban networks," *Nature Communications*, vol. 11, no. 1, article 1616, 2020.
- [2] L. Wang, J. Chen, X. Cao, J. Chen, and C. Zhang, "Vehicle delay model applied to dynamic and static traffic impact analysis of large parking lots," *Applied Sciences*, vol. 11, no. 20, p. 9771, 2021.
- [3] M. Pereda, J. Ozaita, I. Stavrakakis, and A. Sánchez, "Competing for congestible goods: experimental evidence on parking choice," *Scientific Reports*, vol. 10, no. 1, article 20803, 2020.
- [4] C. Lai, Q. Li, H. Zhou, and D. Zheng, "SRSP: a secure and reliable smart parking scheme with dual privacy preservation," *IEEE Internet of Things Journal*, vol. 8, no. 13, pp. 10619–10630, 2021.
- [5] G. W. Yoon, J. B. Kim, and B. K. Kim, "An efficient urban outdoor localization and navigation system for car-like mobile robots," *Journal of Institute of Control, Robotics, and Systems*, vol. 19, no. 8, pp. 745–754, 2013.
- [6] Y. Tazaki and Y. Yokokohji, "Outdoor autonomous navigation utilizing proximity points of 3D Pointcloud," *Journal of Robotics and Mechatronics*, vol. 32, no. 6, pp. 1183–1192, 2020.
- [7] T. Lei and Y. Chu, "Analysis on the current development of indoor navigation technology," *Modern Trade Industry*, vol. 41, no. 24, p. 153, 2020.
- [8] Z. Jiang, Y. Zhou, L. Wei, and X. Lu, "Research and realization of key technologies of indoor navigation system," *Computer Programming Skills and Maintenance*, vol. 16, pp. 26–28, 2017.
- [9] Z. Zhang, "Investigation and analysis of the status quo of parking lot planning and management in Handan City," *Technological Wind*, vol. 2018, no. 9, pp. 163–166, 2018.
- [10] M. Li, F. Zhang, and J. Fang, "Optimal path solution based on dijkstra algorithm," *Frontiers in Economics and Management*, vol. 2, no. 6, pp. 170–176, 2021.
- [11] J. Wang, Y. Sun, Z. Liu, P. Yang, and T. Lin, "Route planning based on Floyd algorithm for intelligence transportation system," in *2007 IEEE International Conference on Integration Technology*, pp. 544–546, Shenzhen, China, 2007.
- [12] M. F. Gerard, G. Stegmayer, and D. H. Milone, "Metabolic pathways synthesis based on ant colony optimization," *Scientific Reports*, vol. 8, no. 1, article 16398, 2018.
- [13] S. Wang, G. Tan, X. Jiang, and C. Su, "Mobile robot path planning based on improved a~* algorithm," *Computer Simulation*, vol. 38, no. 9, pp. 386–389, 2021.
- [14] J. Zhao, Q. Wu, J. Chen, and Y. Huang, "Parking, intelligent parking system," *2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, vol. 1, pp. 2262–2267, 2019.
- [15] S. Wang and A. Li, "Multi-adjacent-vertexes and multi-shortest-paths problem of Dijkstra algorithm," *Computer Science*, vol. 41, no. 6, pp. 217–224, 2014.
- [16] T. Fu, P. Liu, K. Liu, and P. Li, "Privacy-preserving vehicle assignment in the parking space sharing system," *Wireless Communications and Mobile Computing*, vol. 2020, Article ID 8862652, 13 pages, 2020.
- [17] H. Wang, L. Zhu, and J. Wang, "Research on path planning of parking system based on Dijkstra-ant colony algorithm," *Journal of Engineering Design*, vol. 23, no. 5, pp. 489–496, 2016.
- [18] D. U. Pizzagalli, S. F. Gonzalez, and R. Krause, "A trainable clustering algorithm based on shortest paths from density peaks," *Science Advances*, vol. 5, no. 10, article eaax3770, 2019.
- [19] Z. Liu, D. Li, Y. Yang, X. Li, X. Lv, and X. Chen, "Design and implementation of the optimization algorithm in the layout of parking lot guidance," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 6639558, 2021.
- [20] C. Han, F. Liu, H. Li, and L. Wang, "Application of Dijkstra algorithm in parking guidance," *China New Communications*, vol. 21, no. 6, p. 167, 2019.
- [21] W. Cho, S. Park, M. Kim et al., "Robust parking occupancy monitoring system using random forests," in *2018 International Conference on Electronics, Information, and Communication (ICEIC)*, pp. 1–4, Honolulu, HI, USA, 2018.
- [22] M. Bachani, U. M. Qureshi, and F. K. Shaikh, "Performance analysis of proximity and light sensors for smart parking," *Procedia Computer Science*, vol. 83, pp. 385–392, 2016.
- [23] Y. Cha, "Key technologies of cache and computing in 5G mobile communication network," *Procedia Computer Science*, vol. 83, pp. 385–392, 2016.

- [24] Y. Mostafa, "A new shape descriptor for road network separation from parking lots and intersection detection on VHR remote sensing images," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 3099272, 2021.
- [25] P. Colonna, P. Intini, N. Berloco, V. Fedele, G. Masi, and V. Ranieri, "An integrated design framework for safety interventions on existing urban roads—development and case study application," *Safety*, vol. 5, no. 1, p. 13, 2019.
- [26] Y. Zhang and S. Tian, "Application of Dijkstra optimization algorithm in parking guidance system of parking lot," *Computer Measurement and Control*, vol. 22, no. 1, pp. 191–193, 2014.
- [27] J. Shin, H. Jun, and J. Kim, "Dynamic control of intelligent parking guidance using neural network predictive control," *Computers & Industrial Engineering*, vol. 120, pp. 15–30, 2018.
- [28] M. Ge and Z. Zhao, "Internet of things indoor positioning system based on Bluetooth technology," *Internet of Things Technology*, vol. 11, no. 11, pp. 52–57, 2021.
- [29] S. Xiang and Z. Chunhua, "On-line monitoring and auxiliary system for intelligent parking based on embedded design," in *2020 12th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA)*, pp. 308–312, Phuket, Thailand, 2020.