Journal of Advanced Transportation

Emerging Technologies and Methods in Shared Mobility Systems

Lead Guest Editor: Zhuping Zhou
Guest Editors: Dongfang Ma and Zhao Xing

Emerging Technologies and Methods in Shared Mobility Systems

Journal of Advanced Transportation

Emerging Technologies and Methods in Shared Mobility Systems

Lead Guest Editor: Zhuping Zhou
Guest Editors: Dongfang Ma and Zhao Xing

Copyright © 2021 Hindawi Limited. All rights reserved.

This is a special issue published in "Journal of Advanced Transportation." All articles are open access articles distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Associate Editors

Juan C. Cano, Spain
Steven I. Chien, USA
Antonio Comi, Italy
Zhi-Chun Li, China
Jinjun Tang, China

Academic Editors

Kun An, China
Shriniwas Arkatkar, India
José M. Armingol, Spain
Socrates Basbas, Greece
Francesco Bella, Italy
Abdelaziz Bensrhair, France
Hui Bi, China
María Calderon, Spain
Tiziana Campisi, Italy
Giulio E. Cantarella, Italy
Maria Castro, Spain
Mei Chen, USA
Maria Vittoria Corazza, Italy
Andrea D'Ariano, Italy
Stefano De Luca, Italy
Rocío De Oña, Spain
Luigi Dell'Olio, Spain
Cédric Demonceaux, France
Sunder Lall Dhingra, India
Roberta Di Pace, Italy
Dilum Dissanayake, United Kingdom
Jing Dong, USA
Yuchuan Du, China
Juan-Antonio Escareno, France
Domokos Esztergár-Kiss, Hungary
Saber Fallah, United Kingdom
Gianfranco Fancello, Italy
Zhixiang Fang, China
Francesco Galante, Italy
Yuan Gao, China
Laura Garach, Spain
Indrajit Ghosh, India
Rosa G. González-Ramírez, Chile
Ren-Yong Guo, China
Yanyong Guo, China
Jérôme Ha#rri, France
Hocine Imine, France
Umar Iqbal, Canada
Rui Jiang, China
Peter J. Jin, USA
Sheng Jin, China
Victor L. Knoop, The Netherlands
Eduardo Lalla, The Netherlands
Michela Le Pira, Italy
Jaeyoung Lee, USA
Seungjae Lee, Republic of Korea
Ruimin Li, China
Zhenning Li, China
Christian Liebchen, Germany
Tao Liu, China
Chung-Cheng Lu, Taiwan
Filomena Mauriello, Italy
Luis Miranda-Moreno, Canada
Rakesh Mishra, United Kingdom
Tomio Miwa, Japan
Andrea Monteriù, Italy
Sara Moridpour, Australia
Giuseppe Musolino, Italy
Jose E. Naranjo, Spain
Mehdi Nourinejad, Canada
Eneko Osaba, Spain
Dongjoo Park, Republic of Korea
Luca Pugi, Italy
Alessandro Severino, Italy
Nirajan Shiwakoti, Australia
Michele D. Simoni, Sweden
Ziqi Song, USA
Amanda Stathopoulos, USA
Daxin Tian, China
Alejandro Tirachini, Chile
Long Truong, Australia
Avinash Unnikrishnan, USA
Pascal Vasseur, France
Antonino Vitetta, Italy
S. Travis Waller, Australia
Bohui Wang, China
Jianbin Xin, China

Hongtai Yang, China
Vincent F. Yu, Taiwan
Mustafa Zeybek, Turkey
Jing Zhao, China
Ming Zhong, China
Yajie Zou, China

Contents

A Distribution Model for Shared Parking in Residential Zones that Considers the Utilization Rate and the Walking Distance

Wenhui Zhang, Fan Gao, Shurui Sun, Qiuying Yu, Jinjun Tang, and Bohang Liu

Research Article (11 pages), Article ID 6147974, Volume 2020 (2020)

Hindawi
Journal of Advanced Transportation
Volume 2021, Article ID 8885671, 16 pages
https://doi.org/10.1155/2021/8885671

Research Article

Recurrent Neural-Based Vehicle Demand Forecasting and Relocation Optimization for Car-Sharing System: A Real Use Case in Thailand

Peerapon Vateekul,[1] Panyawut Sri-iesaranusorn,[1,2] Pawit Aiemvaravutigul,[1] Adsadawut Chanakitkarnchok,[1] and Kultida Rojviboonchai[1]

[1]*Chulalongkorn University Big Data Analytics and IoT Center (CUBIC), Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University, Bangkok, Thailand*
[2]*Division of Information Science, Nara Institute of Science and Technology, Nara, Japan*

Correspondence should be addressed to Kultida Rojviboonchai; kultida.r@chula.ac.th

Received 26 June 2020; Revised 3 January 2021; Accepted 12 January 2021; Published 28 January 2021

Academic Editor: Dongfang Ma

Copyright © 2021 Peerapon Vateekul 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.

A car-sharing system has been playing an important role as an alternative transport mode in order to avoid traffic congestion and pollution due to a quick growth of usage of private cars. In this paper, we propose a novel vehicle relocation system with a major improvement in threefolds: (i) data preprocessing, (ii) demand forecasting, and (iii) relocation optimization. The data preprocessing is presented in order to automatically remove fake demands caused by search failures and application errors. Then, the real demand is forecasted using a deep learning approach, Bidirectional Gated Recurrent Unit. Finally, the Minimum Cost Maximum Flow algorithm is deployed to maximize forecasted demands, while minimizing the amount of relocations. Furthermore, the system is deployed in the real use case, entitled "CU Toyota Ha:mo," which is a car-sharing system in Chulalongkorn University. It is based on a web application along with rule-based notification via Line. The experiment was conducted based on the real vehicle usage data in 2019. By comparing in real environment in November of 2019, the results show that our model even outperforms the manual relocation by experienced staff. It achieved a 3% opportunity loss reduction and 3% less relocation trips, reducing human effort by 17 man-hours/week.

1. Introduction

With the growth in population and economy, a car-sharing system becomes an alternative public transportation that alleviates road traffic congestion [1]. Experts in the transportation field predicted that car sharing can significantly increase in the next ten years, especially in Asia-Pacific [2], and can become a possible bridging mode between private cars and traditional public transportation such as bus and train. Furthermore, the number of car-sharing users is predicted to reach 36 million accounts, and the value of the global car-sharing market is forecast to be 11 billion US dollars in 2024 [3].

From the past to the present, there have been two main design choices for a car-sharing system. The first design is a round-trip car-sharing system. In this design, a customer needs to reserve a vehicle in advance and return the vehicle at the station they picked. The other design is a one-way car-sharing system. Multiple vehicles are available for customers at multiple stations. A customer can pick a vehicle from a station and return the vehicle at any station in the operation area. If the one-way car-sharing system operates efficiently, it can be more beneficial for both customers and operators, compared to the round-trip car-sharing system [4]. However, in the one-way car-sharing system, the number of vehicles at each station can become imbalanced in the sense

that too many vehicles are available at the low-demand stations while the high-demand stations contain very few vehicles [5]. Such a situation leads to opportunity loss for the car-sharing system. In order to reduce the opportunity loss, an effective vehicle relocation strategy needs to be considered.

Several strategic solutions have been proposed for vehicle relocation in the one-way car-sharing system, which can be classified into two categories. The first category is the user-based strategy, in which customers/users are incentivized by marketing campaigns and then help relocating the vehicles [6–11]. This strategy easily scales with the number of users and does not need efforts from the vehicle operators. Nevertheless, it is not guaranteed that the users will always follow the vehicle relocation plan as expected. The second category is the operator-based strategy, in which the vehicle operator forecasts the demands at each station and then sends the operation staff to relocate the vehicles according to the demand forecasting results. This strategy ensures that the vehicle relocation tasks are done as needed but requires highly experienced staff to work for the vehicle relocation [8, 12–18]. In [12, 15], solutions based on the mathematical models and rule-based algorithms were proposed. In [13], the battery consumption and recharging issues were considered in order to optimize the relocation operation. Recently, in [14, 19], the deep neural network for demand management in a car-sharing system is introduced. Although all of the aforementioned solutions show improvements in relocation operations based on the realistic data, none of them has been implemented in the real-world car-sharing system.

In this paper, we propose a novel vehicle relocation system with major improvements in data preprocessing, demand forecasting, and relocation optimization. Our proposed system is designed for one-way car-sharing system with an operator-based strategy. Firstly, the data preprocessing automatically removes fake demands caused by search failures and application errors. Then, the real vehicle demand at each station is forecasted using a deep learning approach called Bidirectional GRU. We have described the reason for using BiGRU in Section 4.2. Finally, the Minimum Cost Maximum Flow algorithm is deployed to maximize the vehicle availability according to the forecasted demands, while minimizing the amount of relocations. Furthermore, our proposed system is deployed in the real use case entitled, "CU Toyota Ha:mo," which is a car-sharing system in Chulalongkorn University, Bangkok, Thailand. Monitoring dashboard and relocation notification for the operation staff are also implemented in our system. In the experiment, the model was trained using the usage log in 2019. The results show that the relocation plan prepared by our model is better than the plan managed by human experts. It can reduce human effort to a great extent. Furthermore, the system has been deployed in the real-world scenario. The results show that our system can successfully operate and replace the existing human process at the operation room.

In summary, our contributions compared to prior attempts are as follows: The main goal of our work is to propose the whole vehicle relation system in the real-world situation. To provide a suitable relocation plan, it is crucial to have vehicle demands in the future. Thus, ours is the first research work that contains both vehicle demand forecasting and relocation plan optimization in the real-world situation. However, all prior attempts [9, 10, 12–15, 17] are just based on a simulated environment, and none of them proposed both modules as a complete solution. In demand forecasting, we are one of the pioneers who have applied and compared various kinds of deep learning networks. Furthermore, the new loss function has been proposed to take both demand from departure and destination stations into consideration. Also, the vehicle assignment procedure is proposed as a post-processing step of our demand forecasting model. From an intensive experiment, our results show that Bi-GRU outperformed the previous approach, LSTM [14, 19]. For relocation optimization, the optimization approach called "Minimum Cost Maximum Flow" is applied in order to achieve the best relocation plan, while others [11, 16] are based on a simple rule on a simulated environment.

The remainder of this paper is organized as follows: Section 2 discusses the related works. Section 3 explains the overall system and the existing operation of CU Toyota Ha:mo. Our proposed vehicle relocation system including data preprocessing, demand forecasting, and relocation optimization are described in Section 4. The performance evaluation is discussed in Section 5. The conclusion is elaborated in Section 6.

2. Related Works

Vehicle relocation strategies can be classified into two main categories: the user-based strategy and the operator-based strategy [8]. The user-based strategy is based on the users who are available or comfortable to help perform the required relocation tasks. Basically, some incentives such as discounts or free trips are offered to the users. Febbraro et al. [6] proposed a discount for users who volunteer to perform vehicle relocation. Barth et al. [7] introduced two user-based relocation methods called trip joining and trip splitting. When the system realizes that it is becoming imbalanced, it will split users that have more than one passenger to relocate multiple vehicles to the destination station. Conversely, if two users are taking the exact same route at the same time, the system will request the two users to merge into one usage. In addition, Cepolina and Farina [9] investigated a fully user-based vehicle relocation for a car-sharing system in the urban areas. In order to deal with the unbalanced demand, their proposed system notifies the users who are interested in moving the vehicles to the target parking lots. This is not only good for the system management but also benefits the users. Recently, Angelopoulus et al. [10] introduced the user-based strategy for e-motorbike sharing. The idea behind is to combine the public transportation information and graph theory to grant discounts to users. Furthermore, to evaluate the idea, they applied their strategy for a free-floating system in the city of Vitoria-Gasteiz, Spain. These previous researches show that there are various solutions for the user-based strategy. In this paper, our scenario is based on a car-sharing system in a campus of the university. So, the operator-based strategy is simpler and more efficient since it is suitable for a small area (campus).

Contrary to the user-based strategy, the operator-based strategy is based on vehicle relocation performed by the operation staff. Herbawi et al. [12] identified the vehicle relocation as an NP-hard optimization problem and proposed an evolutionary algorithm for a car-sharing system. Their algorithm was evaluated on real-world data published by a car rental and sharing company, "car2go." They also investigated the system adaptation on the different parameters, such as the maximum allowed duration. Gambella et al. [13] considered the vehicle relocation from the viewpoint of the service provider. The objective is to maximize the profit associated with the trips performed by the users. They introduced a mathematical model for managing staff assignments in a large-scale, car-sharing system by considering battery consumption and recharging processes. Their solutions were developed and tested on a set of realistic data derived from an existing car-sharing system. According to their result, solving the relocation in a car-sharing system is worth and makes the service provider achieve larger profits. By considering the operation staff's effort and cost, Kek et al. [15] presented a decision support system to determine a set of near-optimal manpower and operating parameters for the vehicle relocation problem. To evaluate and test their system, a simulation was conducted using a data set of commercially operational data from a car-sharing system in Singapore. Regarding the electric vehicle (EV), in [13, 16], the proper solution for EV is examined because the constraints such as charging stations need to be considered. Lu et al. [17] introduced the solution based on matching between the user request and the remaining quantity of electric of EV on simulation. Martin et al. [18] applied the Markov model, a stochastic model which models temporal or sequence data, for relocation tasks in one-way. With the increasing popularity of deep learning, Qu et al. [20] used a simple feed-forward model with historical traffic flow data and contextual factor data for daily long-term traffic flow forecasting. Yu et al. [14] introduced an analytical system that deployed Long Short-Term Memory (LSTM) for forecasting short-term vehicle demand. They also used multiple temporal features, including the time of day, day of week, and weather condition. For evaluation, they used real-world data from a car-sharing system in Chengdu. Then, they demonstrated the reaction of the car-sharing market toward the operating strategies based on the data from two service providers. Ning et al. [19] also used LSTM for demand forecasting, and confirmed the simulation results using a statistical hypothesis test called "Granger causality". Zhaowei et al. [21] used M-B-LSTM, an end-to-end hybrid deep learning model, for short-term traffic flow forecasting. Ma et al. [22] applied a clustering technique to create many sub-models based on deep neural networks. Then, they [23] proposed the contextual convolutional recurrent neural model to extract inter-and-intra-day traffic patterns for daily traffic flow forecasting. Motivated by the results of these researches, we investigated using a variety of recurrent neural network architecture for a similar purpose. In our scenario, the optimization algorithm must be computationally efficient since the relocation schedule should be run often (hourly) due to a limited number of sharing cars; therefore, the evolutionary algorithm is not suitable for our scenario because of its high computation cost. Also, more variations of deep learning algorithms, for example, GRU and Bidirectional GRU, are investigated in our work apart from LSTM. We not only just employ those networks but also design our model specifically for the car-sharing system.

Even though the previous studies on the user-based and the operator-based strategies have shown outstanding performance, they were mostly evaluated by feeding real-world data to their solution offline, while results from the actual deployment were not reported. This inspires us to evaluate our solution not only by using realistic data but also by deploying it in the car-sharing system operating in the real world. Furthermore, most prior works focused on traffic forecasting in an open environment, which allows the unlimited number of vehicles in open areas. It is quite different from our vehicle relocation, in which the number of stations and vehicles is fixed.

3. CU Toyota Ha:mo System

Since 2017, Chulalongkorn University and Toyota have collaboratively launched a car-sharing system entitled "CU Toyota Ha:mo" (henceforth referred to as Hamo), as shown in Figure 1. With Hamo's system, university personnel, such as students, teachers, and staff, can go anywhere in the campus quickly. This system provides them an environment-friendly alternative mode of transportation that complements a public transport within the university. The system performed well in its early stages, but as it is, it cannot avoid opportunity loss problems when it has more users. To deal with this issue, Hamo's staff employ a manual relocation-based solution by approximating the number of vehicles at each station by their own experience. However, this solution is time-consuming and expensive, while also requiring staff's experience to be effective.

This section is organized as follows: Section 3.1 describes the dataset and the data collection process, as well as distinguishes between successful and failed usages. Section 3.2 explains Hamo's existing operation and the problems it currently faces.

3.1. Dataset. Our dataset contains the history of shared vehicle reservations and usages in Hamo's service. These data were collected through Hamo's mobile application, where users are required to make reservations before picking up and using the vehicles. As shown in Figure 2, each entry contains information on the user's trip, for example, the departure station, the destination station, and the reservation duration. These reservation records can be used to predict the demand at a given time in the future. According to the data, we define a *successful usage* as a feasible reservation for which a complete trip can be provided, and a *failed usage* as an unsuccessful reservation where the desired trip cannot be fulfilled. The failed usages can occur due to lack of vehicles at the departure station and/or unavailability of parking lots at the destination station, leading to opportunity losses in the system.

(a) (b)

Figure 1: An example of a vehicle in CU Toyota Ha:mo system (a) being operated by a user and (b) being parked at a charging station.

	UserID	Date	Reservation time	Departure station	Destination station	End usage time
0	ID#60431	2017-12-01	07:18:06	01 exit cham square	07 chamchuri 5	07:44:52
1	ID#95550	2017-12-01	07:20:35	12 CU terrace	04 engineering	07:44:26
2	ID#17356	2017-12-01	08:14:03	12 CU terrace	04 engineering	08:23:57
3	ID#52570	2017-12-01	08:40:04	03 sala prakeaw	02 economics	08:46:27
4	ID#52570	2017-12-01	08:47:49	02 economics	03 sala prakeaw	08:53:13
5	ID#76189	2017-12-01	09:14:27	12 CU terrace	03 sala prakeaw	09:23:33

(a)

	UserID	Date	Reservation time	Departure station	Destination station	Status
0	ID#90382	2017-12-01	07:06:19	01 exit to cham square	05 arts	1
1	ID#85616	2017-12-01	07:14:40	06 chamchuri 9	06 chamchuri 9	2
2	ID#85616	2017-12-01	07:14:47	06 chamchuri 9	03 sala prakeaw	2
3	ID#85616	2017-12-01	07:15:08	01 exit to cham square	01 exit to cham square	1
4	ID#51386	2017-12-01	07:16:35	01 exit to cham square	01 exit to cham square	1
5	ID#51386	2017-12-01	07:16:36	01 exit to cham square	01 exit to cham square	1

(b)

Figure 2: The log of Hamo's existing system. (a) The log of successful usages and (b) the log of failed usages (Status 1 means no available vehicle at the departure station and Status 2 means no available parking lot at the destination station).

To provide Hamo's service for customers, there are 22 stations and 30 vehicles available from 7 a.m. to 7 p.m. on weekdays. The service is closed on weekends and national holidays. Figure 3 shows the service area in the university campus. In our study, we used the data recorded from December of 2017 to October of 2019. According to the data, user behaviors can be analyzed. For example, it can be seen that a lot of usage records were created 30 minutes before 9 a.m., which was the starting time of the first lecture. Moreover, it can be seen that only a few usage records were made during the ongoing time of the lecture.

Figure 4 shows the number of failed usages separated by stations in November of 2018. As can be seen, the top three stations that have the highest number of failed usages are stations 4, 12, and 2, respectively. This is because these stations are located near the shopping mall and/or the faculty with a high number of students. On the other hand, stations 13, 17, and 20 are the top three stations that have the lowest number of failed usages. This is because these stations are located near the faculty with a low number of students and/or difficult to arrive on foot. According to these results, we can know which stations have a high demand, and which stations have a low demand. Thus, at a given time, we can design the vehicle relocation based on the demand in each station. In our design, such information will be considered together with other important factors such as distances between two stations and staff's workloads.

3.2. Hamo's Existing Operation. Traditionally, Hamo's staff have resolved the opportunity-loss problem by performing the following statistical analysis. First, the operation staff divide the operating time into multiple time slots and then manually assess the demands at each station in each time slot. The ranking of the stations with the highest demand to the lowest demand is presented in a table. Next, vehicles will be assigned to each station ordered by its forecasted demand, so the station with higher demand will be assigned first. Also, we are going to leave at least one parking lot at each station for an incoming vehicle. In contrast, the stations with low demand will be assigned a few vehicles or none. Lastly, based on the table, the operation staff will relocate the appropriate amount of vehicles every two hours.

We investigated that in the data there exist a lot of *fake demands*. This occurred in the data records due to user/system

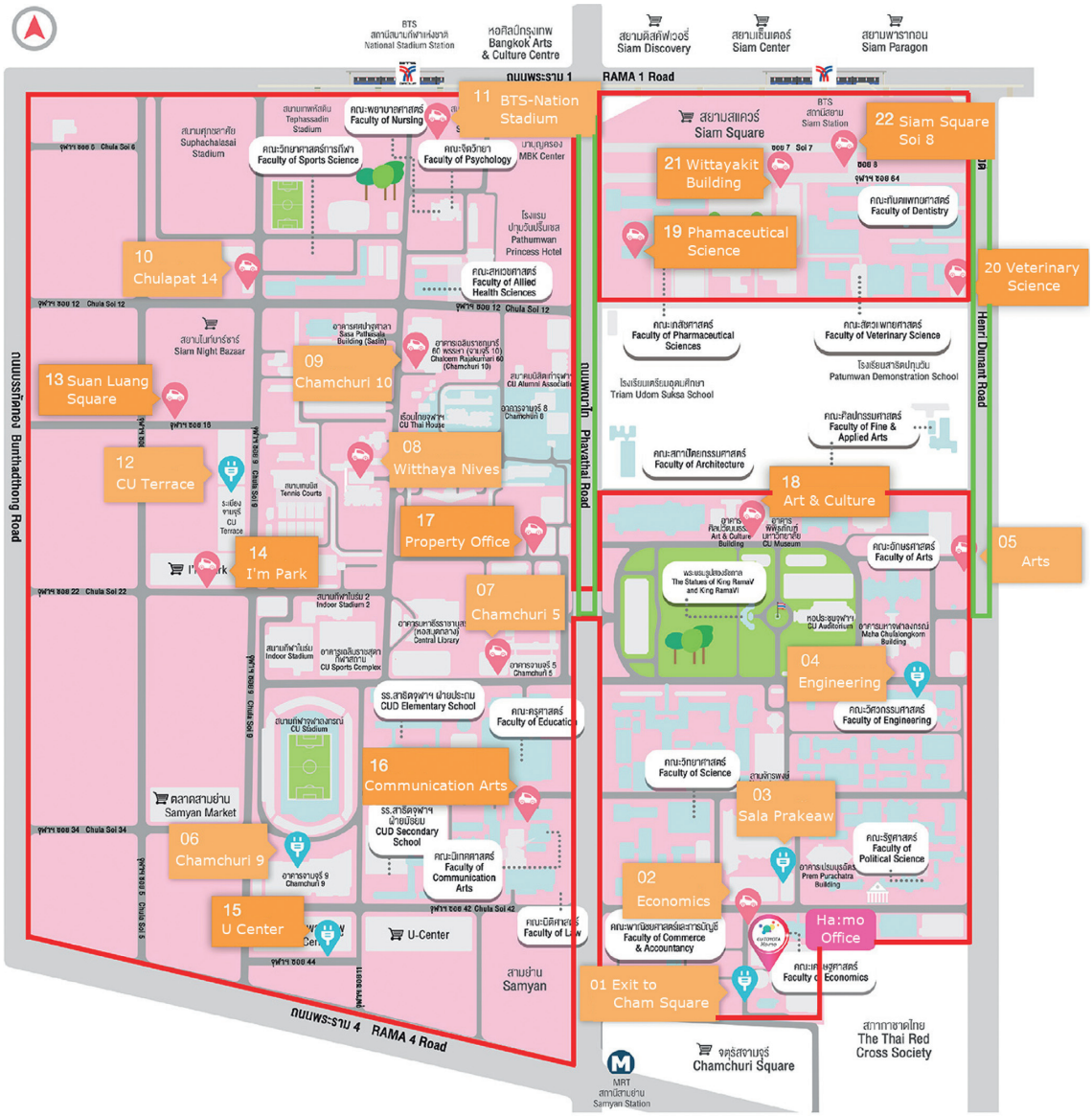


Figure 3: Service area in the university campus (source: CU Toyota Ha:mo website available at https://www.cutoyotahamo.com/en/graphic_map-2/).

errors. For example, some users repeat their reservation several times after their first attempt fails. This causes the number of reservation records no longer reflects the number of the actual usage. Another example is when the system creates multiple reservation records from users while they are driving the vehicles. This is a system error and should not happen in the system. However, since these fake demands actually appear in the data, the operation staff manually remove them one by one. This consumes a lot of time and effort of the staff.

4. Our Proposed System

The relocation system has been proposed as illustrated in Figure 5 and includes four modules: (i) automatic data preprocessing, (ii) vehicle demand forecasting, (iii) relocation optimization, and (iv) web application. Given the history of reservations of the service, the task of our system is to automatically determine optimal vehicle relocation trips for the staff. First, fake demands are removed with an automatic data preprocessing algorithm according to the staff's experience. Second, for each hour in the operating time, we normalize the departure demand fraction and the destination demand fraction, so that the summation of all stations is one at each time step. In our data set, the interval of data is hourly. We aim to forecast one day ahead, so there are 12 forecasted time steps representing 12 operating hours/day. The input window is 30 days prior to the forecasted day (360 time steps), and the output is one time step ahead with the rolling strategy to

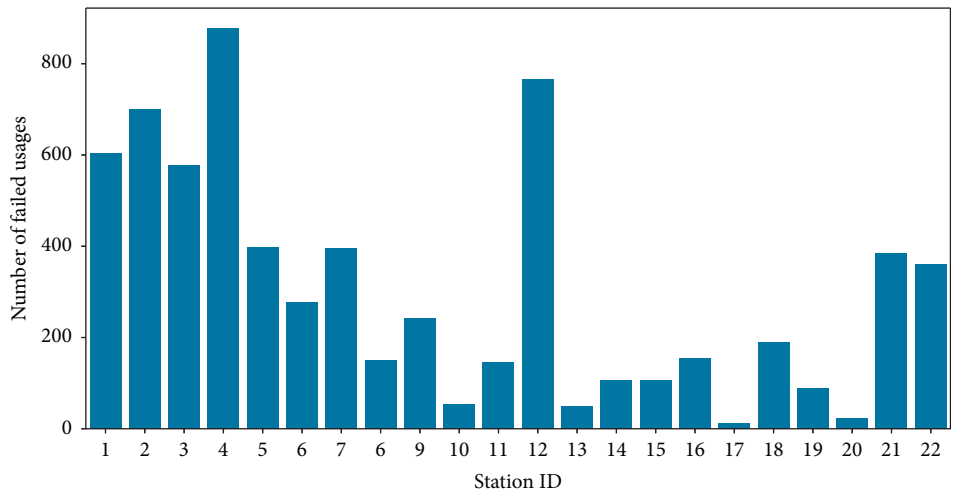


Figure 4: Failed usages at each station in November of 2018.

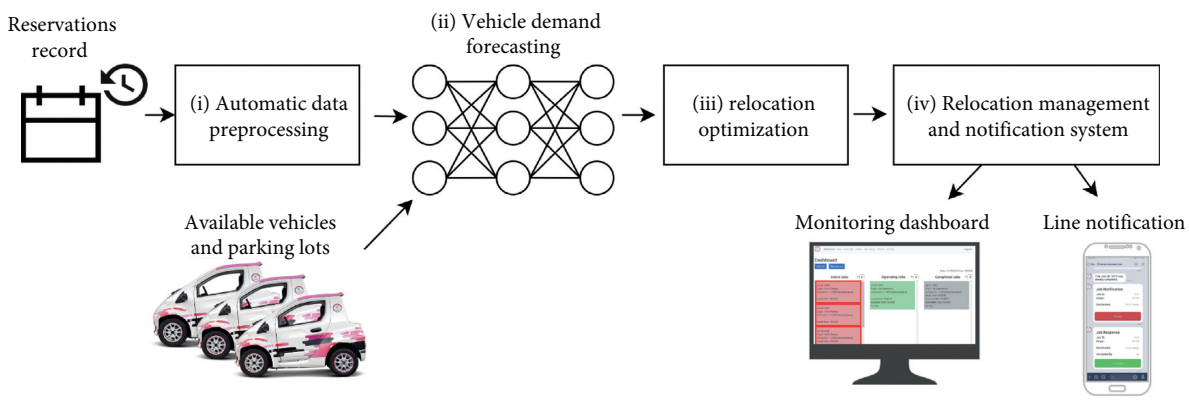


Figure 5: Overview of our proposed relocation system with four modules: (i) automatic data preprocessing, (ii) vehicle demand forecasting, (iii) relocation optimization, and (iv) web application.

complete the whole time steps of the forecasted day. The model's architecture is based on a recurrent neural network, which is suitable for dealing with time series data. The rankings are then transformed into suitable amounts of vehicles that each station should have. Third, the Minimum Cost Maximum Flow algorithm is applied on the existing and suitable amount of vehicles at each station, to find the optimal vehicle relocation trips for the staff. Finally, the relocation plans are deployed in our web application and they are automatically notified to all staff.

4.1. Automatic Data Preprocessing. To reduce the staff's workload on manual preprocessing, we implemented the following automatic data preprocessing based on the procedures described by the staff in order to remove fake demands. Our preprocessing is shown in Figure 6. First, we took the reservation data, a mix of real and fake demand, as the input. Then, based on actual usage records, we considered each user's extra reservations during their usage period to be fake demand, and discarded them.

There are two main incorrect demands: (i) abnormal search logs after successful booking and (ii) repeated reservation attempts. The abnormal search logs can be explained by an example; in the case of a successful usage of a user lasting from 7 : 00 p.m. to 7 : 30 p.m., any reservations from that user during that time should be fake demand. For the repeated reservation attempts, only the first reservation made is considered as the real demand, and according to staff's experience, any reservations made within the next 15 minutes are fake demands.

4.2. Vehicle Demand Forecasting. The approach of deep neural networks was selected due to its high performance in various fields such as computer vision, natural language processing, and traffic management [24–26]. In particular, Recurrent Neural Network (RNN) is a type of deep neural network that is suited to variable length time-series problems, allowing the use of past information to predict "future" values. One of the popular RNN models is Gated-Recurrent Unit (GRU), the performance of which can be compared with Long Short-Term Memory (LSTM) while using less number of parameters [27]. Since GRU computes the input only in the forward direction, Bidirectional GRU (BiGRU) is implemented to process the data in both directions. We also compare these three models, with a manual

```
Data : reservation logs
Result: demand logs with fake demand cleaned
sort all reservationsby event start time;
group all reservations by user;
for each user do
    // (i) remove abnormal search logs after sucessful booking
    for each reservation from user do
        if reservation is successful usage then
            While next reservation time < current reservation end time do
                delete next reservation;
            end
        end
    end
    // (ii) remove repeated reservation attempts
    for each reservation from user do
        if current reservation time – previous reservation time <= 15 min. then
            if current and previous reservations have the same or nearby departure and destination stations then
                delete current reservation;
            end
        end
    end
end
```

Figure 6: Automatic fake demand removal algorithm.

model by experts, and then select the model with minimum root mean square error (RMSE) and the best simulation result as the final model. This section proposes a deep neural network based on the BiGRU structure for demand forecasting.

The architecture of the proposed model is shown in Figure 7. The demand of each station is normalized to be a fraction, where the summation of all stations is one at each time step, separately between departure and destination demands. BiGRU was used to extract features that should capture the correlation between departure and destination demand at each station across time. To improve upon accuracy, temporal features are included in our model. However, the order of time should be treated cyclically, for example, daily, weekly, monthly, etc. Therefore, we add an auxiliary input of cyclical time features, consisting of representations of the hour of day, day of week, and month of year. The extracted features and the cyclical time features are concatenated, and finally used as an input in the next fully connected layer, to separately forecast the destination and departure demand in the future time step. In practice, to predict the next day, our model will predict for the first hour of that day using the last 30 day's data as the input window. Then, to predict the next hour, we discard the oldest data and instead add the previous hour's prediction to the input. This process is repeated until we obtain the full day's predictions. We describe each part of the model as follows:

(i) BiGRU: it is a recurrent neural network that has been proven to be effective and fast as explored in previous studies, especially in the domains of speech and bioinformatics [28–30]. BiGRU has the advantage over one-directional GRU in that it combines the outputs from a forward GRU with a backward unit, allowing it to capture relationships from both past and future time steps. In our network, we used two layers of BiGRU, each with 32 neural nodes, to extract features from departure and destination demand inputs.

(ii) Fully connected layer: after obtaining the extracted features, we feed them into fully connected layers to predict the *destination* and departure demand in the target time step. All hidden, fully connected layers in our model have 256 neural nodes each. Each layer's weights are initialized according to He et al. [31] because their strategy is suitable for layers with ReLU activation function, which we describe in a later section.

(iii) Activation function: rectified linear unit function (ReLU) was selected as an activation function to alleviate the vanishing gradient problem [32, 33]. We used this activation function for all hidden, fully connected layers.

(iv) Regularization: to increase the numerical stability of our neural network, we perform batch normalization after every fully connected layer [34]. This is done before applying the activation function. After each activation layer, we also employed *dropout*

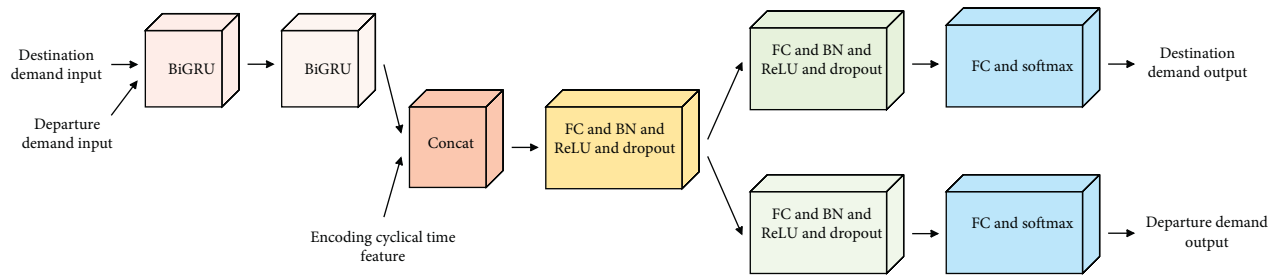


Figure 7: An architecture of our vehicle demand forecasting based on BiGRU.

layers to prevent overfitting by forcing the model not to rely on the same patterns all the time [35].

(v) Encoding cyclical time features: as we expect time in our data to be cyclical most of the time, we need to encode this information into the model's input, so that the model can treat all points in time the same way [36]. For example, we want the model to address the boundary between December and January the same way it does with January and February. By assigning numerical values to each month (0 for January, 1 for February... and 11 for December), the gap between December and January will be different from that of consecutive months. Instead, we encode all cyclical time features with sine and cosine transformations. Each cyclical time feature is calculated as shown in equations (1) and (2), where i is the numerical representation of time and T is the maximum value of that time feature. For example, T for the day of week feature will be 7, and i being 1 represents Monday.

$$\mathrm{cir}_x(i) = \sin\left(\frac{2\pi i}{T}\right), \tag{1}$$

$$\mathrm{cir}_x(i) = \cos\left(\frac{2\pi i}{T}\right). \tag{2}$$

Our model has two outputs, one for the departure demand and another for the destination demand. As both of them are fractions that sum up to one, we use softmax as an activation function for both output layers. From preliminary investigation, we found that using softmax as is gives the model a tendency to assign high fractions to only a few stations; the remaining stations' fractions are not learned properly. To deal with this issue, we used temperature scaling, which adds a hyperparameter to the softmax function that can be calibrated to smooth down this tendency and allow the model to learn fractions of the low-ranked stations as well [37]. The softmax function with temperature scaling is defined as follows:

$$\sigma(s_i) = \frac{\exp((1/T)s_i)}{\sum_{j=1}^{N}\exp((1/T)s_j)}, \tag{3}$$

where s_i is the predicted demand for station i, N is the total number of stations, and T is the temperature scaling hyperparameter.

To train the model, we used a loss function (L), or objective function, based on categorical cross-entropy. Normally, categorical cross-entropy is used in classification tasks, which has only one correct label. In this study, however, we applied this loss to make the model's output converge toward the ground truth fractions. Since our model has two outputs, our final loss function is the sum of categorical cross-entropy for both demand predictions. As a result, the loss function is defined as follows:

$$L = -\left(\sum_{i=1}^{N} t_{i,\mathrm{depart}}\log\left(s_{i,\mathrm{depart}}\right)\right) + \left(\sum_{i=1}^{N} t_{i,\mathrm{dest}}\log\left(s_{i,\mathrm{dest}}\right)\right), \tag{4}$$

where $t_{i,\mathrm{depart}}$ and $t_{i,\mathrm{dest}}$ are the ground truth fractions of departure and destination demand outputs for station i, respectively, $s_{i,\mathrm{depart}}$ and $s_{i,\mathrm{dest}}$ are the predicted demand fractions, and N is the total number of stations.

The predicted demands will be used to rank the stations, and the vehicle assignment table for the target time will be created accordingly. More specifically, the vehicle assignment process has to follow certain rules, such as the minimum requirement that almost all stations must have at least one vehicle and one empty parking lot. Here, we describe our vehicle assignment procedure that follows these rules:

(1) We assign one vehicle to every station, except some of the stations, because they have only one parking lot each and the nearby areas are not as crowded as the other stations.

(2) Following the ordering from the model's ranking output, we assign vehicles to the highest ranked station until there is only one remaining parking lot. For example, 3 vehicles are assigned to a station with 4 parking lots if the station is in the high ranking category.

(3) The process is repeated until there are no more vehicles remaining. The final output is a list that shows the required number of vehicles at each station for the target hour.

In practice, our system removes fake demand and retrains the model once a week. This generally helps to reduce computational costs and make the system capable of capturing any changes in demand.

4.3. Relocation Optimization. After obtaining the vehicle assignment table, the next step is to produce relocation trips for operation staff according to the number of available

```
Data: G: the network structure
Result: p: list path of minimum cost
k = maximum-flow(G);
p = [];
f = 0;
while f < k do
    t = shortest-path(G);
    m = infinity;
    for each node in t do
        if node.capacity < m then
            m = node.capacity;
    p.push(t);
    f = f + m;
    for each node in t do
        node.capacity = node.capacity - m;
return p;
```

Figure 8: Minimum Cost Maximum Flow algorithm.

vehicles currently presented at each station. In order to ensure that the produced trips are efficient in terms of distance, energy consumption, and staff's efforts, we applied the Minimum Cost Maximum Flow algorithm for the relocation optimization. This algorithm is well-known and widely used in cloud resource allocation [38], energy sharing management [39], resource management [40, 41], and network traffic management [42, 43]. The objective is to minimize the cost required to deliver the maximum amount of flows possible in a network. In our experiment, the nodes in the network represent the stations and the edges represent the distances between them.

The Minimum Cost Maximum Flow algorithm is illustrated in Figure 8. The input is the network structure (G), where each node refers to a station with its maximum capacity (the number of vehicles), and the link between nodes refers to a distance between them. First, the algorithm performs the "maximum-flow" to find a feasible flow through a capacity that obtains the maximum flow rate (k). Second, the algorithm performs the "shortest-path" to find the path (t) wherein the cost summation is the minimum. Then, the capacity of each node is subtracted by the minimum capacity (m) of the shortest path (t), while the cumulative flow (f) is added. The algorithm repeats until the cumulative flow is equal to the maximum flow rate (k). Finally, we obtain the list path of minimum cost. In this study, we used the Minimum Cost Maximum Flow algorithm implemented by Aric et al. [44].

Our relocation optimization can be represented by a network, as depicted in Figure 9. In our scenario, this optimization algorithm is customized to maximize the number of relocation vehicles, while minimizing the total distance to move those vehicles; therefore, the operation plan is efficient with less staff's efforts. The source node (S) is the node from which the flow starts, while the target node (T) is the node where all flows are terminated. In this case, the flow represents the total amount of vehicles in the service. As can be seen, there are two groups of nodes between the source and the target nodes, which are departure stations and destination stations. Every departure station node has edges outbound to every destination station node. Each edge represents the path from the departure station to the destination station. These edges have costs corresponding to the distance between the departure station and the destination station, and have capacities corresponding to the available vehicles at each departure station. Meanwhile, the edges connecting the source node to departure stations have zero cost and capacity, which is equal to the number of vehicles available at the current time. Similarly, the edges from the destination stations to the target node have zero cost and capacity, which is equal to the number of vehicles available at the prediction time.

By running the algorithm on this network, we can obtain the optimized relocation trips for operation staff. The flows from each departure station to each destination station indicate the vehicle relocation trips that have to be done between these two stations. By optimizing the distance that the vehicle needs to travel, we can ensure the minimum energy and time consumption for our vehicle relocation tasks.

4.4. Relocation Management and Notification System. In order to completely deploy our vehicle relocation system in the real-world use case, it is necessary to implement an application that can automatically notify the operation staff whenever a relocation trip is needed. Figure 10 shows the monitoring dashboard and Line notification that we implemented for our system. As shown in the monitoring dashboard, the vehicle relocation trips issued by our model are classified into three types: active trips, operating trips, and completed trips. The active trips are the trips issued by the model but not yet accepted by any staff. The operating trips are the trips accepted by the operation staff and still in the relocation process. The completed trips are the trips already done by the staff. The staff leader usually monitors the dashboard and also has an authority to add or remove the trips manually. This is allowed in order to increase the flexibility of the relocation management. Once a new vehicle relocation trip is issued as an active trip, all of the operating staff will be notified by Line application, as shown in Figure 10(b). Then, the staff who are available can press "Accept" to accept this new trip. When the staff finish moving the vehicle to the destination station, the staff will press "Complete" to inform the system that this trip has been completely done.

For more clarification, Figure 11 depicts the sequence diagram of our vehicle relocation process. When the demand forecasting model pushes a new optimized vehicle relocation trip to the system, the system will notify all operation staff via Line application. Only the first operation staff who accepts the trip will work for this relocation trip. Other operation staff who press accept later will receive failed responses and the details showing the name of the operation staff who is working for this relocation trip. After the operation staff finishes working for the relocation trip, the operation staff has to report to the system using the Line application.

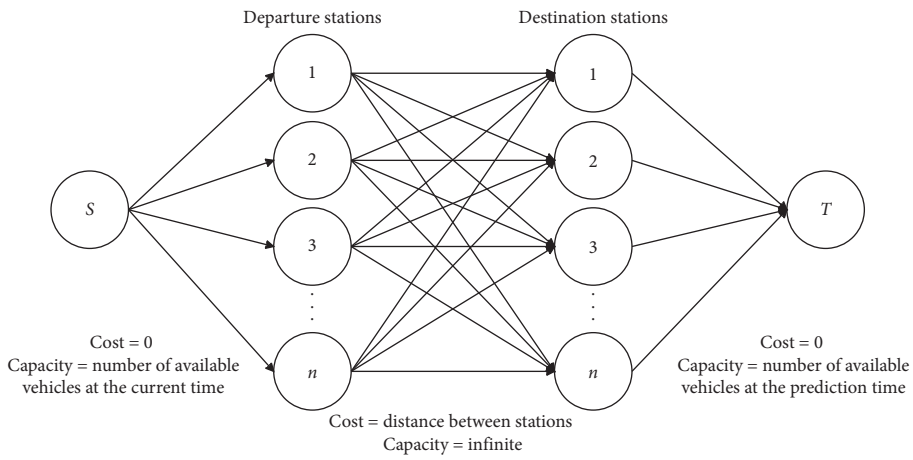
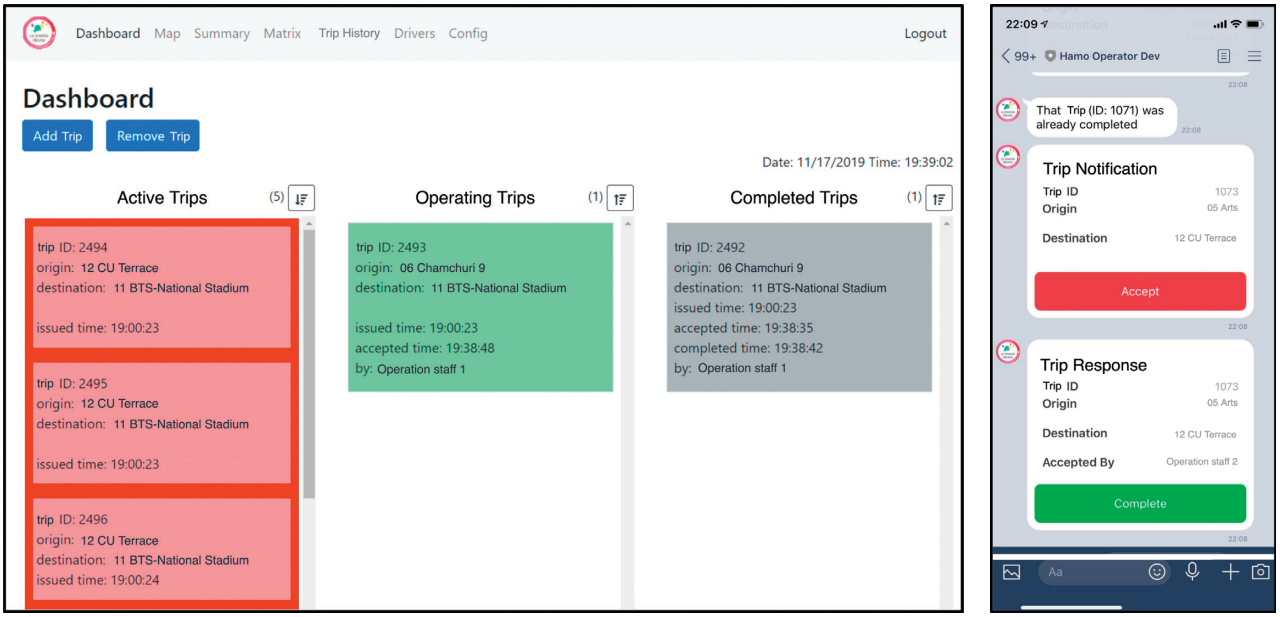


Figure 9: The network when the Minimum Cost Maximum Flow algorithm is applied to our relocation optimization.



(a) (b)

Figure 10: (a) Monitoring dashboard and (b) line notification.

Figure 12 illustrates our deployment diagram. Our computation server runs three processes including the shared database, the model service, and the backend of the core service. Another server, named CU-HAMO.COM, serves the administrative dashboard website, which is the frontend of the core service. We use Cloudflare's service to certify our domain name "cu-hamo.com" and to use the HTTPS protocol. The HTTPS protocol is required by Line service when communicating with Line application servers.

5. Performance Evaluation

In this section, there are three experiments reported. First, we aim to show that our data preprocessing can really remove fake demands compared to experienced staff. Second, many variations of the relocation models have been compared. Finally, the winner model was compared to experienced staff in the real environment showing that it can really improve in all measures.

5.1. Experimental Setup. In this study, the model is trained based on a history of usage logs. To make the experiment more realistic, a scenario based on the real data was simulated during September to November of 2019 in order to examine the performance of our system. The simulation environment is setup as follows:

(i) The number of vehicles and the station are 30 and 22, respectively

(ii) The service time in one day starts from 7 a.m. and ends at 7 p.m

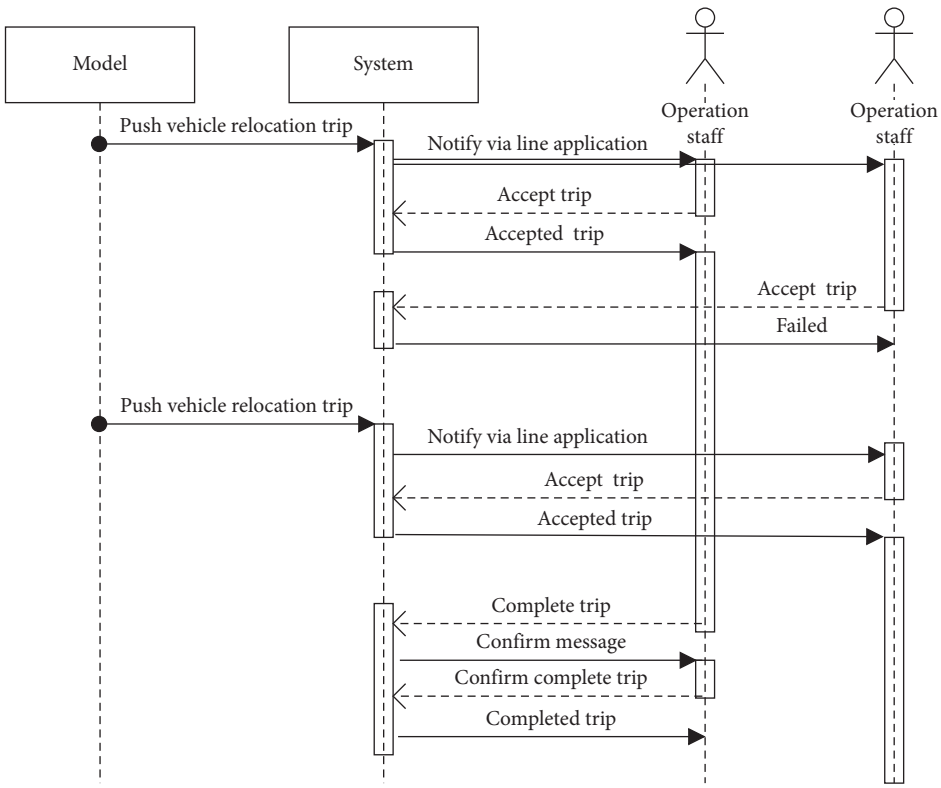


Figure 11: Sequence diagram of our vehicle relocation process.

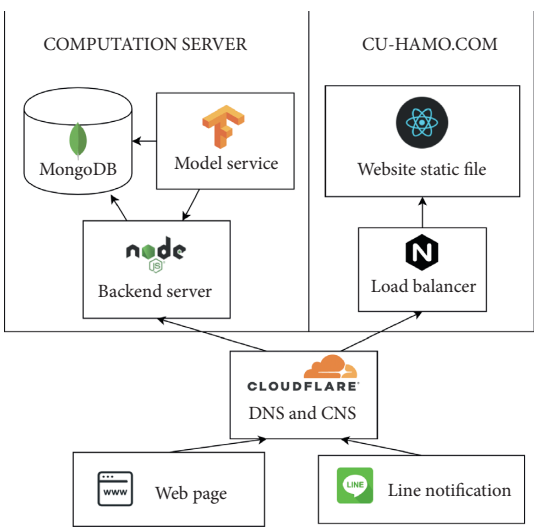


Figure 12: Deployment diagram.

(iii) Each usage in the service history is processed chronologically

(iv) Each user books a vehicle for no longer than 15 minutes

(v) The system runs every hour, and its output will be used to relocate vehicles

In this setup, there are two main performance measures: the opportunity loss (booking search failures) and the number of relocation trips (staff efforts). The first metric was used to measure how opportunity loss reduces, which is our main goal, while the second metric was selected to measure staff's workload in our system.

To make the results more reliable, the cross-validation in time series is employed by using the rolling basis method since there is an order of sequences in the vehicle usage. Thus, no future data are allowed to be used in the training data. As shown in Figure 13, the whole data form the usage log in 2019. There is a threefold cross-validation in time series. In the first iteration, the data in September were treated as test data for the model trained by the data from January until August. After that (in the second iteration), the data in September were integrated into the training set and the retrained model again, and then tested on the data in October. Finally (the last iteration), this procedure was also applied to the data in November as the test set.

5.2. Results of Data Preprocessing on Fake Demand Usages. Since the model is trained based on vehicle usage logs, it is crucial to remove fake demands like search fails and multiple search logs from the same user. In this section, we compared unprocessed logs in September and October of 2019 to cleaned usage logs by (i) manual data preprocessing by experienced staff and (ii) our automatic data preprocessing. Table 1 shows that there are a lot of fake demands caused by no vehicle available and no parking lots available with more than 50% false demands compared to the cleaned data by experts. Also, our data preprocessing can really reduce those fake demands by more than 50% on average, which is comparable to how experts clean the data. The elimination of

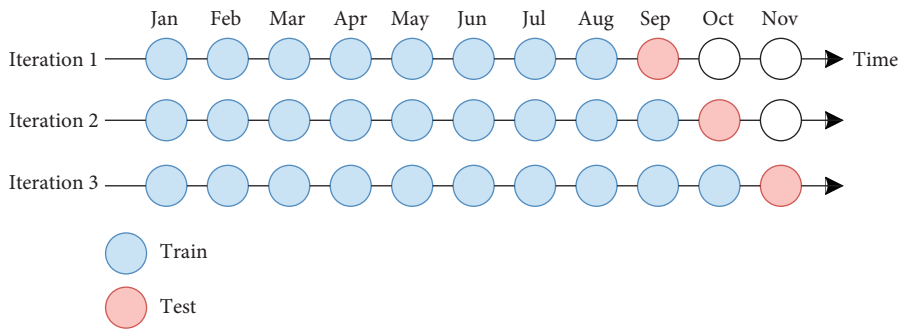


Figure 13: The threefold cross validation in time series of the vehicle usage in 2019, where the data in September, October, and November are treated as test sets.

Table 1: A comparison of usage logs among (1) unprocessed data, (2) cleaned data by experts, and (3) cleaned data by our algorithm.

Experiment	Unprocessed	Cleaned by experts	Cleaned by system
September 2019			
Failed usage (no vehicle case)	592	150 (−442)	297 (−295)
Failed usage (no station case)	545	174 (−371)	263 (−282)
Total failed usage	1,137	324 (−813)	560 (−577)
October 2019			
Failed usage (no vehicle case)	715	130 (−585)	322 (−393)
Failed usage (no station case)	492	149 (−343)	227 (−265)
Total failed usage	1,207	279 (−928)	549 (−658)

The number in the parentheses refers to the number of fake usages that can be reduced.

staff can benefit in many aspects: reduce human efforts, data cleansing consistency, and more accurate model due to better quality of the training data.

In conclusion, data preprocessing is really important and required in any real deployment scenario. Without removing fake demands, the forecasting results can be overestimated. Also, other relocation systems may encounter the same issue as in Hamo's use case, so that they can directly adapt our data cleansing strategy to their systems.

5.3. Results of Relocation Algorithms. This experiment aims to compare various relocation algorithms based on two measures. The first measure is the Root Mean Square Error (RMSE), which is the evaluation of model precision. The second measure is the opportunity loss, which is the amount of search failures resulting in less services. The third measure is the number of relocation trips, which is the amount of efforts by staff to relocate vehicles in order to have vehicles available in the stations with more demands.

There are four methods in the comparison: three models are based on different forecasting techniques (BiGRU, GRU, and LSTM) and the last one is a relocation by experts. The experiment was conducted on the real usage data of three months (September, October, and November). Note that the results of November are based on data after we deployed our system. Table 2 shows the model precision of three models through RMSE. The results show that an averaging RMSE of BiGRU is slightly higher than that of LSTM and GRU. Figure 14 shows the result of each month separately, while Figure 15 shows the overall result of three months. In Figure 14(a), the results show that BiGRU is the winner in terms of the reduction of opportunity losses unanimously in all three months. The number of total reduction losses is 6,778 or 4.10% in comparison to the manual relocation by experts (7,068 losses). In Figure 14(b), the results show that LSTM is the winner with total relocation trips of 7,102 rounds; however, BiGRU also provides a comparable result with total relocation trips of 7,142 rounds or 5.43% reduction of relocation efforts in comparison to the manual relocation by experts (7,552 rounds). It can be concluded that BiGRU-based relocation is the winner in both opportunity losses and relocation costs. In addition, all variations of our automatic relocation algorithms unanimously outperform a process managed by experts in both opportunity losses and less relocation efforts.

Table 2: A comparison of RMSE among (1) LSTM, (2) GRU, and (3) BiGRU.

Model	September	October	November	Average
LSTM	0.1677	0.1762	0.1680	0.1706
GRU	0.1676	0.1761	0.1678	0.1705
BiGRU	0.1676	0.1751	0.1679	**0.1702**

5.4. Results of the Real Deployment. Apart from a contribution in terms of algorithm advancement, we also implement a web application with automatic rule-based notification via Line to staff. Our model was trained based on the historical data from December 2017 to October 2019. Then, the deployment in the real-world scenario was conducted in November 2019. The first half of November (1st–11th) was controlled manually by experts from the

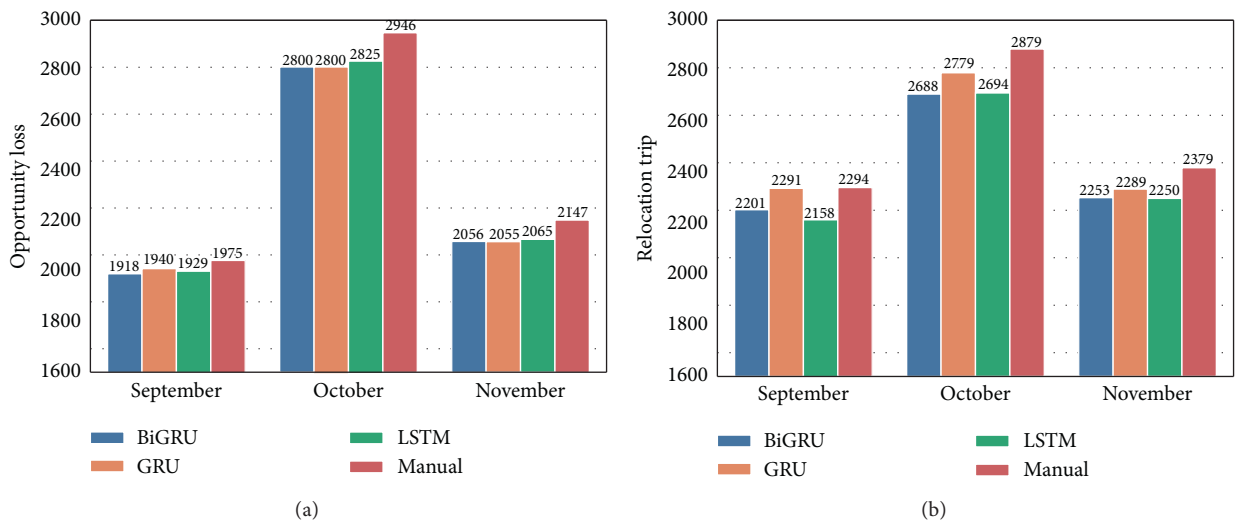


Figure 14: A comparison of four relocation algorithms: three are automatic models (BiGRU, GRU, and LSTM) and the last one is a manual model by experts in terms of (a) opportunity losses and (b) relocation trips.

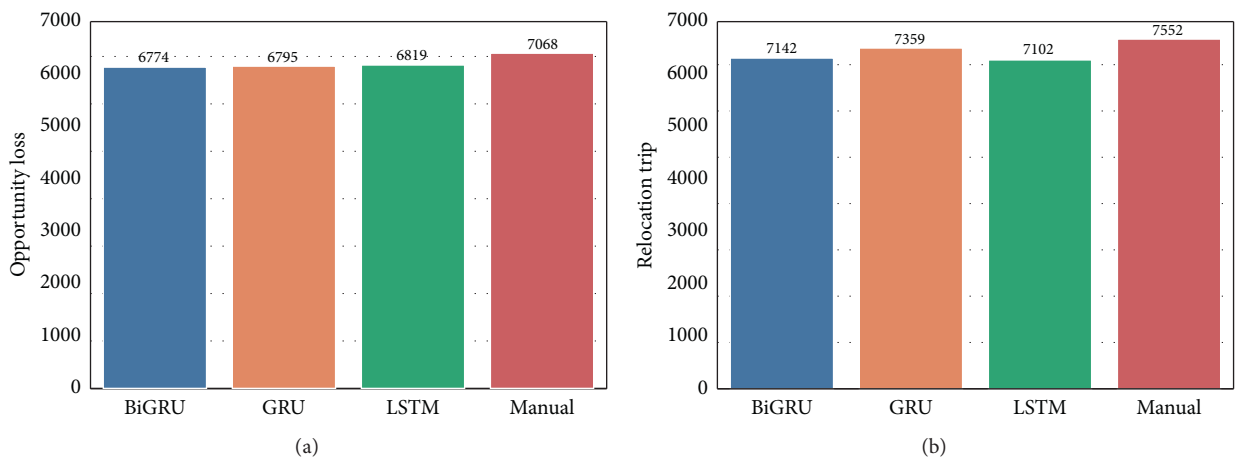


Figure 15: A comparison of four relocation algorithms on overall result of all three months in terms of (a) opportunity losses and (b) relocation trips..

operation room, while the other half (12th–26th) was operated automatically by our model (without any effort by the experts). There are three measures including opportunity losses (search failures), relocation trips (costs), and vehicle usages (higher means more services). The first two measures are compared to the last measure (vehicle usage) in order to show a ratio between losses over gains.

In Figure 16(a), the results show that there are 89 (63%) usages and 53 (37%) losses over the total demand of 142 during the relocation period by experts, and there are 86 (66%) usages and 45 (34%) losses over the total demand of 131 during the relocation period by our application. This shows that the system can provide higher usages (+3%) with lower losses (−3%) when compared with experts.

In Figure 16(b), the results demonstrate that there are 64 (42%) relocation trips (costs) over the total demand of 142 during the relocation period by experts, and there are 54 (39%) relocation trips (costs) over the total demands of 131 during the relocation period by our application. This illustrates that the system also reduces the amount of relocation efforts by 3%. Moreover, there was a report from the staff that the amount of their work can be reduced for about 17 man-hours/week based on the saving hours per week manually spent by Hamo staff on this task before having our system.

In conclusion, our application cannot only increase the number of usages, but it also reduces opportunity losses and relocation efforts. This is one of the biggest contributions of our paper since this result is based on the real use case, while most prior works were evaluated in a simulated environment. Although our algorithms are designed specifically for the Hamo car-sharing system, they can be applied to other relocation systems with large scale users, vehicles, and stations. Demand forecasting and relocation optimization are the common modules in every relocation system. Thus, those systems with large-scale setups can benefit from our proposed methods.

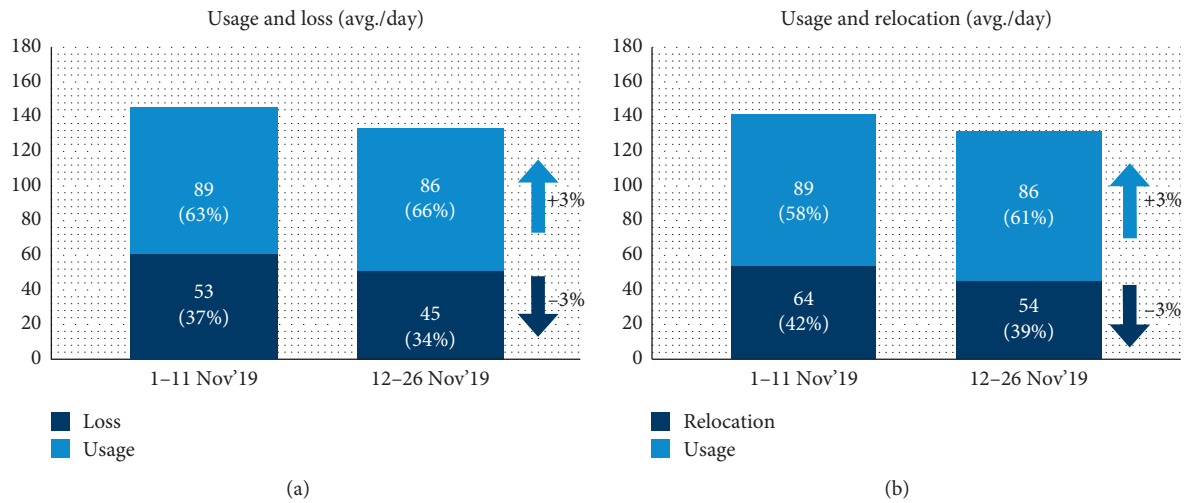


Figure 16: A comparison in November of 2019 between the relocation by experts during the first half and by our application during the second half: (a) usages over opportunity losses (failed usages) and (b) usages over relocation trips (staff costs).

From our experience in this real use case, it is concluded that there should be a minimum number of vehicles in each station in order to guarantee the service in all stations. If there is no limit on the number of relocated vehicles, it can cause a shortage of available vehicles in some stations. To address this concern, we have already implemented the condition of a minimum number of vehicles at each station in our system.

6. Conclusion

In this paper, we aim to propose an automatic vehicle relocation platform for a car-sharing scenario. The contributions are shown in terms of the deep-learning-based algorithm and the real production system. For the algorithm, the data preparation is proposed to remove fake demands, then the recurrent neural networks are applied to forecast the demands, and finally the Minimum Cost Maximum Flow algorithm (Min-Cost Max-Flow) is employed to optimize the relocation trips. The experiment was conducted in a real scenario, entitled "CU Toyota Ha:mo," at Chulalongkorn University, Thailand in 2019. The results show that our data preparation can reduce more than 50% of fake demands, which is comparable to manual processing by human experts. Also, the best relocation algorithm is Bidirectional GRU along with Min-Cost Max-Flow; it outperforms the operation by human experts with lower losses for 4.10% and less number of relocation trips for 5.43%. The real web application has fully replaced a system operated by humans. By comparison to the previous human-operated system, the result is so remarkable that it increases usage by 3% (from 86% to 89%), while reducing both opportunity losses and staff efforts by 3% (from 37% to 34%) and 3% (from 42% to 39%). There was a report from the staff that the amount of their work can be reduced by about 17 man-hours/week.

As in the case of our relocation solution for the Hamo case study, the whole system can be integrated into other car-sharing systems where the relocation is performed by the operator. Furthermore, each of our proposed modules can be applied separately to solve the issue and increase the performance of other systems. First, the data preprocessing module can be used to deal with the fake demand issue. Second, the vehicle demand forecasting module can be helpful for organizing our supplies in advance. Finally, the relocation optimization module can maximize the profit while minimizing the operation costs.

In the future, we plan to extend our solution into larger areas, such as districts or cities. If the car-sharing system covers an entire area of the city, it may be infeasible to rebalance the vehicles only by the staff. For such large areas, there should be an alternative strategy, such as user cooperation-based relocations.

Data Availability

Access to data is restricted due to commercial confidentiality.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This research was supported by CU Toyota:Hamo. The authors would also like to thank Attawit Chaiyaroj and Amornpong Trakarnkulphun, Department of Computer Engineering, Chulalongkorn University, for their valuable comments. This research was funded under the project "CU Toyota:Hamo" in 2019.

References

[1] M. J. Beckmann, "Traffic congestion and what to do about it," *Transportmetrica B: Transport Dynamics*, vol. 1, no. 1, pp. 103–109, 2013.

[2] F. Zhou, Z. Zheng, J. Whitehead, R. Perrons, L. Page, and S. Washington, "Projected prevalence of car-sharing in four Asian-Pacific countries in 2030: what the experts think," *Transportation Research Part C: Emerging Technologies*, vol. 84, pp. 158–177, 2017.

[3] S. Frost, *Future of carsharing market to 2025 - technology advancements, market consolidation and government initiatives to influence market growth over the next decade*, Vol. 8, Frost and Sullivan, San Antonio, TX, USA, 2016.

[4] S. A. Shaheen, N. D. Chan, and H. Micheaux, "One-way carsharing's evolution and operator perspectives from the Americas," *Transportation*, vol. 42, no. 3, pp. 519–536, 2015.

[5] S. Illgen and M. Höck, "Literature review of the vehicle relocation problem in one-way car sharing networks," *Transportation Research Part B: Methodological*, vol. 120, pp. 193–204, 2019.

[6] A. Di Febbraro, N. Sacco, and M. Saeednia, "One-way carsharing: solving the relocation problem," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2319, no. 12, pp. 113–120, 2012.

[7] M. Barth, M. Todd, and L. Xue, "User-based vehicle relocation techniques for multiple-station shared-use vehicle systems," in *Proceedings of the Transportation Research Board 80th Annual Meeting*, Washington, DC, USA, January 2004.

[8] E. Cepolina, A. Farina, and A Pratelli, "Car-sharing relocation strategies: a state of the art," *WIT Transactions on State-of-the-Art in Science and Engineering*, vol. 109–120, 2014.

[9] E. M. Cepolina and A. Farina, "A new shared vehicle system for urban areas," *Transportation Research Part C: Emerging Technologies*, vol. 21, no. 1, pp. 230–243, 2012.

[10] A. Angelopoulos, D. Gavalas, C. Konstantopoulos, D. Kypriadis, and G. Pantziou, "Incentivized vehicle relocation in vehicle sharing systems," *Transportation Research Part C: Emerging Technologies*, vol. 97, pp. 175–193, 2018.

[11] M. V. Behzad, M. Iraj, M. Nezam, and K. Esmaile, "Balancing public bicycle sharing system using inventory critical levels in queuing network," *Computers & Industrial Engineering*, vol. 141, p. 106277, 2020.

[12] W. Herbawi, M. Knoll, M. Kaiser, and W. Gruel, "An evolutionary algorithm for the vehicle relocation problem in free floating car sharing," in *Proceedings of the 2016 IEEE Congress on Evolutionary Computation (CEC)*, pp. 2873–2879, Vancouver, BC, USA, December 2016.

[13] C. Gambella, E. Malaguti, F. Masini, and D. Vigo, "Optimizing relocation operations in electric car-sharing," *Omega*, vol. 81, pp. 234-235, 2017 12.

[14] D. Yu, Z. Li, Q. Zhong, Y. Ai, and W. Chen, "Demand management of station-based car sharing system based on deep learning forecasting," *Journal of Advanced Transportation*, vol. 2020, pp. 1–15, 2020 02.

[15] A. G. H. Kek, R. L. Cheu, Q. Meng, and C. H. Fung, "A decision support system for vehicle relocation operations in carsharing systems," *Transportation Research Part E: Logistics and Transportation Review*, vol. 45, no. 1, pp. 149–158, 2009 01.

[16] D. Zhao, X. Li, and J. Cui, "A simulation-based optimization model for infrastructure planning for electric autonomous vehicle sharing," *Computer-Aided Civil and Infrastructure Engineering*, 2019.

[17] X. Lu, Q. Zhang, Z. Peng, Z. Shao, H. Song, and W. Wang, "Charging and relocating optimization for electric vehicle carsharing: an event-based strategy improvement approach," *Energy*, vol. 207, p. 118285, 2020.

[18] R. Martin, K. Mor, B. Burak, and G. Nikolas, "Dynamic prediction-based relocation policies in one-way station-based carsharing systems with complete journey reservations," *Transportation Research Part B: Methodological*, vol. 130, pp. 82–104, 2019.

[19] W. Ning, G. Jiahui, L. Xiang, and F. Tong, "A service demand forecasting model for one-way electric car-sharing systems combining long short-term memory networks with Granger causality test," *Journal of Cleaner Production*, vol. 224, p. 118812, 2020.

[20] L. Qu, W. Li, W. Li, D. Ma, and Y. Wang, "Daily long-term traffic flow forecasting based on a deep neural network," *Expert Systems with Applications*, vol. 121, pp. 304–312, 2019.

[21] Q. Zhaowei, L. Haitao, L. Zhihui, and Z. Tao, "Short-term traffic flow forecasting method with M-B-LSTM hybrid network," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–11, 2020.

[22] D. Ma, B. Sheng, S. Jin, X. Ma, and P. Gao, "Short-term traffic flow forecasting by selecting appropriate predictions based on pattern matching," *IEEE Access*, vol. 6, pp. 75629–75638, 2018.

[23] D. Ma, X. Song, and P. Li, "Daily traffic flow forecasting through a contextual convolutional recurrent neural network modeling inter-and intra-day traffic patterns," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–10, 2020.

[24] A. Voulodimos, N. Doulamis, A. Doulamis, and E. Protopapadakis, "Deep learning for computer vision: a brief review," *Computational Intelligence and Neuroscience*, vol. 2018, pp. 1–13, Article ID 7068349, 2018.

[25] J. Bai and Y. Chen, "A deep neural network based on classification of traffic volume for short-term forecasting," *Mathematical Problems in Engineering*, vol. 2019, pp. 1–10, Article ID 6318094, 2019.

[26] G. Lingani, D. B. Rawat, and M. Garuba, "Smart traffic management system using deep learning for smart city applications," in *Proceedings of the 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC)*, Las Vegas, NV, USA, March 2019.

[27] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," in *Proceedings of the NIPS 2014 Workshop on Deep Learning*, Montreal, Canada, December. 2014.

[28] S. Yu, J. Wang, J. Liu, R. Sun, S. Kuang, and L. Sun, "Rapid prediction of respiratory motion based on bidirectional gated recurrent unit network," *IEEE Access*, vol. 8, pp. 49424–49435, 2020.

[29] H. M. Lynn, S. B. Pan, and P. Kim, "A deep bidirectional GRU network model for biometric electrocardiogram classification based on recurrent neural networks," *IEEE Access*, vol. 7, pp. 145395–145405, 2019 09.

[30] V. Vukotić, C. Raymond, and G. Gravier, *A Step beyond Local Observations with a Dialog Aware Bidirectional GRU Network for Spoken Language Understanding*, pp. 3241–3244, Interspeech, San Francisco, CA, USA, 2016.

[31] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: surpassing human-level performance on ImageNet classification," in *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 1026–1034, Santiago, CL, USA, December 2015.

[32] C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, "Activation Functions: Comparison of Trends in Practice and Research for Deep Learning," https://arxiv.org/abs/1811.03378.

[33] G. Dahl, T. Sainath, and G. Hinton, "Improving deep neural networks for LVCSR using rectified linear units and dropout,"

IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 26, pp. 8609–8613, 2013.

[34] S. Ioffe and C. Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," *Proceedings of the Machine Learning*, vol. 37, pp. 448–456, 2015.

[35] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, p. 1929, 2014 –1958.

[36] G. Petneházi, "Recurrent neural networks for time series forecasting," 2019, https://arxiv.org/abs/1901.00069.

[37] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *Proceedings of the ICML*, Haifa, Israel, June 2017.

[38] M. Hadji and D. Zeghlache, "Minimum cost maximum flow algorithm for dynamic resource allocation in clouds," in *Proceedings of the 2012 IEEE Fifth International Conference on Cloud Computing*, pp. 876–882, Honolulu, HI, USA, June 2012.

[39] D. Benda, X. Chu, S. Sun, T. Q. S. Quek, and A. Buckley, "Renewable energy sharing among base stations as a min-cost-max-flow optimization problem," *IEEE Transactions on Green Communications and Networking*, vol. 3, no. 1, pp. 67–78, 2019.

[40] D. Gulotta, D. Kane, and A. Spann, "Application of min-cost flow to airline accessibility services," *UMAP Journal*, vol. 27, pp. 367–385, 2006.

[41] P. S. Pattnayak, S. S. Patra, and J. Pradhan, "Optimizing the network flow in cloud supply chain management," *International Journal of U- and E- Service, Science and Technology*, vol. 10, no. 5, pp. 43–54, 2017.

[42] T. H. Szymanski, "Max-flow min-cost routing in a future-internet with improved QoS guarantees," *IEEE Transactions on Communications*, vol. 61, no. 4, pp. 1485–1497, 2013.

[43] C. Ababei and R. Kavasseri, "Speeding-up network reconfiguration by minimum cost maximum flow based branch exchanges," in *Proceedings of the IEEE PES Transmission and Distribution Conference and Exposition: Smart Solutions for a Changing World*, vol. 1–7, New Orleans, LA, USA, April 2010.

[44] A. A. Hagberg, D. A. Schult, and P. J. Swart, "Exploring network structure, dynamics, and function using NetworkX," in *Proceedings of the 7th Python in Science Conference (SciPy2008)*, pp. 11–15, Pasadena, CA, USA, August 2008.

Hindawi
Journal of Advanced Transportation
Volume 2021, Article ID 5943567, 15 pages
https://doi.org/10.1155/2021/5943567

Research Article

A Data-Driven Scalable Method for Profiling and Dynamic Analysis of Shared Mobility Solutions

Bogdan Toader,[1] **Assaad Moawad**,[2] **Thomas Hartmann**,[2] **and Francesco Viti**[1]

[1]*Mobilab Transport Research Group, University of Luxembourg, L-4364, Esch-sur-Alzette, Luxembourg*
[2]*DataThings S.A.R.L., L-1811 Luxembourg, Luxembourg*

Correspondence should be addressed to Francesco Viti; francesco.viti@uni.lu

Received 17 January 2020; Revised 26 August 2020; Accepted 31 December 2020; Published 19 January 2021

Academic Editor: Zhuping Zhou

Copyright © 2021 Bogdan Toader 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.

The advent of Internet of Things will revolutionise the sharing mobility by enabling high connectivity between passengers and means of transport. This generates enormous quantity of data which can reveal valuable knowledge and help understand complex travel behaviour. At the same time, it challenges analytics platforms to discover knowledge from data in motion (i.e., the analytics occur in real time as the event happens), extract travel habits, and provide reliable and faster sharing mobility services in dynamic contexts. In this paper, a scalable method for dynamic profiling is introduced, which allows the extraction of users' travel behaviour and valuable knowledge about visited locations, using only geolocation data collected from mobile devices. The methodology makes use of a compact representation of time-evolving graphs that can be used to analyse complex data in motion. In particular, we demonstrate that using a combination of state-of-the-art technologies from data science domain coupled with methodologies from the transportation domain, it is possible to implement, with the minimum of resources, the next generation of autonomous sharing mobility services (i.e., long-term and on-demand parking sharing and combinations of car sharing and ride sharing) and extract from raw data, without any user input and in near real time, valuable knowledge (i.e., location labelling and activity classification).

1. Introduction

The transportation industry is on the edge of unprecedented change. A survey shows that 81% of respondents believe Internet of Things (IoT) will revolutionise the transport sector [1] and the transportation industry spending is expected to surpass the $1 trillion mark in 2022 [2]. The advent of IoT in the last decade enabled the connectivity of people, goods, means of transportation, and the entire transportation infrastructure. The result is an unparalleled amount of data, delivered at revolutionary speed and in continuous expansion [3]. At the same time, the transportation industry remains the sector with the fastest-growing concerns in terms of emissions and one solution is provided by the shared mobility services [4].

The large-scale adoption of IoT caused similar issues in different industries and domains, e.g., electrical smart grid domain [5, 6], where the objective is to analyse data collected in a cyber-physical system in near real-time and to ultimately support decision-making processes based on the results of this analysis [7]. Similarly, ITS drives the implementation of data science techniques for real-time analytics of data in the transportation domain. This means that new methodologies must be able to handle not only data at rest applications (i.e., data that have been collected and are then analysed after the event occurs) but also data in motion (i.e., the analytics occur in real time as the event happens). Data in motion gathered from advanced sensing (such as built-in sensors from mobile devices) and other types of traffic information (such as traffic metering) can be combined to better analyse in near real time users' travel behaviour and derive their mobility needs.

The literature shows the necessity of shifting from classical ITS to smart social mobility services (SSMS) [8], which consists of an integrated and cooperative approach to sense users' individual needs and interactions and offers user-centred mobility services. Following the recommendations from [4], additional research must be done as the ITS

must be prepared for analysing data in motion in real time, learning users' behaviour and performing fast searches in large datasets, which could instead contribute to a more integrated, fast, and flexible method for implementing collaborative mobility services at different levels and for different needs.

In this study, we propose a method for dynamic and near real-time profiling of travel behaviour in time and space, using data in motion. In our case, profiling means the extraction of user habits for visiting a specific location. The real-time aspect is a mandatory requirement for shared collaborative systems (e.g., car sharing and parking sharing) where a large number of people and goods are moving at high speeds and solutions to combine them in an efficient way must be provided in real time (e.g., peer-to-peer ride sharing). In this sense, the proposed methodology makes use of multidimensional profiling techniques described in Section 3 to automatically build the profile as soon as the data become available and proposes efficient techniques to store the results in a temporal index for fast access.

The remainder of this paper is organised as follows. First, Section 2 presents a background of this study making the link with previous work and the proposed methodology in Section 3. The evaluation of the proposed method is presented in Section 4, along with practical usage examples in Section 5 and future work in Section 6. We conclude the paper with a discussion of future directions in Section 7.

2. Literature and Background

Travel behaviour is an interdisciplinary problem, which combines (*a*) specific methodologies for travel behaviour analysis and user's habits profiling and (*b*) data science methods for efficient computation. In the following sections, we present the general background and literature from each domain and the links with the previous work. This will help to define the base terminology, notions, and scientific background necessary for understanding the contribution of this study.

2.1. Travel Behaviour Analytics. In general, data-driven travel behavioural profiling refers to the process of constructing and applying various learning techniques, using the mobility data generated by users and other entities (e.g., sensors from different means of transportation and traffic counters). In the transportation domain, profiling is a method used with different objectives. Driver behaviour has been profiled using the advanced motion sensors from cars and smartphones to detect driving events and to classify drivers in specific categories. Profiling methods are used in fleet management, insurance policies, fuel consumption optimisation, or gas emission reduction [9–12], as well as in route choice in multimodal networks in order to consider the individual preferences in route recommendation systems [13] and in Internet-oriented user centric intelligent transportation systems [14].

More recently, attention has been focused on understanding the human mobility using the profiling of users [15]. Data generated by static and mobile sensors implemented in different transportation systems and smartphones allow to understand the patterns and citizens' habits at large scale [16]. This is used for semantic information extraction on the mobility of users but also for the study of spatiotemporal variation in travel regulations through transit data [17]. Mobility user profiles can offer valuable information for understanding the disaggregate and aggregate spatiotemporal activity patterns [18], but the proposed methods are static and do not take into consideration data in motion and the performance has not been tested with large datasets.

Several challenges have been identified in order to effectively profile user behaviour in smart mobility systems, including the learning issues for missing values, data cleansing, dimension reduction, sparse learning, and heterogeneous learning [19]. Massive amounts of raw data collected by nomadic devices (e.g., smartphones) must be cleaned, aggregated, and then processed using state-of-the-art methods and algorithms. This is the case of the shared mobility services. Previous research focused on the investigation of ride sharing opportunities. Bicocchi and Mamei [20] showed that through mobility data analysis, efficient solutions for extraction of suitable information from mobility traces can be used to identify ride sharing opportunities. The literature shows that there is a need for optimisation of these systems, which need to solve different problems related to the required features and characteristics, e.g., the dynamic character, automated matching, and cost sharing [21]. A suggested solution comes from a good understanding of users' behaviour and preferences, which is an essential feature when designing dynamic shared mobility systems.

In order to make use of collected data for large-scale mobility sharing services, users' travel behaviour and preferences must be extracted. The very first step in this process is the extraction of the duration and location of activities from raw data. A detailed review and comparison of the methodologies from literature is presented in [22]. However, all mentioned methodologies suffer from limitations when applied to dynamic and live profiling on large datasets. In practice, recommendation systems require to profile users in an environment with continuous data generated by dynamic movements of users and means of transport. They need to extract knowledge that can contribute to the mobility services to understand the human travel behaviour and automatically recommend suitable sharing services for each individual. Moreover, the analytics must be done at different levels of aggregation and resolutions, with dynamic precision and scaling, e.g., ride sharing requires a higher accuracy than the classification of secondary activities (e.g., shopping, gym, or restaurant).

In a next step, using the detected locations from the previous step, methodologies must be implemented in order to learn user mobility patterns and to perform the knowledge discovery from raw data. An example of knowledge discovery from literature is the trip purpose identification from GPS tracks [23]. They identified two main groups of trip purpose imputation routines in the literature: rule-based systems based on the position of the activity, timing, and

geographic information system (GIS) data and machine learning approaches which focus more on the activity and less on position. Montini et al. [23] used random forests [24], a machine learning algorithm that has been successfully applied in different transport-related classification problems. Data from GPS and accelerometer sensors were used as input, and the respondents were asked to correct an automatically generated travel diary that was used to extract specific features for semantic interpretation of the said data.

A similar application used in the current research is the identification/classification of each activity/visited location (e.g., home and work). The proposed profiling methodology uses only the GPS data, specific data science techniques for indexing, clustering, and querying, and as the training data, a set of known location visit patterns for each location type. The key novelty of our approach is that our methodology is able to capture detailed and complex visit patterns of users and locations through the profiling layer, which can be used in a multitude of applications simultaneously. Some usage examples are explained and evaluated in the rest of the paper (e.g., parking sharing, ride sharing, location type, and activity classification).

2.2. Data Science Methods for Efficient Computation. One of the most complex operations in big data systems is the study of the relationships between a group of entities interacting with each other in a given space. This is also the case of smart mobility systems (e.g., ride sharing), where a high number of entities represented by people, cars, and locations interact with each other. The problem becomes even more difficult when those entities interact in multidimensional spaces, represented by the properties of the above entities in motion (e.g., day, hour, geolocation, and so on). In physics, this type of problem is defined as the n-body problem [25], which is one of the most challenging topics in high-performance scientific computing, addressed by the Barnes–Hut simulation [26].

In order to reduce the costly $O(n^2)$ computation time complexity for calculating the distance between each entity, alternative optimisations have been developed in form of tree algorithms. Binary search trees are used to efficiently search and sort as by traversing the tree from root to leaf means that each comparison allows the operations to skip about half of the tree. This results in a complexity of $O(\log n)$. In a quadtree [27], each internal node has exactly four children and is most often used to partition a two-dimensional space by recursively subdividing it into four quadrants or regions. Similarly, octrees [28] are the three-dimensional analog of quadtrees. For spatial representations, each node in an octree subdivides the space it represents into eight octants.

In order to profile on more dimensions, trees with unlimited number of spaces must be used. K-dimensional trees (K-d trees) [29] are a special case of binary space partitioning trees for organising points in a k-dimensional space. Even if k-d trees are a solution for more than three dimensions, this is not suitable for efficiently finding the nearest solution in high-dimensional spaces (as if the dimensionality is k, the number of points in the data should be $N = 2^k$). Using the k-d tree with high-dimensional data, the efficiency is no better than an exhaustive search [30].

Recent studies demonstrate that ND-tree [31] data structures are efficient in high-dimensional spaces. For example, in the nearest neighbour search problem, ND-tree algorithms are effective in improving query performance in both uniform and nonuniform datasets [32] but also in the study of similarity searches in multidimensional nonordered discrete data spaces [31], a feature that is used in this study.

3. Methodology

3.1. Multidimensional Profiling in Previous Work. Multidimensional and dynamic profiling requires technologies which allow the fast processing, indexing, and querying of big datasets. In the remainder of this section, we present the data modelling framework which incorporates graphs and time series in multidimensional data models, along with the major technological implementation challenges.

The GreyCat [33] framework, formerly known as KMF [34], presented in a previous work [35], is a solution for analysing complex data in motion at scale with temporal graphs [5]. There are a number of required features (e.g., modelling with graphs, temporal aspects, and what-if analysis exploring different alternatives) presented in [35].

Another feature that is important in big data systems is the ability to lazy load nodes, meaning to load into main memory only the necessary data that need to be processed rather than to load and query each time the entire dataset. Naturally, many analytic tasks are processing only parts of the dataset. This also counts for the case study of this paper. Therefore, we suggest to load data, i.e., the nodes of our data graph only on-demand, while the graph is traversed. As an example, even if high accuracy datasets are available at an order of a few meters, if the application needs to profile to a maximum of, e.g., one kilometer accuracy, there is no need to load and process the data at a higher resolution. This will save both resources and time.

If in the previous work the proposed framework is used for the first time in the transportation domain to find possible groups of users that can use a ride sharing system using data at rest, the current work proposes a more user centric approach that can handle data in motion at scale for multiple applications (e.g., parking sharing, location classification, and nonrecurrent trips profiling).

3.2. Overview of Data-Driven Scalable Method. In this paper, we present a new way of profiling multidimensional and temporal data, specifically designed to deal with large quantities of data—in live and with different physical constraints, such as limited memory or processing power (as is the case of nomadic devices like smartphones).

Our methodology is generic in the sense that it can use any specific profiling algorithm that uses a tree-like structure to divide a parent space into two or more children subspaces. For instance: binary trees, quadtrees, octrees, and K-d trees

are all easily implementable and can be integrated in our methodology.

The proposed architecture shown in Figure 1 has three independent layers. The base layer is the lowest level which represents the *raw data layer*, dealing with data management (e.g., collect and store data). The *processing layer* deals with data processing to produce well-structured and fast-to-query spatiotemporal profiles. Among different processing tasks, the profiling is our main focus in this work, which is done through other different tasks (e.g., reduce, map, and apply to spatiotemporal data trees). Finally, the highest layer is the *application layer* where any specific transportation problem can be translated in high-level profile queries. The main advantage of the proposed architecture is that the profile layer is built once and then shared across several transportation applications, hence reducing the required infrastructure and resources.

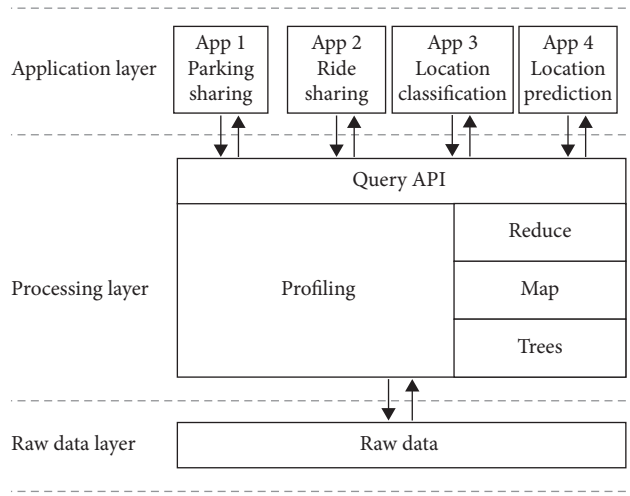


Figure 1: Architecture and abstract layers.

3.3. Terminology. As the current work includes terms from different domains, it is necessary to define the terminology that will be used throughout the entire work.

(i) *Tree*: a directed acyclic graph starting from a root node.

(ii) *Space coverage*: the N-dimensional min-max vectors that define the boundaries of the space covered by the subtree.

(iii) *The root node* is the top node in the tree. It covers the widest N-dimensional space of the tree.

(iv) *Child node*: a node directly connected to another node when moving away from the root node. Child nodes have always a smaller space coverage than their parent nodes.

(v) *Parent*: the opposite notion of a child.

(vi) *Leaf node*: a node without children.

(vii) *Siblings*: a group of nodes with the same parent.

(viii) *Degree*: the number of children of a node.

(ix) *Path*: a sequence of nodes connecting a node with a descendant.

(x) *Level*: the number of connections between the root and the node. The root node is of level 0.

(xi) *Size* of the tree is the total number of data indexed in the tree.

(xii) *Height*: the height of a tree is the maximum level reached by its nodes.

(xiii) *Resolution*: the smallest space coverage allowed for the leaf nodes. It is an N-dimensional vector representing the minimum difference allowed between the minimum and the maximum on each of the N dimensions.

(xiv) *Number of dimensions* represents the number of different features we want to profile (e.g., day of the week, time of the day, and geolocation). By default, the proposed architecture supports until 32 dimensions that are easily extensible to 64.

(xv) *Max buffer size*: the maximum size of the data stored in a node before creating a sublevel of child nodes.

(xvi) *Timeline*: a sequence of ordered timepoints.

(xvii) *Temporal resolution* represents the maximum quota in time for each profiling tree before creating another tree.

3.4. Live Profiling, Indexing, and Preprocessing. In this section, we describe specifically the preprocessing step, in which the data in motion are indexed as soon as they are received from specific sensing systems (e.g., mobile devices) or databases.

The following example deals with location data represented by points in the geographical space, represented in Figure 2. We describe how each tree structure is created based on the space partitioning and indexing of each quadrant, from the root (*Level 0*) to the leaf level (in our example Level 3). It is important to stress that the indexing and profiling methods are completely independent of the final applications that will access and use the data on the application layer.

The profiling methodology can be summarised as the following chronologically ordered steps, which are continuously performed as soon as new data are available:

(1) Start with an empty timeline.

(2) Create the first profiling tree once the data are loaded from a dataset or received through a sensing system, as can be seen in Figure 2, at *Level 0*.

(3) Once the buffer is full at the root *Level 0* (reached the max *buffer* limit of data stored at the node level, defined at design time), create child nodes of *Level 1* and redistribute the data into the corresponding subspaces. Any new data received at *Level 0* will be automatically forwarded to *Level 1* subspaces. At this step, the node at *Level 0* is transformed from a node that stores data in a *router node*, defined as a node

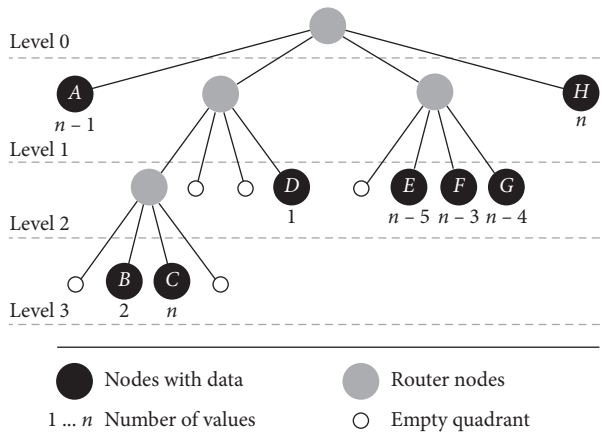


Figure 2: Profiling tree structure.

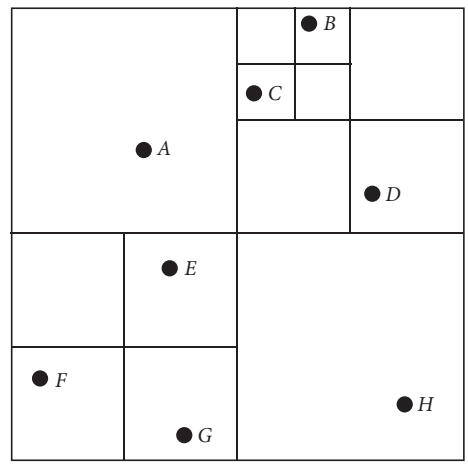


Figure 3: Profiling space partitioning for geolocation data.

that has no data but acts as a *path* that connects the node with its descendant subspaces.

(4) Each subspace has its own buffer and divides the parent dimension boundaries by two or more on each dimension. In the case of geolocation data, Figure 3 shows how the space is divided in quadrants. At *Level 1*, there are four subspaces: (1) *A*, (2) the quadrant formed by *B*, *C*, *D*, and two empty, (3) the quadrant formed by *E*, *F*, *G*, and one empty, and (4) *H*.

(5) Repeat steps 3 to 5 recursively until one of the following conditions is met:

(a) The temporal resolution of the current profiling tree has expired. As an example, if we set the maximum temporal resolution to one hour, even if the max buffer size was not reached, a new tree will be created.

(b) The tree reaches the maximum allowed size. This is a requirement to keep the process as fast as possible, since the use of too large trees makes the search computationally harder and more time-consuming.

(c) We can observe in Figure 2 that nodes (2) and (3) created at Step 4 on *Level 1* are split again into four children and transformed from nodes with data in router nodes. The same process continues also at *Level 3* until one of the above conditions is met.

(6) Once a tree is complete, it is stored and the process continues with the creation of a new tree. As can be observed in Figure 4, the new tree will have a new timepoint and the entire process from 3 to 6 will be repeated.

3.5. Querying and Postprocessing. The multidimensional and temporal features of the proposed profile offer several ways to query it in order to allow a wide range of applications. A query can specify a range in time, specific days and hours of the week, and a level of precision in the multidimensional space and can ask either for all the results within a specific

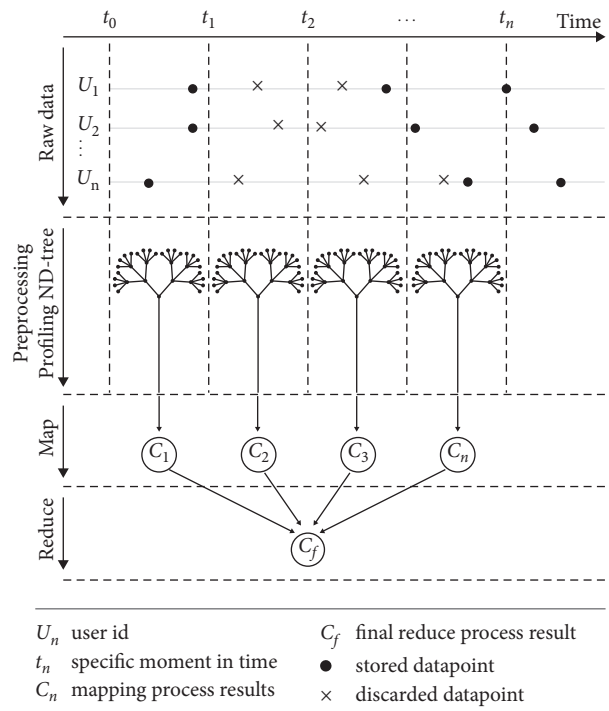


Figure 4: Data structure, query, and process flow.

range or the top *N* results from a specific complex query. The entire flow of the query process is described in Figure 4 and will be described in the remaining of this section.

In order to demonstrate the process flow, we present the example of a smart mobility shared system (e.g., car sharing, carpooling, and parking sharing) that provides the geolocation data for a number of users U_n through a smartphone application using the integrated sensing system. The system can then perform specific complex queries which must return a result in an order of milliseconds.

For example, a ride sharing system can perform a query to get all the locations that a user visited in the last two years in a specific geographical region, in specific days of the week, and specific time intervals during the day, with a specific geographic resolution. This can be useful to find possible

matches with other users that have similar profiles. The results of the queries can be stored to save the user profiles for future fast access to avoid repetitive queries or can be discarded, according to the objectives of the application. Another example can be a location classification application that extracts the user visit pattern for all the visited locations that have been visited at least s times during a specific time period and a specific geographical area. This can be useful to instantly filter trajectories points that do not represent a specific location and to classify the filtered locations based on specific location duration and time of the day.

Moreover, a query can become even more complex and can be used to show the top N locations visited, in a specific day of the week and interval of the day. This query can be used to detect the most visited locations and to quickly detect, e.g., home and work location, in order to propose a personalised itinerary end, e.g., at the user's home or to be used to match users that are compatible for parking sharing.

It is important to mention that for all of the above examples independent of any other application, the same process flow is applied. This is described below:

(1) *Data Management and Temporal Resolution*. The data are captured and processed according to the temporal resolution, in specific time intervals t_n. This temporal resolution must be set in the very beginning and represents the minimum time interval that two specific trees are indexed. For example, if from the applications domain it is known in advance that no application that uses the profiling data will need a higher time resolution than one minute or a higher geographical resolution than four meters, there is no sense to set this limit lower. A lower time interval than the minimum required will also require more resources or time to process the entire flow, providing also redundant data. There is no upper limit but just the one given by the indexing method (e.g., for the geographical space, a quadrant, no matter of the dimension in measurement units). This dynamic is important to mention as in some applications like carpooling it is possible to perform different queries with different parameters and to increase/decrease the resolution to determine, e.g., which specific routes the user uses. This information can be useful to calculate the compatibility for matching different profiles.

(2) *Efficient Storage*. There are some important aspects to mention regarding the data management and temporal resolution. First, if a user is changing the location between two consecutive t_n, t_{n+1} data points, the geolocation is stored in a node. Second, if the user is in the same location for more than two consecutive time intervals, the same information is not replicated through consecutive timepoints but is discarded, represented in Figure 4, raw data layer with x. Thus, for a visited location, only the arrival and departure timestamps are stored, which helps in cleaning the dataset of duplicate values and reduce the required storage resources. Third, if at any time t_n a query is performed and no points are found at t_n, the data from t_{n-1} will be returned, and the process begins to backtrack until a stored point is found.

(3) *Profiling Phase*. The trees are created for each time interval, following the methodology from Section 3.4.

(4) *Map Phase*. When a query is performed, the query is divided into several subqueries touching several trees and several subspaces (e.g., return the top n points from a specific time interval, specific day of the week and hour, and from a specific geospace, with a specific accuracy). The search phase can be distributed among any number of computation units and threads as needed, according to each application and specific domain requirements.

(5) *Results*. The results are then collected, and then the *reduce phase* does the synchronisation, waits for all the running threads to finish, removes the duplicates, sorts the results, and does a final postfiltering if needed.

Another important feature in the context of profile sharing with multiple applications is the ability to have a high level of parallelism. This feature brings important advantages:

(1) All queries can run in parallel: this is an important requirement when multiple applications share the same profiling layer, and multiple queries can be performed, at the same time, on the same tree indexing.

(2) Each query can be mapped to one or several profile trees according to the targeted time range of the query. The search within the targeted trees can be done as well as in parallel.

(3) Since a query can involve several subspaces within a tree, the search within these subspaces can be executed in parallel as well.

3.6. Location Visit Pattern Extraction. In addition to filtering in near real time the visited locations based on spatiotemporal complex queries, the final result of the entire process flow (i.e., starting from indexing and preprocessing, live profiling, querying, and postprocessing) is used also to extract the weekly activity pattern visit for each location visited. This is done by clustering all the visit records from the time range specified in the query parameters in a matrix with dimensions of 24 hours and the seven days of the week. Each matrix element represents the number of times a person visited a specific location, normalised by 1000, as can be seen, e.g., in Figure 5.

3.7. Location Profiling and Activity Classification. An earlier exploratory study from a previous work [36] presents the extended methodology behind the classification methodology. To summarise, the classification and labelling of each location is done by computing the Euclidean distance

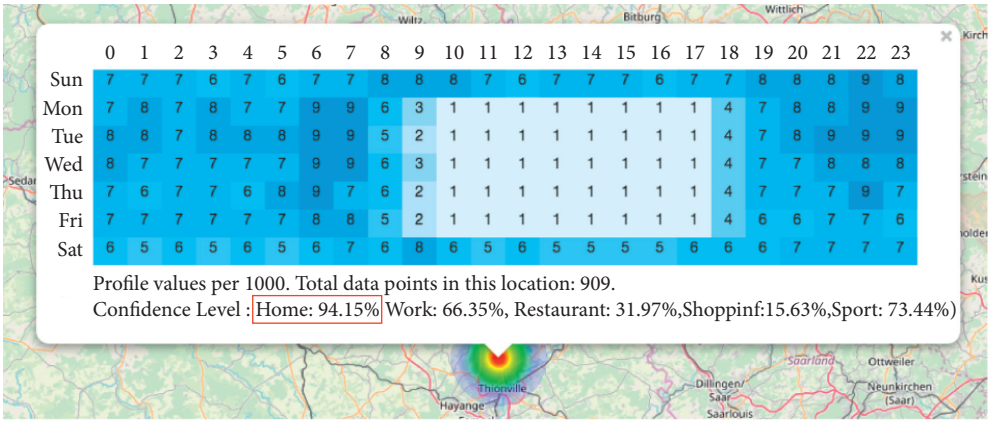


Figure 5: Example of home location classification.

(between 0 and 1) from a generated training matrix representing a known location type (e.g., home and work) and the matrix obtained for each location visit pattern. The smaller the Euclidean distance is obtained, the higher the probability that the profile of a specific location can be labelled as a type of a specific location. Figure 6 shows an example of the training matrix for a home location.

Similar matrix can be generated for other type of locations and activities (e.g., work, restaurant, shopping, and sport). Figure 5 shows an example of the classification result for a location that has 94.15% confidence level that is a home location.

Evaluation and usage examples of location profiling and classification are presented in Sections 4 and 5.

4. Evaluation

4.1. Dataset Description. In this study, two types of datasets were used in order to perform different evaluations of profiling results. Each dataset has different purposes and requirements, as follows.

First, a dataset for which we have the ground truth was used in order to evaluate the location classification and labelling accuracy based on the profile extracted for each location visited by the users. For this type of evaluation, the dataset must be accurate with less GPS errors and the results must be validated by the respondents. As the validation is done at the individual level, we did not use a huge dataset for the validation. The dataset used was based on data collected by Google Maps of 17 users from the University of Luxembourg; it is individual based and each respondent was able to easily test and validate online the results [37]. Moreover, the data come already error-filtered as the data are collected using not only the GPS sensors from the smartphones but also a fusion of data from native Android sensors like Bluetooth, Wifi, and motion sensors, which are used to validate the location even when the GPS signal is poor, e.g., inside the buildings.

Second, a larger dataset was used in order to evaluate the computational speed, performance when scaling, and accuracy. For this type of evaluation, the most important aspects are the size of the dataset, i.e., the period of time covered and the number of users. This dataset must be sufficiently large; more specifically, it should cover a relatively long time period and contain a large number of users in order to meet computational requirements within the scaling phase. The dataset used was the Geolife dataset [38]. This publicly available dataset contains around 24 million GPS points from China, collected in the Geolife project [39] from 182 users with smartphones and GPS loggers in a period of five years.

4.2. Location Classification Accuracy. The evaluation of location classification accuracy was performed by a group of 17 respondents who uploaded their GPS data exported from Google Maps to the publicly online version of the tool developed in the current research [37]. The respondents were then asked if the home and work locations were accurately detected. The results show that 100% of the respondents stated that the home and work location were correctly classified and labelled with the highest confidence level. Of course one can argue that the home and work locations are trivial to be used as an example of classification, but the scope of our evaluation is only to demonstrate that the proposed framework and methodology are able to automatically classify locations and activities performed in near real time, without any user input and only based on the GPS data. In another work [40], we extended the evaluation and methodology also for other type of locations and activities (e.g., restaurants, shopping, and sport activities) which is out of the scope of this paper.

4.3. Computational Speed. First, in order to test the computation speed of the proposed profiling method when scaling, multiple tests have been performed with different amounts of data. Figure 7 shows the results obtained when processing 12 different amounts of data, from 1 million to 24 million valid points. Moreover, as can be seen in Figure 7, the general trend line when scaling is close to logarithmic, something that confirms the complexity reduction explained in Section 2.

Second, a speed comparison has been performed between a classical linear computation in comparison with a

	Home																							
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Sun	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9
Mon	9	9	9	9	9	9	9	9	0	0	0	0	0	0	0	0	0	0	0	9	9	9	9	9
Tue	9	9	9	9	9	9	9	9	0	0	0	0	0	0	0	0	0	0	0	9	9	9	9	9
Wed	9	9	9	9	9	9	9	9	0	0	0	0	0	0	0	0	0	0	0	9	9	9	9	9
Thu	9	9	9	9	9	9	9	9	0	0	0	0	0	0	0	0	0	0	0	9	9	9	9	9
Fri	9	9	9	9	9	9	9	9	0	0	0	0	0	0	0	0	0	0	0	9	9	9	9	9
Sat	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9

Figure 6: Generated matrix for location classification training process.

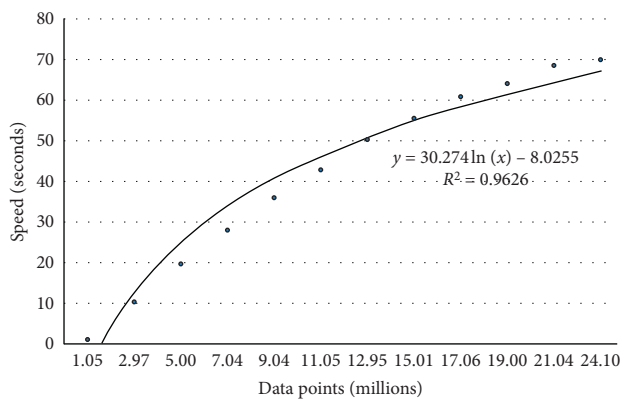


Figure 7: Computation speed when scaling.

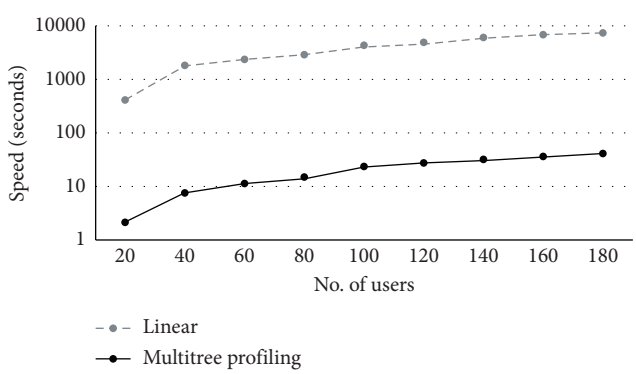


Figure 8: Linear profiling vs multitree profiling speed.

multitree profiling method, presented in Figure 8. In this experiment, 9 speed tests have been performed, with data from different number of users, from 20 to 180. The results are displayed on a log scale at y axis. The experiment clearly shows that the speed of using a multitree architecture is around 10^2 faster compared with a classic linear architecture.

Another important aspect in practical implementation of fast processing is the amount of resources needed. Most of the time, the bigger the size of the dataset is, the more the processing resources are needed in order to have the highest speed. The database research community identified graphics processing units (GPUs) as the most effective co-processors for parallel data processing [41] mainly because the dataset is processed by hundreds or thousands of small CPU nodes.

To the best of our knowledge, one of the fastest massive parallel architectures is MapD [42]. Recent experiments show that massive datasets with billions of geolocation routes can be processed and visualised in milliseconds [43]. But everything comes at a cost. Table 1 shows a comparison of speed and resources needed to process the Geolife dataset using MapD with a linear in hardware scaling method and very powerful but costly hardware, compared with a multitree profiling performed by a user PC which is a logarithmic in time or hardware method.

4.4. Accuracy. Since the profiling is done for different subspace sizes (resolution), the accuracy is highly correlated with the resolution (the smaller the resolution, the better the accuracy) and the number of levels that need to be queried to reach from the root to the leaf level (more levels means higher details, but longer time to search).

In our specific case, the accuracy depends on the size (width and height) of each resolution. Using the Geolife dataset, accuracy tests have been performed, and the results are presented in Table 2 from a minimum of 4.77×4.77 m to a maximum of 5000×5000 km.

The experiments confirm that the dataset average error is very close to the mathematical predicted error. The provided results can be used as a guideline for choosing the minimum resolution for each transportation application, based on the average and maximum error of each resolution. Thus, for any type of application, it must be assessed if the average and maximum errors are acceptable and tolerated in the domain. Different transportation applications require different precision and maximum errors. Table 3 presents an example of comparison for different applications with the precision and amount of data needed.

For some applications like ride sharing, the accuracy is important as, e.g., the meeting point of different users that can share the same car cannot have large error. A study [44] shows that on average only about 60% of the passengers will accept to walk 150 meters for transit to another bus stop and 90% of the passengers will accept to walk 50 meters. The same strict requirement has also the location classification (e.g., home, work, shops, and restaurants) or activity classification (e.g., sport and shopping) systems. In order to keep a higher quality of service and a higher rate of user retention, the maximum error must be lower than the acceptable distance that a passenger has to walk if, e.g., the suggested

Table 1: Resource and performance comparison.

Speed (ms)	Cost (EUR)	Hardware	Scaling	Method	Summary
0.5–20	40,000	8x Nvidia Tesla K80	Linear in hardware	Load all points in memory	Big data on big hardware
36–51	2000	User PC	Log in time or hardware	Multitree profiling	Big data on small hardware + smart models
6895–7579	2000	User PC	Linear on time or hardware	Linear parsing	Big data on small hardware

Table 2: Accuracy.

L_x	L_y	Unit	Worst err.	Dataset avg. err.	Math. err.
4.77	4.77	m	3.372899	1.824923	1.824993
38.2	19.1	m	21.354449	11.330335	11.330761
153	153	m	108.187337	58.537250	58.537314
1.22	0.61	km	0.682000	0.361813	0.361873
4.89	4.89	km	3.457752	1.870842	1.870902
39.1	19.5	km	21.846398	11.589896	11.590412
156	156	km	110.308657	59.691685	59.685239
1250	625	km	698.771243	370.746174	370.771200
5000	5000	km	3535.533906	1913.060314	1912.992000

Table 3: Usage examples.

	Type	Precision needed	Maximum error	Amount of data needed
1	Ride sharing	High	<50 m	Medium
2	Location/activity classification	High	<50 m	Small
3	Parking sharing	Medium	<150 m	High
4	Nonrecurrent trips	Low	<50 km	Medium

location/meeting point is not precisely in the location designated by, e.g., a recommendation system/trip planner.

For other transportation problems like parking sharing, the error can be higher as it is not unusual to park the car and walk for a decent distance until the destination, but again under the limit of the maximum user's tolerated error [45]. There are also other applications where the error can be bigger; there is no need of very detailed profile and the error can be much higher, of an order of dozens of kilometers. This is the case of nonrecurrent trip analysis, e.g., holidays or business trips where anyway the clusters and visualisation are much bigger than the above examples.

The capability to be versatile in order to handle various application requirements in the same system represents a requirement in a shared architecture. In the next section, practical usage examples of all the applications compared in Table 3 will be presented.

5. Usage Examples

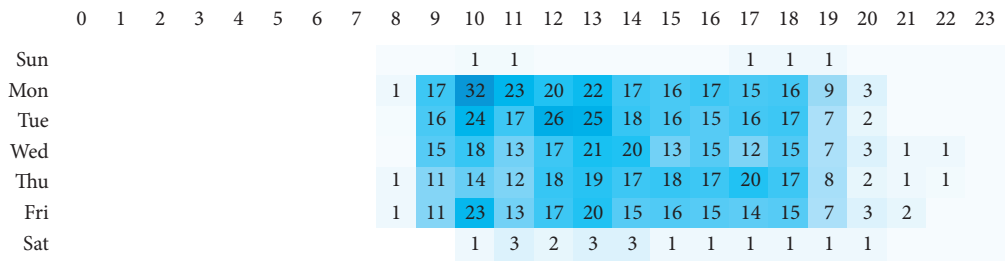
Collaborative mobility services (e.g., ride sharing) represent one of the best case studies of the methodology presented in Section 3. Different types of data collected from users' smartphones represent a new dimension that adds complexity on finding efficient solutions for combining users and transportation resources in collaborative systems. Each dimension is represented by the properties of those entities, combined with the types of queries which are performed, e.g., day, hour, location, age, sex, etc. Some of them have subdimensions, e.g., the location where users perform activities can have as subdimensions the starting and ending hour of an activity, the geographic coordinates of the location, and the radius of the geographic space that represents a location. In this study, the multidimensional profiling refers to the concept of profiling all of these dimensions, where each entity has specific properties.

5.1. Parking Sharing. A group of two or more users can share the same parking place if they use it at different times of the day. In other words, the more dissimilar the users' profiles are for the same location, the higher the compatibility for parking sharing there is. The dynamic character of the proposed profiling methodology makes possible the assessment of parking sharing both for (*a*) planned long-term parking sharing and (*b*) short-term or ad hoc parking sharing.

In order to demonstrate the usefulness of a flexible, dynamic, and fast profiling framework, we present the following case study.

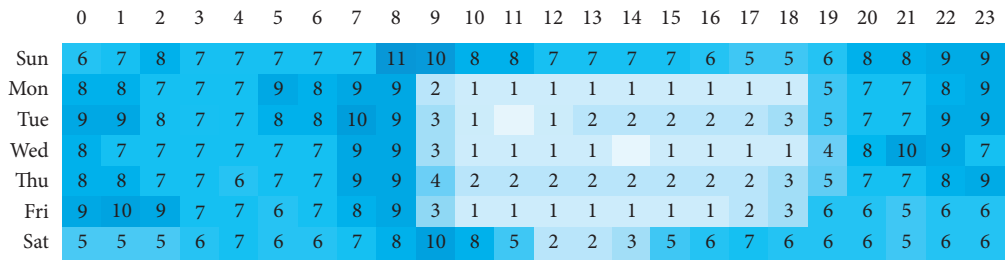
Figure 9 shows the profile of *User 1* who works in the proximity of the home of *User 2*.

User 2 is part of a peer-to-peer (P2P) parking sharing application. As we can observe, for *User 1*, the highest probability to be in the parking location is from *9 AM* to *7*



Profile values are per 1000. Total data points in this location : 59845

Figure 9: Profile of User 1 for long-term parking sharing.



Profile values are per 1000. Total data points in this location : 152440

Figure 10: Profile of User 2 for long-term parking sharing.

PM, from *Monday* to *Friday*. Similarly, Figure 10 shows the profile of *User 2*, who has the lowest probability to be in the same location when *User 1* is in the same location.

With this information, a system can match the two profiles with highest compatibility index (i.e., the highest Euclidean distance between profiles) to share the same parking location as they do not overlap. Moreover, this is done without asking the users any prior information but profiling their behaviour, extracting their pattern to visit the location, classifying the location, and matching profiles that are synchronised for specific sharing services.

The results denote a good example of how the presented profiling methodology can be used to assess the compatibility for long-term parking sharing of two or more users using, e.g., specific indicators for collaborative mobility between individuals [22]. More precisely, the profiling can be used to search in a specific region and users that have profiles that match other users for specific applications/sharing services. Nevertheless, there are some conditions that must be met to have an accurate long-term profiling.

First, there should be enough time data in order to have an accurate profiling. This will ensure that the location profile has a specific pattern over time and is not just a location that is visited randomly. *Home* and *work* locations are typical examples of locations that have a specific pattern over time.

Second, it is not required for the location accuracy to be extremely precise, as in the case of long parking duration, people are likely willing to walk for a decent location from the parking lot to destination [45].

In the same way, a P2P parking sharing service can be used also for ad hoc or instant parking sharing, presented in the following example. Particularly, if a user is part of the P2P parking sharing application and during a trip he notifies the application that he must stop for a specific time period in a specific place, the application can instantly search for other users in the system which has free parking slots during that specific time interval. In order to test this case study, using the Geolife dataset described in Section 4, we took a random user and a random visited location and performed a search for compatible users to simulate a match of an ad hoc P2P parking sharing request. Table 4 presents the results of different searches, at different resolutions.

As can be observed, at very small resolutions (i.e., 4.77×4.77 m and 38.2×19.1 m), no compatible users have been found, as the search is too detailed. When increasing the resolution to 153×153 m, three compatible users were found and the maximum error can be 108.187337 m which is acceptable. If we increase the resolution, more compatible users are found, but also the maximum possible error increases, to the extent that some results are not relevant, as the walking distance will be then too long and most likely not acceptable for the user to walk. In this case, we can argue that the best resolution would be in this case 153×153 m, which can give the best results regardless of the maximum possible error.

This concrete example demonstrates the capabilities that the proposed profiling methodology offers: fast processing large quantity of data, on demand and in near real time, coupled with the ability to extract user insights, behaviour, and travel pattern with minimum computation, storage resources, and user input. This is of a great importance for the next generation of AI autonomous travel planners and sharing services, as in most of the cases, the data collection

Table 4: Ad hoc P2P parking sharing matching.

L_x	L_y	Unit	Users matched	Max error
4.77	4.77	m	0	3.372899
38.2	19.1	m	0	21.354449
153	153	m	3	108.187337
1220	610	m	7	682.00000
4890	4890	m	11	3457.752000
39100	19,500	m	24	21846.398000

and processing are done through the passenger's mobile device. In a matter of seconds, the system must process years of geolocation data, extract the insights, user habits, and preferences, and provide reliable services. The fact that now it is possible to use the online tool [37] on any browser without installing any software and all the computation is done locally on the device is in line with the requirements of mobile devices which have limited autonomy and computation resources. At the same time, it also fits the mobile applications' user preferences because asking continuously user input information is no more applicable and sustainable in our days.

5.2. Ride Sharing. The profiling of users' mobility for the days and hours of the week is an important information that can be used for a recommendation system in order to analyse which users can match for ride sharing/carpooling. There are some conditions that should be met in order to organise a ride sharing between two or more users, such as the departure and arrival position to be suitable for all the participants and the departure and arrival time to synchronise matching at best their schedule. The latter condition can be assessed by analysing the probability to be in a specific location, by the days and hours of the week, and can be used in a collaborative mobility system [22].

In order to exemplify this case study, we searched in the small database presented in Section 4 for compatible respondents that can match a carpooling service. Figure 11 presents the extracted weekly heatmap of time spent in the residence for two neighbours (User 1 and User 2) that are working also in the same area, as can be seen in Figure 12. This is a typical situation where users can participate in a long-term ride sharing, as their schedule is pretty fixed in most cases.

As can be observed from the heatmap, they are at home typically outside of the working hours. Moreover, they both leave the house during the week around *9 AM* and they return at home around *7 PM*, resulting in a good synchronisation. Figure 13 shows that the schedule that they have for the work location makes it suitable for long-term carpooling as they can share the same car for commuting to work.

At the same time, we noticed also that also User 3 works close to User 1 and User 2. Moreover, User 3 can join the ride sharing when available. Analysing Figure 12, we can observe that in order to pick up User 3, the trip must be rerouted. If the first two users consider the travel time very important for them, this can be an inconvenience because the trip will be four minutes longer. However, if the users are flexible on timing and accept this trade-off, another passenger will be picked up in the same car and there will be a car less on the road and an available parking space at the destination. This approach will maximise the objectives of car sharing and parking sharing, by reducing the traffic and, respectively, the demand for parking space. Of course, users should assess different variations and accept or deny different options. In order to simplify the process, each user can state his preferences, limitations, and flexibility in an application interface so that the searching algorithm provides only suitable results for each user. Nevertheless, if this principle is applied at the level of big cities, it will result in more people in less cars and a reduced traffic congestion.

This concrete example shows how the profiling can be used both for long-term carpooling and short-term ride sharing as combined services. It is important to observe that using the proposed profiling methodology, all the necessary steps for matching people and sharing services (i.e., location visit pattern extraction, search of compatible users, and trip planner) can be done automatically and dynamically, without any user input but only the access of history GPS data, which in our days can be easily obtained via mobile devices.

5.3. Location Type and Activity Classification. Profiling the pattern for visiting specific locations gives also a possibility to automatically classify the location to a specific category. As we can observe in Figures 9–11, the location visit patterns obtained can be clearly identified as *Home* and *Work* locations and dynamically displayed as in Figure 14.

The same can be done with any type of location for which there are defined patterns which can be identified and the classification can be done automatically. In an extended exploratory study [36], we showed that using the combination of the proposed profiling method combined with observed user habits learned from surveys and extracted as activities matrix, it was possible to automatically classify the type of activity performed in a specific location.

Moreover, as the profiling can take a large time period into consideration, it is now possible to detect changes of user travel habits by detecting changes of regular visit to specific locations. In Figure 14, we can see that for the same user, two homes and workplaces are detected, as the profiling detects recursive similar patterns in different time periods. This means that the proposed profiling method can not only detect recurrent habits to secondary activities but also change of habits, something that is not easy to detect with a static method. The insights obtained based on these changes can be used to adapt and personalise transportation services to match the passenger's habits, which will result in a better service quality.

6. Future Work

Future work includes finding common subqueries across several requests and execute them once, something that can reduce the number of operations and tasks executed.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Sun	6	7	8	7	7	7	7	7	11	10	8	8	7	7	7	7	6	5	5	6	8	8	9	9
Mon	8	8	7	7	7	9	8	9	9	2	1	1	1	1	1	1	1	1	1	5	7	7	8	9
Tue	9	9	8	7	7	8	8	10	9	3	1		1	2	2	2	2	2	3	5	7	7	9	9
Wed	8	7	7	7	7	7	7	9	9	3	1	1	1	1		1	1	1	1	4	8	10	9	7
Thu	8	8	7	7	6	7	7	9	9	4	2	2	2	2	2	2	2	2	3	5	7	7	8	9
Fri	9	10	9	7	7	6	7	8	9	3	1	1	1	1	1	1	1	2	3	6	6	5	6	6
Sat	5	5	5	6	7	6	6	7	8	10	8	5	2	2	3	5	6	7	6	6	6	5	6	6

(a)

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Sun	7	6	6	6	5	6	7	10	10	9	10	9	9	7	7	6	8	8	8	7	9	8	9	10
Mon	9	7	6	7	7	7	10	11	5	1										2	8	10	12	11
Tue	8	7	7	6	7	8	11	12	7	1										3	9	11	12	11
Wed	8	7	5	5	6	7	10	10	6	1	1	1					1	1	2	3	6	9	9	9
Thu	8	6	6	6	7	7	10	11	8	1									1	3	7	9	11	9
Fri	8	6	6	6	6	7	10	11	7	2	1	1	1	2	1	1	1	2	4	6	8	9	9	9
Sat	8	7	7	5	6	7	8	10	12	9	8	7	5	4	4	4	5	5	7	6	6	6	6	7

(b)

Figure 11: Home location profile for User 1 and User 2.

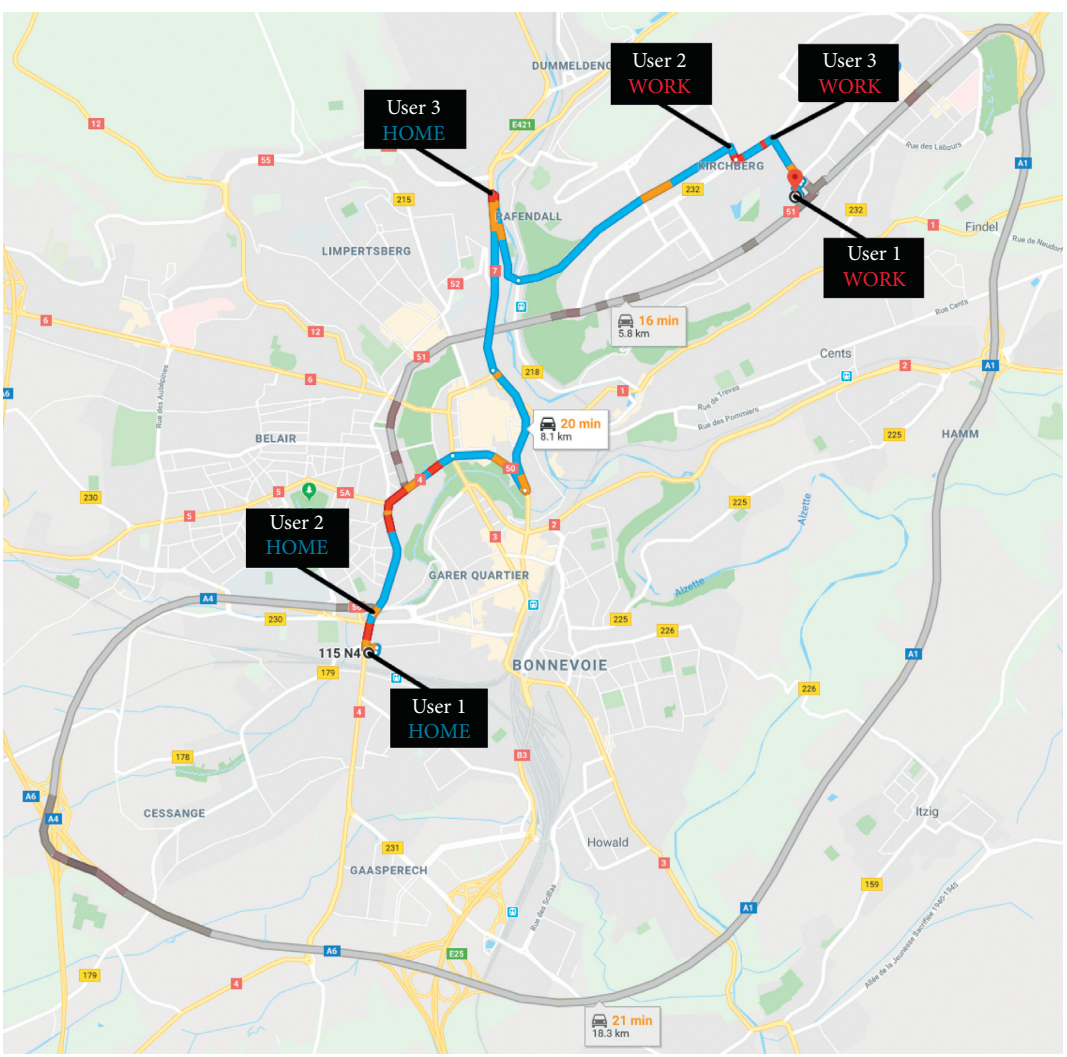


Figure 12: Ride sharing case study with three users.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Sun											1	1						1	1	1				
Mon									1	17	32	23	20	22	17	16	17	15	16	9	3			
Tue										16	24	17	26	25	18	16	15	16	17	7	2			
Wed										15	18	13	17	21	20	13	15	12	15	7	3	1	1	
Thu									1	11	14	12	18	19	17	18	17	20	17	8	2	1	1	
Fri									1	11	23	13	17	20	15	16	15	14	15	7	3	2		
Sat											1	3	2	3	3	1	1	1	1	1	1			

(a)

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Sun	1														2	3	3	3	3	2	1	1		
Mon									6	16	17	15	10	15	19	22	17	10	8	8	6	3	1	
Tue									5	17	20	20	11	21	21	19	17	14	11	6	5	1		
Wed									6	15	19	18	16	20	18	15	16	15	12	12	8	6	2	
Thu									3	15	18	17	9	17	19	17	16	13	8	9	5	2		
Fri									5	17	17	17	9	15	15	15	16	14	9	8	3	1	1	
Sat												4	2		1	5	6	3	5	2	2		1	1

(b)

Figure 13: Profile of work location for User 1 and User 2.

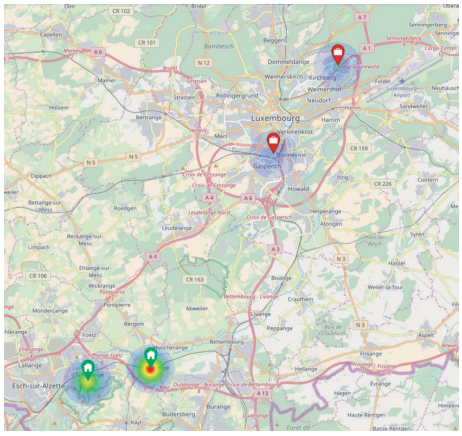


Figure 14: Location type classification: Home and Work classification. Residence and workplace change detection.

At the same time, caching techniques can be useful in the future to store temporally the results of the most asked queries on the most recent trees. Another optimisation would be the implementation of a subscription system which will perform automatically live updates of latest queries and trees.

The presented usage examples show that only using the geolocation data, it is already possible to support some sharing services (e.g., parking sharing and car sharing—as for those services only presence/absence of users/resources in time and space is needed), detect travel habits, and label/classify location and activities. In the future, an implementation of a route planner can offer the possibility to have a complete ride sharing service which can match people and vehicles. On the other hand, the usage of semantic external data of visited locations (e.g., type of facilities from existing maps) can better infer secondary activity types and reduce the identification and classification errors.

7. Conclusion

The contribution of this paper is twofold: on the one hand, we present a novel methodology that provides a dynamic profiling of users' mobility and locations' visit pattern. The proposed profiling method can be used in many applications and even in a simultaneous manner. The usage examples explained and evaluated throughout the current paper (i.e., parking sharing, ride sharing, location type, and activity classification) provide the first directions on how the profiling can be used for a dynamic analysis of sharing mobility users and solutions.

On the other hand, using state-of-the art technologies from data science and computer science, we provide a complete implementation of the proposed methodology which can be tested through an online demonstrative prototype. The demo application demonstrates how it is possible to load the data and extract complex profiles from geolocation data (i.e., location data from Google Maps), with different accuracy levels and spatiotemporal scales, in an order of magnitude of milliseconds. Moreover, for any visited location, a classification is dynamically performed, which demonstrates that different actions and computations can be performed in motion, at scale and in near real time. Different evaluations were performed in order to assess the speed, scalability, and to evaluate the required resources for implementation, which demonstrates that the proposed profiling can be implemented in a distributed way at the

smallest hardware level (e.g., microcomputers or mobile devices).

Data Availability

This study employed Google Visited Places API for the small dataset example, whereas it used the Microsoft GeoLife dataset (https://www.microsoft.com/en-us/download/details.aspx?id=52367) for the large scale example.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was funded by the Luxemburgish FNR (Fonds National de la Recherche) through an AFR grant for the PLAYMOBeL project (9220491) and by the EU Marie-Curie-funded project InCoMMune (618234).

References

[1] Inmarsat research programme report: the future of IoT in enterprise, 2017, https://goo.gl/rvumkN.

[2] IDC, "Worldwide internet of things spending guide report," https://www.idc.com/getdoc.jsp?containerId=prUS44596319.

[3] J. Manyika, M. Chui, B. Brown et al., *Big Data: The Next Frontier for Innovation, Competition, and Productivity*, Technical Report, McKinsey Global Institute, Washington, DC, USA, 2011.

[4] ITF Forum, *ITF Transport Outlook 2017*, OECD Publishing, Paris, France, 2017.

[5] T. Hartmann, F. Fouquet, M. Jimenez, R. Rouvoy, and Y. Le Traon, "Analyzing complex data in motion at scale with temporal graphs," in *Proceedings of the 29th International Conference on Software Engineering & Knowledge Engineering (SEKE'17)*, p. 6, KSI Research, Pittsburgh, PA, USA, July 2017.

[6] T. Hartmann, F. Fouquet, J. Klein et al., "Generating realistic smart grid communication topologies based on real-data," in *Proceedings of the 2014 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pp. 428–433, IEEE, Venice, Italy, November 2014.

[7] T. Hartmann, *Enabling model-driven live analytics for cyber-physical systems: The case of smart grids*, Ph.D. thesis, University of Luxembourg, Luxembourg, Luxembourg, 2016.

[8] A. Sassi and F. Zambonelli, "Coordination infrastructures for future smart social mobility services," *IEEE Intelligent Systems*, vol. 29, no. 5, pp. 78–82, 2014.

[9] G. Castignani, T. Derrmann, R. Frank, and T. Engel, "Driver behavior profiling using smartphones: a low-cost platform for driver monitoring," *IEEE Intelligent Transportation Systems Magazine*, vol. 7, no. 1, pp. 91–102, 2015.

[10] S.-T. Cheng, G.-J. Horng, and C.-L. Chou, "Using cellular automata to form car society in vehicular ad hoc networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 4, pp. 1374–1384, 2011.

[11] J. F. Júnior, E. Carvalho, B. V. Ferreira et al., "Driver behavior profiling: an investigation with different smartphone sensors and machine learning," *PLoS One*, vol. 12, no. 4, p. 959, Article ID e0174, 2017.

[12] E. Ozatay, S. Onori, J. Wollaeger et al., "Cloud-based velocity profile optimization for everyday driving: a dynamic-programming-based solution," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 6, pp. 2491–2505, 2014.

[13] P. Campigotto, C. Rudloff, M. Leodolter, and D. Bauer, "Personalized and situation-aware multimodal route recommendations: the favour algorithm," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 1, pp. 92–102, 2017.

[14] S. Canale, A. Di Giorgio, F. Lisi et al., "A future internet oriented user centric extended intelligent transportation system," in *Proceedings of the 2016 24th Mediterranean Conference on Control and Automation (MED)*, pp. 1133–1139, IEEE, Athens, Greece, June 2016.

[15] F. Giannotti, L. Gabrielli, D. Pedreschi, and S. Rinzivillo, "Understanding human mobility with big data," in *Solving Large Scale Learning Tasks. Challenges and Algorithms*-Springer, Berlin, Germany, 2016.

[16] E. Tonnelier, N. Baskiotis, V. Guigue, and P. Gallinari, "Smart card in public transportation: designing a analysis system at the human scale," in *Proceedings of the 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, pp. 1336–1341, IEEE, Rio de Janeiro, Brazil, November 2016.

[17] E. Manley, C. Zhong, and M. Batty, "Spatiotemporal variation in travel regularity through transit user profiling," *Transportation*, pp. 1–30, 2016.

[18] J. Ghosh, M. J. Beal, H. Q. Ngo, and C. Qiao, "On profiling mobility and predicting locations of wireless users," in *Proceedings of the 2nd International Workshop on Multi-Hop Ad Hoc Networks: from Theory to Reality*, pp. 55–62, ACM, Florence, Italy, 2006.

[19] J. Zhang, F.-Y. Wang, K. Wang et al., "Data-driven intelligent transportation systems: a survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 4, pp. 1624–1639, 2011.

[20] N. Bicocchi and M. Mamei, "Investigating ride sharing opportunities through mobility data analysis," *Pervasive and Mobile Computing*, vol. 14, pp. 83–94, 2014.

[21] N. Agatz, A. Erera, M. Savelsbergh, and X. Wang, "Optimization for dynamic ride-sharing: a review," *European Journal of Operational Research*, vol. 223, no. 2, pp. 295–303, 2012.

[22] B. Toader, F. Sprumont, S. Faye, M. Popescu, and F. Viti, "Usage of smartphone data to derive an indicator for collaborative mobility between individuals," *ISPRS International Journal of Geo-Information*, vol. 6, no. 3, p. 62, 2017.

[23] L. Montini, N. Rieser-Schüssler, A. Horni, and K. W. Axhausen, "Trip purpose identification from GPS tracks," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2405, no. 1, pp. 16–23, 2014.

[24] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[25] D. Heggie and P. Hut, *The Gravitational Million-Body Problem: A Multidisciplinary Approach to Star Cluster Dynamics*, Cambridge University Press, Cambridge, UK, 2003.

[26] M. Winkel, R. Speck, H. Hübner, L. Arnold, R. Krause, and P. Gibbon, "A massively parallel, multi-disciplinary Barnes-Hut tree code for extreme-scale N-body simulations," *Computer Physics Communications*, vol. 183, no. 4, pp. 880–889, 2012.

[27] H. Samet, "The quadtree and related hierarchical data structures," *ACM Computing Surveys*, vol. 16, no. 2, pp. 187–260, 1984.

[28] K. Yamaguchi, T. Kunii, K. Fujimura, and H. Toriya, "Octree-related data structures and algorithms," *IEEE Computer Graphics and Applications*, vol. 4, no. 1, pp. 53–59, 1984.

[29] S. J. Redmond and C. Heneghan, "A method for initialising the *k*-means clustering algorithm using *k*-*d*-trees," *Pattern Recognition Letters*, vol. 28, no. 8, pp. 965–973, 2007.

[30] C. D. Toth, J. O'Rourke, and J. E. Goodman, *Handbook of Discrete and Computational Geometry*, CRC Press, Boca Raton, FL, USA, 2004.

[31] G. Qian, Q. Zhu, Q. Xue, and S. Pramanik, "Dynamic indexing for multidimensional non-ordered discrete data spaces using a data-partitioning approach," *ACM Transactions on Database Systems*, vol. 31, no. 2, pp. 439–484, 2006.

[32] D. Kolbe, Q. Zhu, and S. Pramanik, "Efficient *k*-nearest neighbor searching in nonordered discrete data spaces," *ACM Transactions on Information Systems*, vol. 28, no. 2, p. 7, 2010.

[33] "DataThings: greycat framework," 2017, https://github.com/datathings/greycat.

[34] F. Francois, G. Nain, B. Morin et al., "Kevoree Modeling Framework (KMF): efficient modeling techniques for runtime use," 2014, http://arxiv.org/abs/1405.6817.

[35] B. Toader, A. Moawad, F. Fouquet, T. Hartmann, M. Popescu, and F. Viti, "A new modelling framework over temporal graphs for collaborative mobility recommendation systems," in *Proceedings of the 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, pp. 1–6, IEEE, 2017.

[36] B. Toader, G. Cantelmo, M. Popescu, and F. Viti, "Using passive data collection methods to learn complex mobility patterns: an exploratory analysis," in *Proceedings of the 2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pp. 993–998, IEEE, Maui, HI, USA, November 2018.

[37] Online profiler tool link, 2018, https://mobilab.lu/profiler-demo/.

[38] Microsoft Research Asia: Geolife GPS Trajectories, 2018, https://www.microsoft.com/en-us/download/details.aspx?id=52367.

[39] Y. Zheng, Y. Chen, X. Xie, and W. Y. Ma, "Geolife 2.0: a location-based social networking service," in *Proceedings of the Tenth International Conference on Mobile Data Management: Systems, Services and Middleware*, pp. 357-358, IEEE, Taipei, Taiwan, 2009.

[40] Guido Cantelmo Bogdan Toader, C. A. F. V.: Inferring Urban Mobility and Habits from User Location History, 2019.

[41] S. Breß, M. Heimel, N. Siegmund, L. Bellatreche, and G. Saake, "GPU-accelerated database systems: survey and open challenges," in *Transactions on Large-Scale Data-and Knowledge-Centered Systems XV*Springer, Berlin, Germany, 2014.

[42] T. Mostak, *An Overview of Mapd (Massively Parallel Database), White Paper*, Massachusetts Institute of Technology, Cambridge, MA, USA, 2013.

[43] M. Litwintschik, "1.1 billion taxi rides with mapd 8 nvidia tesla k80s," 2018, http://tech.marksblogg.com/billion-nyc-taxi-rides-nvidia-tesla-mapd.html.

[44] Transportation Research Board, *Transit Capacity and Quality of Service Manual-TCRP Report 165*, 3rd Edition, Washington DC, USA, 2003.

[45] P. van der Waerden, H. Timmermans, and 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, 2017.

Hindawi
Journal of Advanced Transportation
Volume 2020, Article ID 8843783, 10 pages
https://doi.org/10.1155/2020/8843783

Research Article

Dynamic Scheduling Model of Bike-Sharing considering Invalid Demand

Liu He,[1,2] **Tangyi Guo**,[1,2] **and Kun Tang**[1,2]

[1]*Department of Automation, Nanjing University of Science and Technology, Nanjing, Jiangsu 210094, China*
[2]*MIIT Key Lab of Traffic Information Fusion & System Control, Nanjing 210094, China*

Correspondence should be addressed to Tangyi Guo; transtor@njust.edu.cn

Received 26 June 2020; Revised 19 October 2020; Accepted 2 December 2020; Published 15 December 2020

Academic Editor: Dongfang Ma

Copyright © 2020 Liu He 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.

System resources allocation optimization through dynamic scheduling is key to improving the service level of bike-sharing. This study innovatively introduces three types of invalid demand with negative effect including waiting, transfer, and abandoning, which consists of the total demand of bike-sharing system. Through exploring the dynamic relationship among users' travel demands, the quantity and capacity of bikes at the rental points, the records of bicycles borrowed and returned, and the vehicle scheduling schemes, a demand forecasting model for bike-sharing is established. According to the predicted bikes and the maximum capacity limit at each rental point, an optimization model of scheduling scheme is proposed to reduce the invalid demand and the total scheduling time. A two-layer dynamic coupling model with iterative feedback is obtained by combining the demand prediction model and scheduling optimization model and is then solved by Nicked Pareto Genetic Algorithm (NPGA). The proposed model is applied to a case study and the optimal solution set and corresponding Pareto front are obtained. The invalid demand is greatly reduced from 1094 to 26 by an effective scheduling of 3 rounds and 96 minutes. Empirical results show that the proposed model is able to optimize the resource allocation of bike-sharing, significantly reduce the invalid demand caused by the absence of bikes at the rental point such as waiting in a place, walking to other rental points, and giving up for other travel modes, and effectively improve the system service level.

1. Introduction

With the popularization of the concept of sharing economy and green travel, bike-sharing, as an energy-saving, environmental protection, flexible, and healthy way of traveling, is being increasingly popular. Bike-sharing refers to the bike-sharing service provided by enterprises in residential areas, commercial areas, public service areas, public transportation stations, campuses, etc. As a new type of sharing economy based on time-sharing rental model, this service has developed rapidly in recent years. According to the Global Development Report on Bike-sharing (2018) [1] published by Cheetah Lab, a global mobile data research firm showed that, in 2019, there were more than 1,000 bike-sharing companies and more than 300 million users. While vigorously developing bike-sharing, various problems have also emerged. The most prominent problem is unavailability of bikes during peak hours which is due to uneven distribution of bikes; some rental points have excessive bikes which sometimes disrupt the road while some rental points have no bikes. In view of this problem, apart from planning the layout and scale of the bike-sharing rental points [2], it is also imperative to establish an accurate and efficient dynamic scheduling system of bike-sharing and maximize the advantages of sharing the economy by improving the service level of system.

The research of bike-sharing scheduling problem mainly focuses on path selection and optimization. Ho and Szeto designed an iterated tabu search heuristic to solve the static bike repositioning problem [3]. Dell'Amico et al. considered the Bike-sharing Rebalancing Problem (BRP) as a special one-commodity pickup-and-delivery capacitated vehicle routing problem and presented four mixed integer linear programming formulations of this problem [4]. Brinkmann

et al. presented an inventory routing problem for bike-sharing systems [5]. Liu et al. proposed a multiobjective model under one hybrid mode with morning fixed and evening demand-responsive by considering the effect on endogenous demands from bike-sharing and studied the influence on feeder transit services [6]. Shi et al. formulated the VRP model for bike-sharing inventory rebalancing and vehicle routing and designed an improved particle swarm optimization (PSO) algorithm to solve this problem [7]. Bulhões et al. established an integer programming model to describe the static bike relocation problem with multiple vehicles and visits [8]. Schuijbroek et al. took service level requirements at each bike-sharing station into account and designed route optimization model. Then, a cluster-first route-second heuristic algorithm was proposed to solve the model [9]. Caggiani et al. proposed a comprehensive dynamic redistribution methodology to mitigate the imbalance of bicycles among zones. The optimal model is aimed to maximize the satisfaction of users and minimize the repositioning cost of vehicles [10]. Guan et al. proposed a layered scheduling strategy for bike-sharing and a static scheduling model with minimum time cost as the objective and vehicle capacity as the constraint is established [11]. Ahmed et al. proposed four upper bounds based on a genetic algorithm, a greedy search algorithm and two hybrid methods that integrate a genetic algorithm, a local search method, and a branch-and-bound algorithm for the balancing problem of bike-sharing stations [12].

In the existing research, dynamic scheduling of bike-sharing is defined as vehicle routing problems with time window (VRPTW). Pureza et al. presented a mathematical programming formulation for VRPTW, as well as a tabu search and an ant colony optimization heuristic for obtaining minimum cost routes [13]. Ben Ticha et al. presented an empirical analysis based on VRPTW and solved with branch-and-price algorithms developed for the different types of graphs [14]. Harzi and Krichen proposed a VND approach and benchmark instances were applied to demonstrate the performance [15]. Hernandez et al. proposed the branch-price framework to solve the multitrip vehicle routing problem with time windows [16]. Pierre and Zakaria proposed a stochastic optimized cyclic shift crossover operator for the optimization model with time windows using genetic algorithms [17]. Ma et al. proposed a back-pressure-based model with fixed phase sequences for traffic signal optimization under oversaturated networks and a forecasting model for daily traffic flow through a contextual convolutional recurrent neural network as well [18, 19].

The above literature mainly focused on the optimization method of path planning algorithm for scheduling vehicles, while most of them considered the demand for bike-sharing trips as static. Actually, as a new mode of transportation, bike-sharing possesses the characteristics of both public transportation and slow traffic.

The problem comes originally from the dynamic contradiction between the existing vehicles, parking space capacity, and the demand for borrowing and returning vehicles. Based on the traffic travel theory, it is the focus of this paper to predict the vehicle demand of each rental point over multiple periods and to formulate the dynamic scheduling scheme. The cause of the problem is the dynamic contradiction between the existing bicycles, capacity, and the demand for borrowing and returning at rental points. Based on the theory of transportation travel, this paper focuses on the multi-period prediction of bicycle demand at each rental point and the formulation of dynamic scheduling scheme.

The rest of the paper is organized as follows: Section 2 presents the procedure of establishing a two-layer dynamic coupling model through 5 steps. Section 3 conducts a case study to illustrate the feasibility and effectiveness. Section 4 concludes with a summary of the main findings in this research and suggestions for future study.

2. Methodology

2.1. Problem Description. Bike-sharing is generally concentrated in residential areas, commercial areas, subway stations, and other places with high demand for travel and transfer. Therefore, the continuous distribution of bike-sharing in space can be aggregated into a series of discrete rental points on the road network. All bike-sharing users travel from one rental point to another.

The bike-sharing scheduling system consists of a series of bike rental points, scheduling vehicles, and a parking base. The scheduling process is as follows: the command center generates a scheduling scheme according to the scheduling requirements of one or more rental points, which is executed by the scheduling vehicles. The vehicles start from the parking base, load and unload the bikes at multiple rental points according to the scheme, and finally return to the base.

The difference between bike-sharing scheduling problems and general vehicle dynamic scheduling problems is as follows:

(a) Fairness: as a form of public transport, bike-sharing treats all users equally, and there is no indicator of the importance of the rental points.

(b) Flexibility: there is no fixed path or rigid time window for bike-sharing scheduling.

(c) Flow direction: bikes between rental points will generate flow direction according to travel demand; that is, bikes borrowed from one point must return to that point or arrive at another point after a period. This demand is predictable.

(d) Waiting, transfer, and abandoning: when the travel demand at one point is greater than the existing bikes, the excess demand waits in place according to the expected cost, transfers to the nearby rental point (voluntarily looking for bikes), or even cancels (drop for other modes of travel), the probability of which is affected by the user's personal characteristics.

The above characteristics determine that the bike-sharing scheduling scheme must be premised on the dynamic change of travel demand. Furthermore, it depends on the relationship between the number of bikes and the demand at the point when the scheduling vehicles arrive.

Therefore, this paper describes the dynamic scheduling problem of bike-sharing as follows:

The bike-sharing system consists of H scheduling vehicles, 1 parking base, and N rental points.

The scheduling period is from t_0 to t_K. The capacity of each rental point is C_i, $i = 1, 2, \cdots, N$. The bikes at rental point i at time t_k, $k = 0, 1, \cdots, K$ is $V_i^{t_k}$; the total demand of borrowing bikes is indicated as $D_i^{t_k}$. According to the current status and location of the scheduling vehicle h, combined with the borrowing and returning records of the past, the scheduling demand of each rental point is predicted, and a scheduling optimization scheme including scheduling vehicles, scheduling order (set each rental point to be scheduled only once), arrival time, and number of bikes is formulated, so as to ensure the lowest cost of travel for all bike-sharing users at the minimum cost.

2.2. Model Formulation. According to the analysis above, the dynamic scheduling model of bike-sharing includes two submodels: lower demand prediction and upper scheduling optimization. The former is used to establish the dynamic relationship among the users' travel demands, the quantity and capacity of bikes at the rental points, the records of bicycles borrowed and returned, and the vehicle scheduling schemes, while the latter formulates the optimal scheduling scheme according to the prediction results.

2.3. Demand Forecasting Model. Let us start with a definition of the invalid demand of bike-sharing, as shown in Figure 1.

When a user arrives at a rental point, the bike would be successfully borrowed and cycled to another rental point if there are available bikes. Otherwise, the user will face three options: wait in place for other users to return bikes, walk to another rental point to borrow a bike, or give up and switch to other modes of travel.

Let the total demand and returned bikes at the time of t_k rental point i be $L_i^{t_k}$ and $R_i^{t_k}$, respectively, and the scheduling amount is $S_i^{t_k}$; then, the bikes at the next time $V_i^{t_{k+1}}$ are the result of the current bikes $V_i^{t_k}$ minus the total demand bikes $L_i^{t_k}$, plus the valid returned bikes $R_i^{t_k}$ as well as the scheduling bikes $S_i^{t_k}$:

$$V_i^{t_{k+1}} = V_i^{t_k} - L_i^{t_k} + R_i^{t_k} + S_i^{t_k}. \tag{1}$$

The components are shown as follows.

2.3.1. Valid Demand $L_i^{t_k}$. If the current bikes meet the total demand for borrowing at this time, then the valid demand is the total demand; otherwise, all bikes will be borrowed. The valid demand is represented as

$$L_i^{t_k} = \begin{cases} D_i^{t_k}, & D_i^{t_k} \le V_i^{t_k}, \\ V_i^{t_k}, & D_i^{t_k} > V_i^{t_k}. \end{cases} \tag{2}$$

2.3.2. Invalid Demand $\widetilde{L}_i^{t_k}$. The invalid demand $\widetilde{L}_i^{t_k}$ can be expressed as

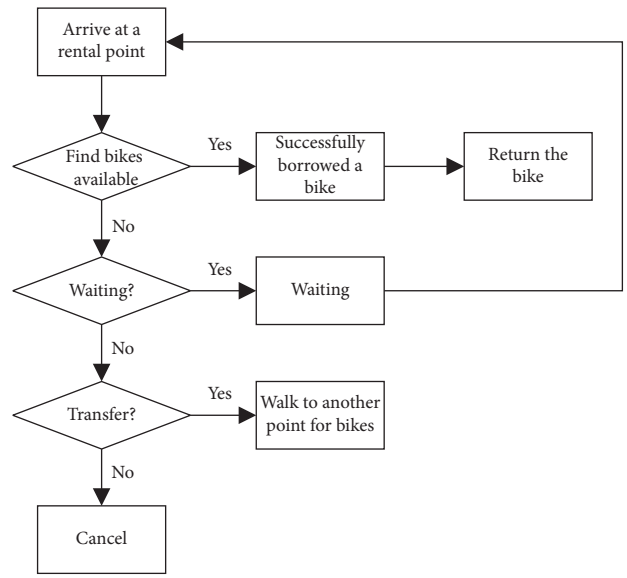


Figure 1: The invalid demand of bike-sharing.

$$\widetilde{L}_i^{t_k} = D_i^{t_k} - V_i^{t_k} = W_i^{t_k} + T_i^{t_k} + A_i^{t_k}, \quad D_i^{t_k} > V_i^{t_k}. \tag{3}$$

In the formula, $\widetilde{L}_i^{t_k}$ represents the invalid number of bikes borrowed by time t_k and at rental point i. $W_i^{t_k}$, $T_i^{t_k}$, and $A_i^{t_k}$ represent the number of waiting, transfer, and abandoning of bikes by time t_k at the rental point i.

The following assumptions are made for the invalid demand and formulas (4) to (7) are obtained:

(a) The waiting probability $pw_i^{t_k}$ of the rental point i is positively correlated with the valid returned bikes during this period and negatively correlated with the waiting time and the users waiting in the previous period.

(b) The transfer probability $pt_{ij}^{t_k}$ of walking from the rental point i to rental point j is positively correlated with the waiting users at the last time period of the rental point i, and the number of bikes that can be borrowed at rental point j, and negatively correlated with the expected returned bikes, and shortest path distance between rental points i and j.

(c) The abandoning probability $pa_i^{t_k}$ at rental point i is positively correlated with the waiting users in the previous period and negatively correlated with the expected returned of bikes at point i during this period.

(d) All invalid users only make one choice within one period and cannot be changed.

$$pw_i^{t_k} = \theta_1 \cdot pw_i^{t_{k-1}} \cdot \frac{R_i^{t_k}}{W_i^{t_{k-1}} C_i}, \tag{4}$$

$$pt_{ij}^{t_k} = \theta_2 \cdot \frac{W_i^{t_{k-1}} \cdot V_j^{t_k}}{R_i^{t_k} \cdot d_{ij}}, \tag{5}$$

$$pa_i^{t_k} = \theta_3 \cdot \frac{W_i^{t_{k-1}}}{R_i^{t_k}}, \tag{6}$$

$$pw_i^{t_k} + \sum_{j=1}^{N} pt_{ij}^{t_k} + pa_i^{t_k} = 1. \tag{7}$$

In the formula, θ_1, θ_2, θ_3, and $pw_i^{t_0}$ are all undetermined coefficients greater than 0, which are related to the user's individual characteristics, and can be determined by SP/RP survey. d_{ij} in formula (5) represents the shortest path distance between the rental points i and j, which is generally implemented by Dijkstra algorithms. Equation (7) guarantees that the sum of the probabilities of waiting, transfer, and abandoning is 1.

The above formula is summed up to get the amount of waiting, transfer, and abandoning in the invalid demand of time t_k at rental point i.

$$\begin{aligned} T_i^{t_k} &= \sum_{s=1}^{\widetilde{L}_i^{t_k}} pt_i^{t_k}, \\ &= \theta_2 \frac{V_j^{t_k}}{R_i^{t_k} \cdot r_{ij}} \sum_{s=1}^{\widetilde{L}_i^{t_k}} W_i^{t_{k-1}}, \\ W_i^{t_k} &= \sum_{s=1}^{\widetilde{L}_i^{t_k}} pw_i^{t_k}, \\ &= \theta_1 \frac{R_i^{t_k}}{C_i} \sum_{s=1}^{\widetilde{L}_i^{t_k}} \frac{pw_i^{t_{k-1}}}{W_i^{t_{k-1}}}, \\ A_i^{t_k} &= \sum_{s=1}^{\widetilde{L}_i^{t_k}} pa_i^{t_k}, \\ &= \theta_3 \frac{1}{R_i^{t_k}} \sum_{s=1}^{\widetilde{L}_i^{t_k}} W_i^{t_{k-1}}. \end{aligned} \tag{8}$$

2.3.3. Valid Return $R_j^{t_k}$. The valid return bikes are determined by the valid demand at the starting rental point and the distribution probability matrix between the starting point and the ending point. The latter is defined as

$$\begin{aligned} P^{t_k} &= \begin{pmatrix} P_{11}^{t_k} & \cdots & P_{1N}^{t_k} \\ \vdots & P_{ij}^{t_k} & \vdots \\ P_{N1}^{t_k} & \cdots & P_{NN}^{t_k} \end{pmatrix}, \\ s.t.\ & 0 \le P_{kj} \le 1, \\ & \sum_{j=1}^{n} P_{kj} = 1, \quad k = 1, 2, \ldots, n. \end{aligned} \tag{9}$$

In the formula, $P_{ij}^{t_k}$ represents the probability of borrowing bikes from rental point i at time t_k and returning them at rental point j, which can be obtained from the operation records of bike-sharing.

Since bikes are generally ridden in nonmotorized lanes, the capacity limit of the road is generally not considered, so the shortest path of distance can be used for conversion. The valid return $R_j^{t_k}$ at rental point j is expressed as

$$R_j^{t_k} = \sum_{i=1}^{N} L_i^{t_k - (d_{ij}/v)} P_{ij}^{t_k - (d_{ij}/v)}, \tag{10}$$

where v represents the average speed of the bike.

2.3.4. Scheduling $S_i^{t_k}$. Set each rental point to be scheduled only once, so $S_i^{t_k}$ is abbreviated as S_i. The scheduling bikes at rental point i are determined by scheduling optimization model. $S_i < 0$ indicates loading at the rental point, while $S_i > 0$ indicates unloading.

2.4. Scheduling Optimization Model. The optimization model for scheduling is based on the predicted result of bikes and the maximum capacity at each rental point. On the one hand, the optimal scheduling scheme is to ensure the service level and reduce the invalid demand. On the other hand, it will also reduce the scheduling cost and improve the efficiency of the operating enterprise. This is a multiobjective problem and the objective function is established as follows:

$$\begin{aligned} \min(Z_1) &= \sum_{k=0}^{K} \sum_{i=1}^{N} \left(\beta_1 W_i^{t_k} + \beta_2 T_i^{t_k} + \beta_3 A_i^{t_k}\right), \\ &= \sum_{k=1}^{K} \sum_{i=1}^{N} \left(\beta_1 \theta_1 \frac{R_i^{t_k}}{C_i} \sum_{s=1}^{\widetilde{L}_i^{t_k}} \frac{pw_i^{t_{k-1}}}{W_i^{t_{k-1}}} + \beta_2 \theta_2 \frac{V_j^{t_k}}{R_i^{t_k} \cdot r_{ij}} \sum_{s=1}^{\widetilde{L}_i^{t_k}} W_i^{t_{k-1}} \right. \\ &\quad \left. + \beta_3 \theta_3 \frac{1}{R_i^{t_k}} \sum_{s=1}^{\widetilde{L}_i^{t_k}} W_i^{t_{k-1}} \right), \end{aligned} \tag{11}$$

$$s.t.\ \forall V_i^{t_k} = 0 \bigcup V_i^{t_k} > C_i B_j^{t_{i \times dt}} < 0, B_l^{t_{i \times dt}} > C_l, \tag{12}$$

$$\min(Z_2) = \sum_{h=1}^{H} \sum_{i=0}^{N} \sum_{j=0, j \ne i}^{N} \left(\text{QUOTE}\, t_a\, t_a X_{ijh} + \gamma_0 |S_i| \right), \tag{13}$$

$$\begin{aligned} s.t.\ & i = 0, \sum_{h=1}^{H} \sum_{j=0}^{N} X_{ijh} = H, \sum_{h=1}^{H} y_{hi} = 1, \left| \sum_{i=0}^{N} \sum_{j=0, j \ne i}^{N} S_i X_{ijh} \right| \le Q_h, \\ & \forall h = 1, 2, \cdots, H, t_j \\ & = \sum_{h=1}^{H} \sum_{j=0, j \ne i}^{N} X_{ijh} \left(t_i + \gamma_{ij} + \gamma_0 |S_i|\right), X_{ijh} \\ & = \begin{cases} 1, \text{ vehicle } h \text{ drive from rental point } i \text{ to } j, \\ 0, \text{ otherwise,} \end{cases} y_{hi} \\ & = \begin{cases} 1, \text{ rental point } i \text{ is served by vehicle } h, \\ 0, \text{ otherwise.} \end{cases} \end{aligned} \tag{14}$$

In the formula, t_a represents the travel time of the scheduling vehicle from rental point i to j, γ_0 represents the average loading and unloading time of one bike, H is the total scheduling vehicles, and Q_h is the maximum load of vehicle h.

The objective function (14) indicates that the minimum invalid demand has a negative effect on the service level of bike-sharing. Since this is a Soft Time Windows Problem, the time window constraints are incorporated into the target function by introducing waiting penalty factor β_1, transfer penalty factor β_2, and abandoning penalty factor β_3. Condition (15) indicates that the schedule needs to meet the situation where there is no bike to borrow or the rental point exceeds its capacity.

The objective function (16) indicates that the scheduling cost of an enterprise is minimized, which is represented by the sum of the driving time, the loading, and unloading time of the bikes. Condition (17) indicates that there are H paths for departure from the rental point and H paths for arrival. Condition (18) indicates that the rental point i can only schedule one bike, condition (19) indicates that the loading number of each vehicle does not exceed the capacity, condition (20) indicates that the time to reach the rental point j is equal to the time when the vehicle reaches point i plus the loading and unloading time at point j and the travel time from point i to point j.

2.5. Model Solution. The lower demand prediction model regards all the bike-sharing rental points in the area as a whole system, and it is appropriate to establish a system dynamics simulation model to solve the changes of system characteristics in the subsequent moments with the current user demand, historical travel distribution, and scheduling scheme for the input and deduction. The upper level scheduling optimization model is a multiobjective optimization problem with a time window (vehicle routing problems with time windows, VRPTW) that determines the time and order of the scheduling vehicle through each rental point based on the dual objectives of invalid demand and enterprise scheduling cost. It is generally solved by heuristic algorithm [20].

In the simulation process, the predicted scheduling demand is influenced by the number of past bikes borrowed, which is the basis for the development of scheduling scheme, which will affect the demand prediction results of all rental points in the process of development. At the same time, the implementation of scheduling affects the valid demand and pushes the time forward. Compare the difference between valid and predicted demand to determine whether the scheduling scheme needs to be adjusted. When the actual time reaches the end, the result will be output. The simulation process is shown in Figure 2.

Throughout the simulation process, on the one hand, demand prediction and scheduling optimization are intertwined with constant feedback; on the other hand, scheduling optimization is a multiobjective VRP problem, which needs to be considered separately. For the traditional VRP problem, heuristic algorithm is generally used to solve the problem. The selection of bike-sharing requirement path needs the support of modelled and quantitative methods, and it has the complete nature of NP. Genetic algorithm has some outstanding advantages for solving this kind of problem. Since the submodel of bike-sharing scheduling involves two objective equations (11), (12) of user service level and enterprise scheduling cost, it is advisable to adopt a multiobjective genetic algorithm (MOGA). This method is an algorithm developed in recent years to deal with multiobjective optimization problems using genetic algorithms. Traditional methods are difficult to deal with large-scale problems; MOGA not only can deal with large-scale problems but also is not limited by the nature of the problem (linear, continuous, microusability, multipeak, etc.) and can search for the global optimal solution of the problem. It is also independent of the form of Pareto's optimal frontier compared to conventional optimization methods [15]. We choose the Ned Pareto Genetic Algorithm (NPGA) which is more economical to calculate.

The specific model solving steps are as follows.

Step 1. Starting from t_0, the matrix is constructed by the demand forecasting model to predict the number of bikes at each rental point at any time in the future based on historical bikes borrowing records.

$$V = \begin{pmatrix} V_1^{t_0} & \cdots & V_N^{t_0} \\ \vdots & V_i^{t_k} & \vdots \\ V_1^{t_K} & \cdots & V_N^{t_K} \end{pmatrix}. \tag{15}$$

Step 2. Select n $(n \leq N)$ rental points to be scheduled from V to satisfy $V_i^{t_k} = 0$ or $V_i^{t_k} > C_i$ or other safety inventory conditions. If there is no rental point that needs to be scheduled, advance a time step and then move on to step 1; otherwise, move to step 3.

Step 3. Develop a scheduling scheme, when a rental point is determined; V needs to be updated. The scheme includes the scheduling sequence and the scheduling number of each rental point S_i.

Step 3.1. The construction of the solution and the generation of the initial group.

A string of genes is used to represent a group of bike-sharing rental point scheduling schemes. Each gene is a positive integer variable of length N, where each represents the order in which the corresponding alternative locations are accessed, such as {3, 6, 1, 2, 5, 4}. According to the structure of the solution, L individuals are randomly generated to form the initial group, which is denoted as $G_0 = \{g_1, g_2, \cdots, g_L\}$.

Step 3.2. Adaptive Equation

The adaptive equation reflects the function of the individual's proximity to the optimal solution in the optimization calculation, and the adaptability equation for multiobjective problems can be directly objective equation (11) and (13).

Step 3.3. Selection

The selection of Pareto genetic algorithm is based on league selection of league-size 2, using a dynamically

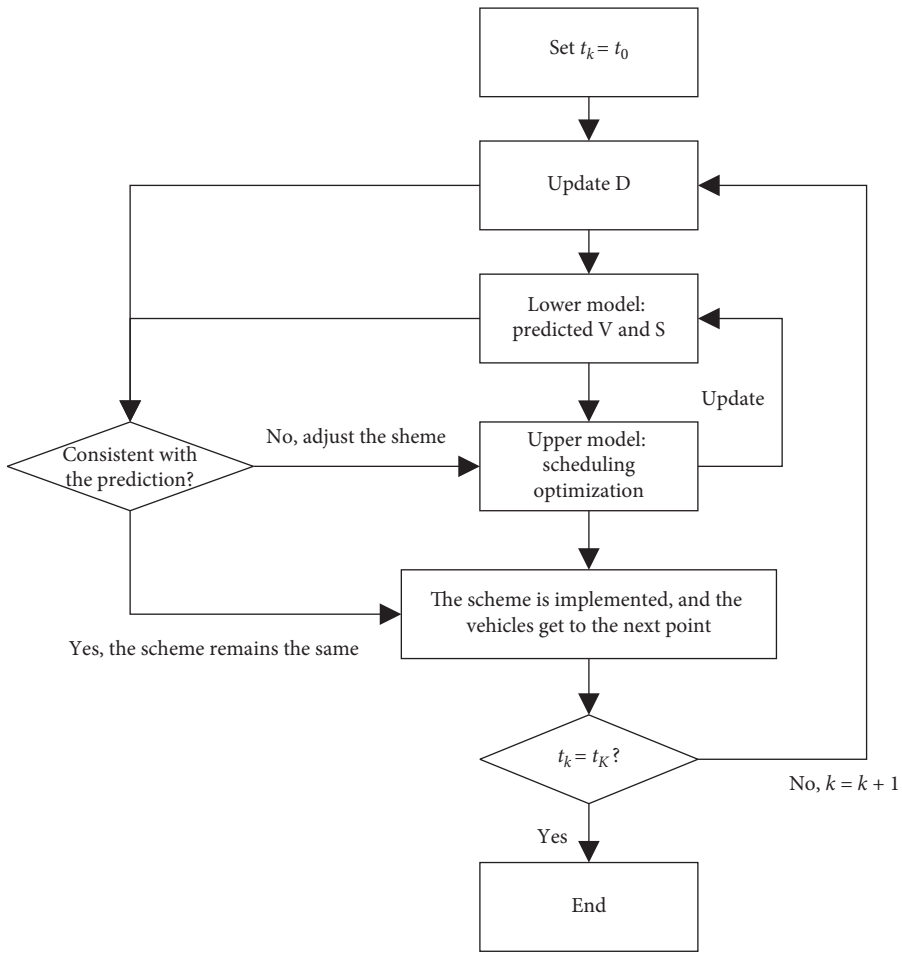


Figure 2: Simulation flow chart of bike-sharing dynamic scheduling system.

updated niche strategy. In league selection, two individuals are randomly selected from a parental population. Compare them with subpopulations originally selected from the population. The two individuals are superior to everyone in the subpopulation. If one of the individuals is not inferior to all individuals of this subpopulation and the other is at least inferior to one individual of this subpopulation, then select the former, and if all of them are inferior to at least one individual of this subpopulation or none of them are inferior to any individual of this subpopulation, then examine the two individuals in the current subpopulation. This method is done by placing the two individuals in separate subpopulations and calculating their totals; the individual with a smaller total number wins the game.

Step 3.4. Crossover

Since the structure of the solution is nonbinary, the method of selecting the crossing point and simply cross recombination cannot be possible. The specific method is to select the father individual and the mother individual from the population and randomly determine the crossing of the two parents, such as {3, 6, 1, 2, 5, 4} and {1, 2|3, 4|5, 6}, the cross-produced child is the part between the cross points of the mother, followed by the parent, and the repeated numbers in the child are removed, that is, {3, 4, 6, 1, 2, 5}.

Step 3.5. Mutations

The specific operation of the mutation is to select two of the mutated individuals and exchange their positions. For example, 1 and 2 of {3, 6, 1, 2, 5, 4} are mutated to {6, 3, 1, 2, 5, 4}.

Step 4. Scheduling Implementation

When a new rental point had been reached and the current time is recorded after loading and unloading, the scheduling is ended when $t = t_K$; otherwise, go to step 5.

Step 5. Check whether the valid bikes at each rental point are consistent with the predicted value, and if consistent, delete the current point from the scheduling scheme and go to step 4; otherwise, return to step 1.

3. Case Study

A bike-sharing travel scenario in the peak hour is built to test the model. Assume that there are 8 bike-sharing rental points and 1 parking base in this area as shown in Figure 3,

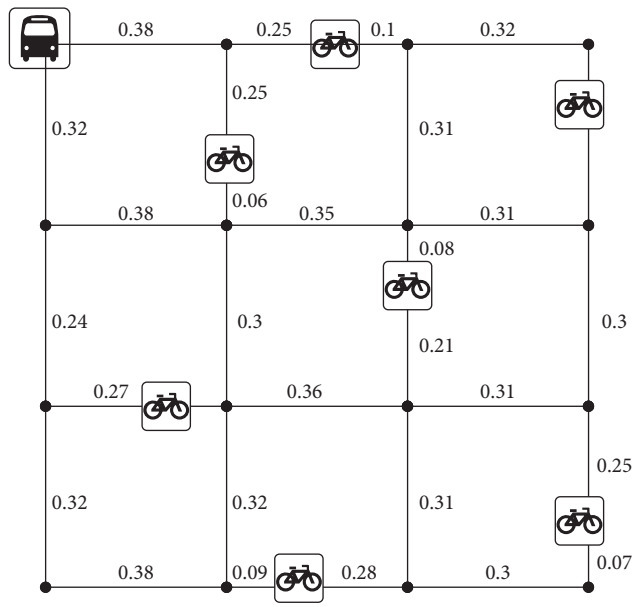


Figure 3: Location of bike-sharing rental points and parking lot.

and the numbers in this figure mean the length of each link and the unit is kilometers. The simulation starts at 7:00 am and ends at 9:00 am with a time interval of 5 minutes. The capacity of each rental point, the number of initial bikes, and the demand by every time interval are shown in Table 1, and the probability distribution of bike-sharing travel destinations is shown in Table 2. The capacity of the parking base is infinity. The rental point will be added to the scheduling scheme if the bikes are all gone or exceed the capacity when the vehicle arrives.

One scheduling vehicle departs from the parked base and eventually returns. The maximum loading is 300, and each rental point is scheduled only once. Regardless of the influence of road traffic, the speed of walking, bike-sharing, and scheduling vehicle is 5 km/h, 8 km/h, and 30 km/h, respectively, and the average loading and unloading time is 3 seconds per bike. The bikes for each period without scheduling are shown in Table 3.

The bikes at points 1 and 5 exceed the capacity, and at points 3, 4, 7, and 8 the bikes are empty. Only points 2 and 6 do not need to be scheduled.

The multiobjective optimization model with the lowest cost and minimum invalid demand was established by using the Niche Pareto genetic algorithm. Suppose that the population size is 50, the maximum evolutionary algebra is 100, the probability of crossover is 0.9, the probability of mutation is 0.04, the waiting penalty factor β_1 is 1, the transfer penalty factor β_2 is 2, and the abandoning penalty factor β_3 is 3. Taking the product of the objective functions (19) and (21) as the optimal solution, after multiple rounds of iteration (the evolution process is shown in Figure 2), the optimal scheduling scheme is 7-4-6-8-3-1-5-2.

As shown in Figure 4, the overall evolution obtained the optimal solution for the first time in the 20th generation and remained stable in the 49th generation.

Table 1: Capacity, initial number, and demand of bike-sharing.

Point	C_i	$V_i^{t_0}$	$D_i^{t_k}$ (random)
1	100	45	1–5
2	100	50	1–5
3	120	65	6–10
4	110	50	6–10
5	120	55	1–5
6	80	40	1–5
7	90	45	6–10
8	70	65	6–10

Table 2: Travel destination selection probability of bike-sharing.

Point	1	2	3	4	5	6	7	8	Sum
1	0	0	0	0	0.2	0.2	0.3	0.3	1
2	0	0	0	0	0.5	0.1	0.2	0.2	1
3	0	0	0	0	0.1	0.5	0.3	0.1	1
4	0	0	0	0	0.2	0.2	0.1	0.5	1
5	0.2	0.5	0.1	0.2	0	0	0	0	1
6	0.3	0.1	0.4	0.2	0	0	0	0	1
7	0.1	0.1	0.3	0.5	0	0	0	0	1
8	0.2	0.4	0.2	0.2	0	0	0	0	1

Figure 5 reflects the Pareto frontier corresponding to the optimal solution set of the double objective function, and the weights can be further clarified according to valid demand.

The bikes at each rental point under the optimal scheme are shown in Table 4, and the records of vehicle are shown in Table 5.

It can be seen from Table 3 to Table 5 that, after the scheme is adopted, the vehicle unloads bikes at points 3, 4, 7, and 8 and loads bikes at points 1 and 5. Most of the time, there will be a bike to borrow and a place to return. Through three rounds of effective scheduling which takes 96 minutes, the invalid demand is greatly reduced from 1094 to 26, and the bike-sharing resource allocation is better.

Table 3: Numbers of bikes for each period without scheduling.

Point	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1	41	40	43	51	61	69	76	86	91	100	103	110	111	115	113	111	112	114	117	117	117	117	116	120
2	47	45	46	47	48	48	48	49	49	51	53	54	55	55	56	52	49	49	49	48	45	42	41	41
3	56	48	43	40	36	35	33	32	32	30	23	21	19	13	5	0	0	0	0	0	0	0	0	0
4	44	36	33	27	25	23	20	14	8	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	52	56	63	71	79	89	96	106	113	122	127	139	143	150	156	162	165	168	172	170	175	180	182	184
6	38	38	40	40	38	41	40	41	44	45	47	47	48	49	47	48	47	49	45	45	46	46	45	42
7	38	31	29	25	19	14	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	55	47	40	35	27	22	15	9	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

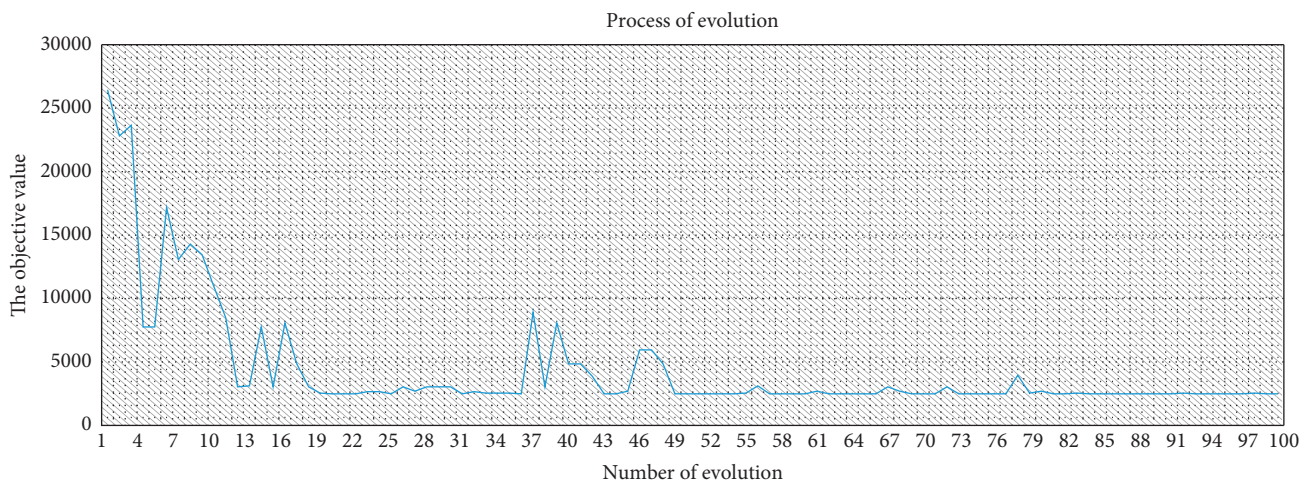


Figure 4: Process of evolution.

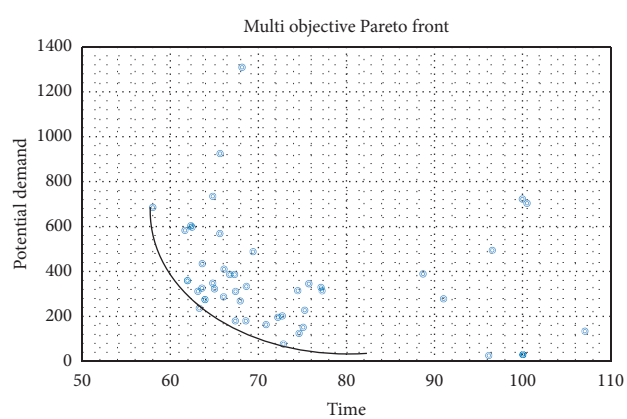


Figure 5: Pareto frontier.

Table 4: Numbers of bikes for each period in the optimal scheduling scheme.

Point	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1	41	40	43	51	61	69	76	86	91	101	106	116	121	9	12	18	24	32	40	44	48	54	58	67
2	47	45	46	47	48	48	48	49	49	51	53	55	58	61	63	61	61	62	63	64	63	62	62	63
3	56	48	43	40	36	35	33	32	32	31	27	146	145	142	138	134	130	128	122	117	117	115	110	108
4	44	36	33	27	25	133	130	124	118	113	111	108	106	99	96	89	82	79	73	66	63	59	53	46
5	52	56	63	71	79	89	96	106	113	122	127	139	143	152	161	171	178	190	17	23	35	47	54	64
6	38	38	40	40	38	41	40	41	44	45	47	47	48	51	50	53	54	58	58	60	64	67	68	67
7	38	31	29	25	109	104	96	90	85	78	75	70	66	59	54	51	47	45	42	36	34	27	23	18
8	55	47	40	35	27	22	85	79	74	66	60	52	44	36	28	23	19	17	11	7	4	0	0	0

Table 5: Records of vehicle in the optimal scheduling scheme.

Scheduling ID	Time	Point	Loaded
1	15	9	300
1	17.217	7	300
1	21.717	7	210
1	23	4	210
1	28.5	4	100
1	30.117	8	100
1	33.617	8	30
1	38.34	9	30
2	51.84	9	300
2	52.792	3	300
2	58.792	3	180
2	61.285	1	180
2	67.285	1	300
2	68.848	9	300
3	83.848	9	0
3	85.413	5	0
3	94.613	5	184
3	96.177	9	184

4. Conclusion

Faced with the practical problem that the bike-sharing is difficult to borrow and return due to the lack of effective scheduling, this paper establishes a dynamic scheduling model of bike-sharing with demand prediction and scheduling optimization and proposes a model solution method based on NPGA. The model is validated by a case study. By optimizing the scheduling, the attitude of waiting, transfer, and abandoning by users when they cannot borrow a bike will be significantly reduced. The results show that rational dynamic scheduling can effectively optimize the resource allocation and improve the service level of system.

In order to simplify the model, some practical problems were not considered, including the prediction of demand, the uncertainty of scheduling time, the gender, age, travel purpose, and other personal characteristics of the users into the waiting, transfer, and abandoning probability functions. These issues will be the focus of future research.

Data Availability

The experimental data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the Fundamental Research Funds for the Central Universities (30919011290 and 30920010010) and the Natural Science Foundation of Jiangsu Province, China (BK20171426).

References

[1] Cheetah Lab, *Bike-sharing Global Development Report*, China, 2018, https://www.chinanews.com/cj/2018/03-07/8462117.shtml.

[2] He Liu, X. Li, and D. Chen, "An optimization model of the layout of public bike rental stations based on B+R mode," *Lecture Notes in Electrical Engineering*, vol. 277, pp. 1341–1348, 2014.

[3] S. C. Ho and W. Y. Szeto, "Solving a static repositioning problem in bike-sharing systems using iterated tabu search," *Transportation Research Part E: Logistics and Transportation Review*, vol. 69, pp. 180–198, 2014.

[4] M. Dell'Amico, E. Hadjicostantinou, M. Iori, and S. Novellani, "The bike sharing rebalancing problem: mathematical formulations and benchmark instances," *Omega*, vol. 45, pp. 7–19, 2014.

[5] J. Brinkmann, M. W. Ulmer, and D. C. Mattfeld, "Short-term strategies for stochastic inventory routing in bike sharing systems," *Transportation Research Procedia*, vol. 10, pp. 364–373, 2015.

[6] L. Liu, L. Sun, Y. Chen, and X. Ma, "Optimizing fleet size and scheduling of feeder transit services considering the influence of bike-sharing systems," *Journal of Cleaner Production*, vol. 236, 2019.

[7] L. Shi, Y. Zhang, W. Rui, and X. Yang, "Study on the bike-sharing inventory rebalancing and vehicle routing for bike-sharing system," *Transportation Research Procedia*, vol. 39, pp. 624–633, 2019.

[8] T. Bulhões, A. Subramanian, G. Erdoğan, and G. Laporte, "The static bike relocation problem with multiple vehicles and visits," *European Journal of Operational Research*, vol. 264, 2018.

[9] J. Schuijbroek, R. C. Hampshire, and W.-J. Van Hoeve, "Inventory rebalancing and vehicle routing in bike sharing systems," *European Journal of Operational Research*, vol. 257, no. 3, pp. 992–1004, 2017.

[10] L. Caggiani, R. Camporeale, M. Ottomanelli, and W. Y. Szeto, "A modeling framework for the dynamic management of free-floating bike-sharing systems," *Transportation Research Part C: Emerging Technologies*, vol. 87, pp. 159–182, 2018.

[11] H. Guan, S. Lu, and M. Song, "Hierarchical scheduling strategy for free-floating bike-sharing," *Journal of Chongqing Jiaotong University (Natural Science)*, vol. 39, no. 02, pp. 1–7, 2020.

[12] A. K. Ahmed, K. Imed, and L. Karim, *Lower and Upper Bounds for Scheduling Multiple Balancing Vehicles in Bicycle-Sharing Systems*, Springer Berlin Heidelberg, vol. 23, no. 14, Berlin, Germany, 2019.

[13] V. Pureza, R. Morabito, and M. Reimann, "Vehicle routing with multiple deliverymen: modeling and heuristic approaches for the VRPTW," *European Journal of Operational Research*, vol. 218, no. 31, pp. 636–647, 2012.

[14] H. Ben Ticha, N. Absi, D. Feillet, and A. Quilliot, "Empirical analysis for the VRPTW with a multigraph representation for the road network," *Computers & Operations Research*, vol. 88, pp. 103–116, 2017.

[15] M. Harzi and S. Krichen, "Variable neighborhood descent for solving the vehicle routing problem with time windows," *Electronic Notes in Discrete Mathematics*, vol. 58, pp. 175–182, 2017.

[16] F. Hernandez, D. Feillet, R. Giroudeau, and O. Naud, "Branch-and-price algorithms for the solution of the multi-trip vehicle routing problem with time windows," *European*

Journal of Operational Research, vol. 249, no. 2, pp. 551–559, 2016.

[17] D. M. Pierre and N. Zakaria, "Stochastic partially optimized cyclic shift crossover for multi-objective genetic algorithms for the vehicle routing problem with time-windows," *Applied Soft Computing*, vol. 52, pp. 863–876, 2017.

[18] D. Ma, J. Xiao, X. Song, X. Ma, and S. Jin, "A back-pressure-based model with fixed phase sequences for traffic signal-optimization under oversaturated networks," *Institute of Electrical and Electronics Engineers Transactions on Intelligent Transportation Systems*, vol. 2020, 12 pages, 2020.

[19] D. Ma, X. Song, and Li Pu, "Daily traffic flow forecasting through a contextual convolutional recurrent neural network modeling inter-and intra-day traffic patterns," *Institute of Electrical and Electronics Engineers Transactions on Intelligent Transportation Systems*, vol. 2020, 10 pages, 2020.

[20] K. Braekers, K. Ramaekers, and I. Van Nieuwenhuyse, "The vehicle routing problem: state of the art classification and review," *Computers & Industrial Engineering*, vol. 99, pp. 300–313, 2016.

Hindawi
Journal of Advanced Transportation
Volume 2020, Article ID 8843513, 12 pages
https://doi.org/10.1155/2020/8843513

Research Article

Collaborative Control Optimization on Reversible Lanes in Intersection Group under Intelligent Vehicle Infrastructure Cooperative System

Guiliang Zhou,[1] Lina Mao,[1,2] Yuke Dai,[1] Yao Liu,[3] Xu Bao,[1] and Pengsen Hu[4]

[1]*Jiangsu Key Laboratory of Traffic and Transportation Security, Huaiyin Institute of Technology, Huaian 223003, China*
[2]*School of Transportation, Southeast University, Nanjing 210096, China*
[3]*Faculty of Management Engineering, Huaiyin Institute of Technology, Huaian 223003, China*
[4]*Department of Civil and Environmental Engineering, The Pennsylvania State University, State College, PA 16802, USA*

Correspondence should be addressed to Guiliang Zhou; zglpaper@qq.com and Lina Mao; 30721409@qq.com

Received 24 July 2020; Revised 8 September 2020; Accepted 14 October 2020; Published 5 November 2020

Academic Editor: Zhuping Zhou

Copyright © 2020 Guiliang Zhou 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.

Based on the Intelligent Vehicle Infrastructure Cooperative System (IVICS), the paper firstly analyzes the operating characteristics of the intersection group under real-time reversible lanes from the capacity; secondly, the signal phase sequence and signal timing parameters of the intersection are designed, and the green wave control is designed based on the phase difference model. The scheme provides traffic flowing in this section with a continuous green light signal. Finally, taking the intersection of Yongle East Road in Wuxi, Jiangsu Province as an example, it optimizes and controls the green wave of the intersection group and verifies the effectiveness of the phase difference model using the evaluation indicators of the number of stops, green wave speed, traffic capacity, and saturation flow. The results show that the optimization of the number of stops is 9.5% and the vehicle speed is optimized by 5.3%. The intersection capacity and saturation flow are greatly improved.

1. Introduction

Intersections in the urban road network are interacting. In order to achieve efficient and smooth traffic flow, this paper designs a collaborative control optimization method for real-time reversible lane intersection groups under the Intelligent Vehicle Infrastructure Cooperative System (IVICS). Lots of researches have been conducted on reversible lanes at home and abroad. Michael W. Levin and Stephen Levin and Boyles [1] proposed a cell reversal model based on dynamic reversible lanes; based on scheduled intersection control and dynamic variable lanes, they formed a dynamic variable lane control using a single link as a complete program. Assi and Ratrout [2] determined the geometric characteristics that affect the operation of double left-turn lanes. Wael KM Alhajyaseen et al. [3] analyzed the application effect of dynamic lane allocation on intersection paths and established a model combining dynamic lane grouping and the entire intersection signal timing parameters optimization to improve the capacity of the intersection. Assi and Ratrout [2] proposed a method for the rapid allocation of dynamic variable lanes at intersections and this method can use the percentage of turning movement when the traffic enters the entrance of the intersection to predict the best lane group on the spot. Dey et al. [4] discussed the application of reversible lanes on the main road in Washington, DC, and discussed the inherent constraints of the city's external environment and the operational constraints imposed by external stakeholders.

Cheng et al. [5] considered the impact of inbound and outbound traffic on intersection coordination and established a dynamic phase difference model for the intersection. Qu et al. [6, 7], based on the theory of vehicle flow fluctuations, established the intersection phase difference calculation model for the assembly and dissipation of queued vehicles, taking into consideration the interactions

between intersections and the relationship between traffic demand and supply, and further built a phase difference optimization model from six different conditions of vehicle operation. Cao [8] used traffic overflow and green light empty as the constraints of the phase difference model and adjusted and optimized the phase difference according to whether the green light-on time of two driving directions at the same intersection is synchronized. Jiao [9] used saturation degree as the direction-change switch condition of reversible lanes—if the saturation degree is greater than 0.9, turn on the direction-change switch—and constructed a dynamic prediction model of intersection traffic flow to predict all directions of car traffic in the next few cycles to determine whether the saturation threshold condition is met. Liu et al. [10] realized the optimization of the intersection signal period and the green signal ratio by establishing the dissipation flow rate model for the interval of the left-turn phase green light signal. Liu et al. [11] constructed a multiobjective optimization model based on the NSGA-II algorithm, using the maximum capacity of intersections and the smallest average delay of vehicles as the objective function to solve the signal timing parameter optimization under reversible lanes. Jiang [12] proposed a method of signal phase sequence combination at each time interval of intersections based on traffic flow indicators. Zeng et al. [13] obtained the scheme of signal timing parameter design at each time interval of intersections in units of time intervals.

At present, the research on the theory of reversible lanes is relatively mature. Scholars have done a lot of research works on the reversible lane change model of intersections, the traffic characteristics of intersections, and the analysis of capacity, but the collaborative control of reversible lanes of intersection groups still needs further research. Specifically, (1) at present, there are few studies on the reversible lanes of intersection groups, and most of the research objects are either single intersection or reversible lanes of main roads, which ignores the connections between intersections; (2) research on the threshold conditions for the change of the attribute of the reversible lanes at the intersection is not enough. Currently, most studies have failed to consider the threshold of the change of the attribute of the reversible lanes under different conditions.

2. The Characteristics of Effects of Real-Time Reversible Lanes on the Traffic Flow of Intersection Groups

The setting of real-time reversible lanes greatly improves the crossing capacity of the intersection. In terms of traditional reversible lanes, on the one hand, reversible lanes can increase the capacity of the corresponding phase; on the other hand, if improperly set, it would affect vehicles passing through intersections, reducing the efficiency of vehicles passing through intersections and causing safety problems. The real-time reversible lanes, based on the IVICS, change the direction of traffic flow at intersections, which greatly avoids the waste of road resources.

The capacity is related to the saturation flow and the green signal ratio, and it is generally calculated using the product of the saturation flow and the green signal ratio [14]. Suppose there are m_i straight lanes and n_i left lanes for the i-th intersection of the intersection group. The capacity of the intersection group in the real-time reversible lane setting of the IVICS is as follows.

The capacity of the i-th intersection: suppose that, at the i-th intersection, there are x_i lane attributes changed, x_{il} represents the number of lanes where the straight direction of the i-th intersection changes to the left-turn direction, x_{is} represents the number of lanes where the left-turn direction of the i-th intersection changes to the straight direction, and $x_{il}, x_{is} >= 0$.

The left-handed phase capacity and saturation flow of the i-th intersection are as follows:

$$C'_{il} = \left(n_i + x_{il} - x_{is}\right) \cdot S'_{il} \cdot \frac{g'_{il}}{c'_i},$$
$$S'_{is} = \frac{m_i + x_{is} - x_{il}}{m_i} \cdot S_{is}, \tag{1}$$

where C'_{il} is the left-hand phase capacity of the i-th intersection after setting reversible lanes; S'_{il} is the left-hand phase saturation flow of the i-th intersection after setting reversible lanes; g'_{il} is the effective green light duration of the left-hand phase of the i-th intersection after setting variable settings; c'_i is the signal period of the i-th intersection after the variable lane is set.

The straight-through phase capacity and saturation flow of the i-th intersection are as follows:

$$C'_{is} = \left(m_i + x_{is} - x_{il}\right) \cdot S'_{is} \cdot \frac{g'_{is}}{c'_i},$$
$$S'_{is} = \frac{m_i + x_{is} - x_{il}}{m_i} \cdot S_{is}, \tag{2}$$

where C'_{is} is the straight-through phase capacity of the i-th intersection after the variable lane is set; S'_{is} is the saturation flow of the straight-through phase after the variable lane is set at the i-th intersection; g'_{is} is the effective green light duration of the straight phase after the variable lane is set at the i-th intersection. Therefore, the capacity of the i-th intersection after setting the variable lanes is $C'_i = C'_{il} + C'_{is}$.

The capacity of the intersection group is

$$C'_0 = \sum_{i=1}^{k} C'_i. \tag{3}$$

3. Collaborative Optimization of Real-Time Reversible Lanes and Intersection Group Signal

The traditional signal phase scheme fails to take into account the real-time changes in the traffic flow. The fixed signal phase sequence causes a waste of traffic resources during off-peak hours and traffic congestion during peak hours. The

general reversible lane signal phase scheme only changes the phase sequence in a fixed time period, which has certain effects on alleviating tidal traffic congestion, but it is difficult to play an effective role in congestion at other times. The real-time reversible lanes of the IVICS consider the real-time running status of the traffic flow and perform real-time collaborative control of the traffic flow at the intersection to improve the efficiency of the traffic flow.

3.1. Signal Phase Sequence Design. The signal phase generally has the following forms [15]: (1) symmetrical release; that is, vehicles in the two driving directions of the intersection synchronously go straight and turn left synchronously, which is the most common signal phase; (2) single-port release; that is, the vehicles in the two driving directions at the intersection are staggered to go straight and turn left. The one direction goes straight, while the other direction stops and waits and then passes after it ends; (3) overlap release; that is, a certain phase is straight, and then the next phase is straight and turn left. These phases are shown in Figures 1–3, respectively.

3.2. Signal Timing Parameter Design. Under the low saturation of the intersection, Webster's best signal cycle model $c_0 = ((1.5L + 5)/(1 - Y))$ is mainly used to calculate the signal cycle [7], where c_0 is the signal cycle, L is the total loss time of the signal cycle, and Y is the sum of the critical phase lane flow ratio; that is, $Y = \sum q_j/S_j$. After setting the real-time dynamic reversible lanes, the flow ratio between straight and left direction will change, so we need to make some changes to the best signal cycle model, specifically as follows.

The change of the flow ratio of the left-turn phase after the real-time dynamic reversible lane is implemented in the IVICS is

$$Y_L' = \frac{q_l}{S_l'} = \frac{q_l}{((n + x_l - x_s)/n) \cdot S_l} = \frac{n}{n + x_l - x_s} \cdot Y_L, \quad (4)$$

where q_l is the vehicle arrival rate of the left-turn phase of the single intersection, S_l' represents the saturation flow of left-turn phase of the intersection after the reversible lane is implemented, and S_l represents the saturation flow of the left-turn phase of the intersection before the real-time reversible lane is implemented.

After the real-time dynamic reversible lanes are implemented in IVICS, the flow ratio of the straight-through phase becomes

$$Y_S' = \frac{q_s}{S_s'} = \frac{q_s}{((m + x_s - x_l)/m) \cdot S_s} = \frac{m}{m + x_s - x_l} \cdot Y_S, \quad (5)$$

where q_s is the vehicle arrival rate of the left-turn phase of a single intersection, S_s' represents the saturation flow of the left-turn phase of the intersection after the reversible lane is implemented, and S_s represents the saturation flow of the left-turn phase of the intersection before the real-time reversible lane is implemented.

Since the key phase is still straight and left turn after setting the reversible lane, the L value remains. Then, the calculation formula of the optimal signal period of the intersection after setting real-time dynamic reversible lanes under the IVICS is

$$c_0' = \frac{1.5L + 5}{1 - (Y_L' + Y_S')} = \frac{1.5L + 5}{1 - (Y - (n/(n + x_l - x_s)) \cdot Y_L - (m/(m + x_s - x_l)) \cdot Y_S)}. \quad (6)$$

At this time, the key phase of the intersection is the sum of the straight-forward and left-turn phases:

$$Y' = Y - Y_L' - Y_S' = Y - \frac{n}{n + x_l - x_s} \cdot Y_L - \frac{m}{m + x_s - x_l} \cdot Y_S. \quad (7)$$

After setting the real-time reversible lanes, the effective green light time of the left-turn phase and the straight-forward phase is, respectively, as follows: the effective green light time of the left-turn phase $g_l' = (c_0' - L) \cdot Y_L'/Y'$ and the effective green light time of the straight-forward phase $g_s' = (c_0' - L) \cdot Y_S'/Y'$.

Under the high saturation of the intersection, the best signal period calculation formula $c = ((1.5L + 5)/(1 - 0.9))$ is used to calculate the signal period. The calculation formula of the effective green light duration is similar to that of the low saturation.

3.3. Collaborative Optimization of Space-Time Resources of Intersection Group. The purpose of establishing the collaborative optimization model of the space-time resources of the intersection group under the IVICS is to improve the intersection group's passing capacity. The key to improving the intersection group's passing capacity lies in the critical path; by giving the "green wave" signal to the heavy traffic flow direction, vehicles in the direction of heavy traffic can get a continuous green light signal and pass through all intersections on the road without hindrance. The traditional green wave cooperative control method is mainly a graphical method. This method is difficult to obtain a two-way green wave scheme, the resulting green wave bandwidth is relatively narrow, and the green wave speed is relatively low, which is difficult to meet the designed speed required by the main traffic line. In this paper, the green wave control is realized by establishing the calculation model of the phase difference between the intersections on the mainline.

Qu et al. [6] analyzed the theory of vehicle flow fluctuations from the perspective of kinematics and obtained the wave velocity equations for the propagation of parking and starting waves. The wave velocity formulas are, respectively,

$$v_b = -\frac{v_0}{h_0 \cdot k_j \cdot v_0 - 1}, \qquad v_d = -\frac{v_1}{h_1 \cdot k_j \cdot v_1 - 1}. \quad (8)$$

Here, v_b is the speed of the parking wave, v_d is the starting wave speed, v_0 is the initial driving speed of the vehicle, v_1 is the driving speed of the vehicle after the queue

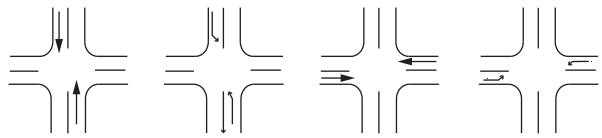


Figure 1: Symmetrical release phase.

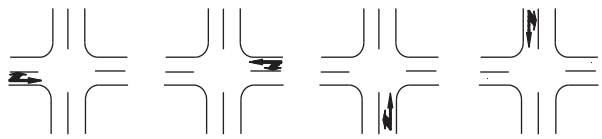


Figure 2: Single-port release phase.

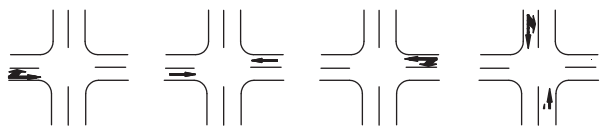


Figure 3: Overlap release phase.

is dissipated, k_j is the blocking density (the density of the traffic that is too dense to drive normally), h_0 is the initial headway, and h_1 is the headway after vehicle queue is dissipated. It can be seen from the trajectory diagram of the traffic flow that the position of the longest queue length is the position where the parking wave meets the starting wave.

3.3.1. Phase Difference Calculation. Let $L_{i-1.i}$ be the distance between the i-1th intersection and the i-th intersection in the intersection group and $v_{i-1,i}$ be the travel speed between the i-1th intersection and the i-th intersection in the intersection group; then the time required for the vehicle to pass through these two intersections is $t_{i-1,i} = L_{i-1,i}/v_{i-1,i}$. The number of vehicles in the queue can be calculated:

$$N_{i-1,i} = \sum_{j=1}^{2} K_{i,j} \cdot L_{i,j}, \tag{9}$$

where $K_{i,j}$ is the traffic density of the j-th phase of the i-th intersection and $L_{i,j}$ is the length of the queue ($j = 1$, representing the straight phase; $j = 2$, representing the left-turn phase). The queuing time of traffic flow at the intersection is the sum of the queuing formation time and the queuing dissipation time. The queuing vehicle dissipation time is $t_{i-1,i} = N_{i-1,i}/C$, where C is the capacity of the intersection, and the queuing formation time is the time period when the first vehicle starts queuing and sends the parking wave from the stopping line backward to the following vehicles until it restarts and sends the starting wave backward. The phase difference between the intersection i-1 and the intersection i is

$$\phi_{i-1,i} = \frac{L_{i-1,i}}{v_{i-1,i}} - \frac{N_{i-1,i}}{C}. \tag{10}$$

3.3.2. Optimization of Phase Difference Model. The optimization of the phase difference model in the IVICS is mainly to enable vehicles with different speeds to pass through all intersections along the way without queuing and parking. If the vehicle speed is lower than the target speed, the intelligent vehicular facility guides to increase its speed to be the speed to pass the intersection. If the vehicle speed is higher than the target speed, then it guides to reduce the vehicle speed to be the speed to pass the intersection without queuing. According to the definition of phase difference [16], the phase difference is the time difference between the red and green lights on the upstream and downstream intersections. Therefore, the optimization of the phase difference is mainly from the time of the green or red lights between the upstream and downstream intersections. According to the three methods of green wave cooperative control, there are three main ways to turn on the signal lights between intersections: the signal lights between the upstream and downstream intersections are on at the same time, opposite, and mixed. The phase difference optimization formulas in the three cases are derived similarly, so the

phase difference optimization is mainly discussed in the case that the signal of the intersection is synchronized.

When the signal lights of the upstream and downstream intersections are synchronized and the signal lights of the upstream and downstream intersections are simultaneously red, the traffic flow of the downstream intersections starts forming a queue from the parking line and forms a parking wave to pass to the upstream intersection. Meanwhile, the upstream intersection turns into a red light, causing the traffic to cut off and the vehicle cannot pass through. Therefore, when the parking wave reaches the end of the queue, the queue length is the longest and remains for a period of time. Its vehicle trajectory is shown in Figure 4. Let $L_D^{i-1,i}$ be the distance from the detector to the downstream intersection; let $L_{\max}^{i-1,i}$ be the maximum queuing length; $t_A^{i-1,i}$ is the time when the parking wave transmits to the detector; $t_B^{i-1,i}$ is the time when the starting wave is transmitted to the detector; $t_C^{i-1,i}$ is the time when the vehicle at the end of the queue travels to the detector when the vehicle flow dissipates; $L_{DE}^{i-1,i}$ is the distance from the detector to the tail of the traffic queue; $T_{\max}^{i-1,i}$ is the moment when the maximum queuing length is formed; $v_b^{i-1,i}$ is the speed of the parking wave due to parking when the vehicle reaches the parking line; $v_d^{i-1,i}$ is the speed of the starting wave when the queuing vehicle starts moving from the parking line; $v_q^{i-1,i}$ is the speed of the vehicle when it is queued to dissipate; $v_d^{i-1',i}$ is the speed of the starting wave of the next cycle; $T_R^{i-1,i,n}$ is the moment when the red light turns on in the n-th cycle; $T_G^{i-1,i,n}$ is the moment when the green light turns on in the n-th cycle; $T_R^{i-1,i,n+1}$ is the moment when the red light of the n+1-th cycle turns on.

Suppose that the moment when the vehicle queued at the tail of the traffic flow moves to the parking line of the downstream intersection is $T_T^{i-1,i}$; then two situations may occur when the vehicle at the tail of the queue reaches the parking line. The first is $T_T^{i-1,i} <= T_R^{i-1,i,n+1}$; that is, the vehicle can pass through the parking line before the red light turns on at the next cycle, and then all the queued vehicles can pass through the intersection. The second case is $T_T^{i-1,i} > T_R^{i-1,i,n+1}$. If the queued vehicles cannot fully pass through the intersection, a secondary queue or stranded queue will be formed, and the queue length is $L_{\min}^{i-1,i}$.

It can be seen from the queuing time interval that

$$L_{\max}^{i-1,i} = L_D^{i-1,i} + L_{DE}^{i-1,i} \tag{11}$$

and

$$\begin{cases} v_d^{i-1,i} = \dfrac{L_{DE}^{i-1,i}}{T_{\max}^{i-1,i} - t_B^{i-1,i}}, \\ v_q^{i-1,i} = \dfrac{L_{DE}^{i-1,i}}{t_C^{i-1,i} - T_{\max}^{i-1,i}}, \end{cases} \tag{12}$$

$$\frac{v_d^{i-1,i} \cdot v_q^{i-1,i}}{v_d^{i-1,i} +_q^{i-1,i}},$$

thus calculating

$$L_{DE}^{i-1,i} = \frac{v_d^{i-1,i} \cdot v_q^{i-1,i}}{v_d^{i-1,i} + v_q^{i-1,i}} \left(t_C^{i-1,i} - t_B^{i-1,i}\right), \tag{13}$$

and getting

$$L_{\max}^{i-1,i} = L_D^{i-1.i} + \frac{v_d^{i-1,i} \cdot v_q^{i-1,i}}{v_d^{i-1,i} + v_q^{i-1,i}} \left(t_C^{i-1,i} - t_B^{i-1,i}\right). \tag{14}$$

In addition, the moment when the vehicle at the end of the queue of the traffic flow passes the parking line of the lane is

$$T_T^{i-1,i} = T_{\max}^{i-1,i} + \frac{L_{\max}^{i-1,i}}{v_q^{i-1,i}},$$

$$\begin{cases} v_d^{i-1,1'} = \dfrac{L_{\min}^{i-1,i}}{T_{\min}^{i-1,i} - T_R^{i-1,i,n+1}}, \\ v_q^{i-1,i} = \dfrac{L_{\min}^{i-1,i}}{T_T^{i-1,i} - T_{\min}^{i-1,i}}, \end{cases} \tag{15}$$

$$\frac{v_d^{i-1,1'} \cdot v_q^{i-1,i}}{v_d^{i-1,1'} + v_q^{i-1,i}}.$$

The length of the stranded queue or the secondary queue is

$$L_{\min}^{i-1,i} = \frac{v_d^{i-1,1'} \cdot v_q^{i-1,i}}{v_d^{i-1,1'} + v_q^{i-1,i}} \left(T_T^{i-1,i} - T_R^{i-1,i,n+1}\right) = \frac{v_d^{i-1,1'} \cdot v_q^{i-1,i}}{v_d^{i-1,1'} + v_q^{i-1,i}} \left(T_{\max}^{i-1,i} + \frac{L_{\max}^{i-1,i}}{v_q^{i-1,i}} - T_R^{i-1,i,n+1}\right). \tag{16}$$

In order to prevent the traffic overflow phenomenon of queued traffic, the signal lights of the downstream intersection need to be kept for a period of time for the vehicles in the queue to dissipate. Therefore, there may be two extreme cases. The first is that the traffic from the upstream intersection crosses the intersection when the traffic light at the intersection turns green; no queuing happens when driving to the end of the queue at the downstream intersection. In this case, when the upstream traffic flows to the end of the queue, the parking wave and the starting wave at the downstream intersection meet; the phase difference needs to be met at this time: the traffic at the upstream intersection does not line up when passing through the downstream intersection; that is,

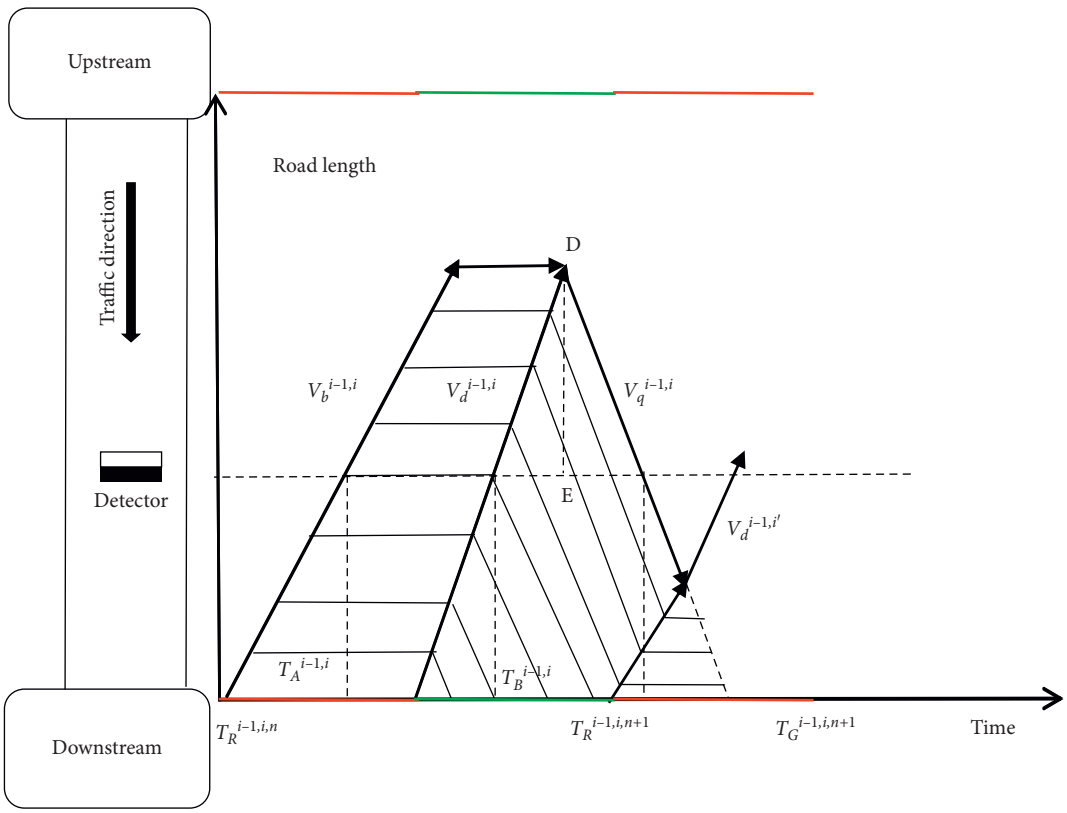


Figure 4: Vehicle trajectory when the intersection signals are synchronized.

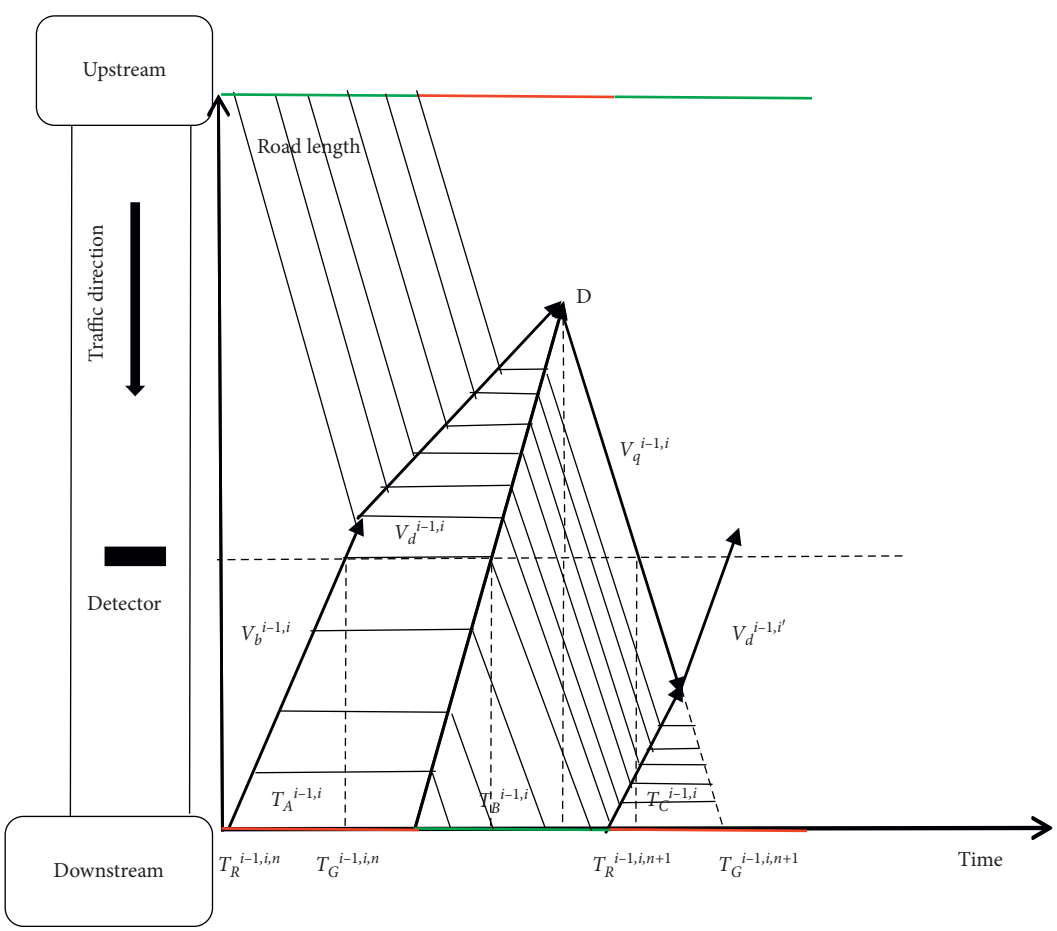


Figure 5: Vehicle trajectory when the intersection signals are controlled oppositely.

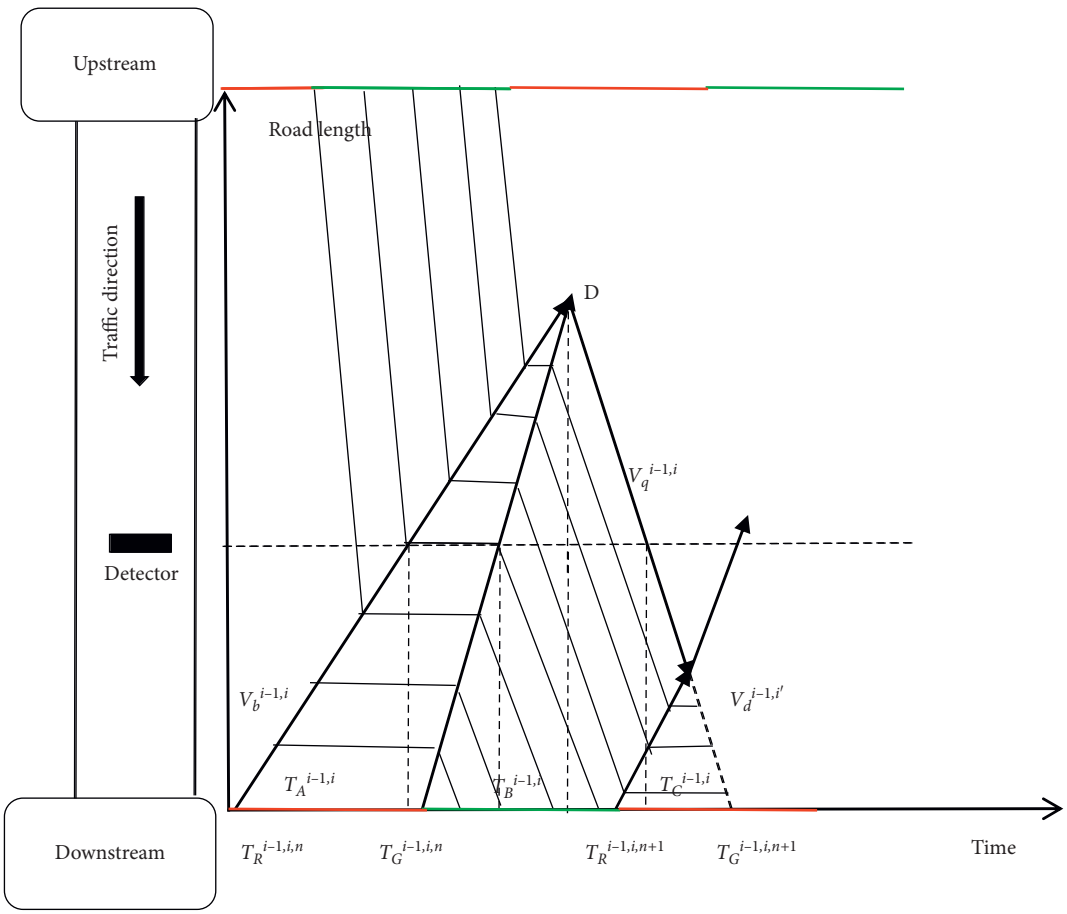


Figure 6: Vehicle trajectory when the intersection signals are mixed and controlled.

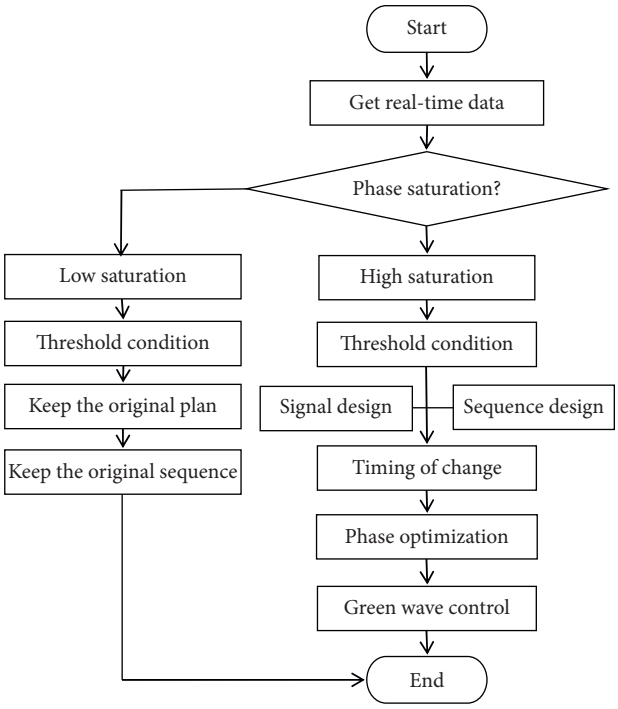


Figure 7: Chart of dynamic control.

$$\phi_{i-1,i} + \frac{L_{\max}^{i-1,i}}{v_d^{i-1,i}} = \frac{L_{i-1,i} + L_{i-1} - L_{\max}^{i-1,i}}{v_{i-1,i}}. \quad (17)$$

The second case is that when the traffic at the upstream intersection reaches the parking line at the downstream intersection, the vehicles at the end of the queue at the downstream intersection just pass through the parking line. At this time, the phase difference needs to be satisfied: the traffic at the upstream intersection just crosses the parking line at the downstream intersection; that is,

$$\phi_{i-1,i} + \frac{L_{\min}^{i-1,i}}{v_d^{i-1,i}} + \frac{L_{\min}^{i-1,i}}{v_q^{i-1,i}} = \frac{L_{i-1,i} + L_{i-1}}{v_{i-1,i}}. \quad (18)$$

We can know

$$\phi_{i-1,i} = \frac{L_{i-1,i} + L_{i-1}}{v_{i-1,i}} - \frac{L_{\min}^{i-1,i}}{v_d^{i-1,i}} - \frac{L_{\min}^{i-1,i}}{v_q^{i-1,i}}. \quad (19)$$

Therefore, when, under the IVICS, the traffic flow at the upstream intersection by getting a green light signal can pass through the downstream intersection without stopping, the phase difference needs to meet

$$\frac{L_{i-1,i} + L_{i-1} - L_{\max}^{i-1,i}}{v_{i-1,i}} - \frac{L_{\max}^{i-1,i}}{v_d^{i-1,i}} <= \phi_{i-1,i} <= \frac{L_{i-1,i} + L_{i-1}}{v_{i-1,i}} - \frac{L_{\min}^{i-1,i}}{v_d^{i-1,i}} - \frac{L_{\min}^{i-1,i}}{v_q^{i-1,i}}. \quad (20)$$

It can be seen from the three methods of green wave cooperative control that there are three main ways to turn on the signal lights between intersections. The derivation of the optimization formula of the phase difference is similar under the three situations of the same, opposite, and mixed interaction of the signal lights between the upstream and downstream intersections. The derivation of the phase difference optimization formula of the opposite and mixed interaction is omitted here. The vehicle trajectory when the intersection signal control is opposite is shown in Figure 5, and the vehicle trajectory when the intersection signal control is mixed is shown in Figure 6.

3.4. Real-Time Dynamic Control Flow of Reversible Lanes Intersection Groups. The attribute change of the real-time reversible lanes is determined by detecting the traffic volume on each phase entrance lane by the detectors set at each intersection. If the traffic at the intersection phase is not saturated, it means that the left-turn phase or the straightforward phase still has another phase of the available lane that is more congested; if the traffic at a certain phase of the intersection has reached saturation, then the intersection signal timing parameter design needs to be optimized. The specific process is as follows (see Figure 7):

(1) Obtain real-time traffic flow and queuing length data through the detector set at real-time reversible lane intersections.

(2) The information control center analyzes and processes these initial data to obtain key data such as flow ratio.

(3) Determine the phase saturation of the real-time reversible lane intersection and determine whether the intersection status is high or low saturation.

(4) Substitute the threshold condition to see if the condition for turning on the change switch of the real-time reversible lane is satisfied. If it is satisfied, the number of lanes and the time for turning on the real-time reversible lane change are determined; otherwise, the original signal timing scheme is maintained.

(5) The phase difference is optimized to realize green wave control.

4. Case Study

Taking Yongan East Road in Wuxi City, Jiangsu Province, as the research object, the selected intersections are the intersections of Tongyang Road and Yongle East Road, Tangnan Road and Yongle East Road, and Xingyuan Middle Road and Yongle East Road. Yongle East Road (Tongyang Road to Xingyuan Middle Road Section) is a two-way six-lane road. The signal phase sequence and signal timing parameters of the intersection are shown in Table 1. The capacity and saturation flow are shown in Tables 2 and 3. The intersection of Yongle East Road is now under collaborative control, and green wave control is designed to improve the operating efficiency of the Yongle East Road intersection. The optimized signal phase sequence and signal timing parameters are shown in Table 4.

Take the intersection of Tongyang Road and Yongle East Road as the reference intersection, and then calculate the phase difference of the intersection of Tangnan Road and Yongle East Road and the intersection of Xingyuan Middle Road and Yongle East Road with respect to the reference intersection to form green wave traffic. The phase difference between the intersections of Tongyang Road and Yongle East Road set as a reference intersection is zero; the phase difference of the intersection of Tangnan Road and Yongle East Road relative to the intersection of Tongyang Road and Yongle East Road is 38.2 seconds. The phase difference of the intersection of Xingyuan Middle Road and Yongle East with respect to the intersection of Tongyang Road and Yongle East Road is 52.8 seconds. The green wave control chart is shown in Figure 8. It can be seen from Figure 8 that the green wave bandwidth is 36 s, and it can be calculated that the average green wave speed is 37.9 km/h.

The evaluation indicators for the effect of the implemented green wave control are the number of stops, green wave bandwidth, green wave vehicle speed, capacity, and saturation flow. The effectiveness of the green wave control is verified by calculating and comparing the evaluation indexes of Yongle East Road before and after implementing the green wave control. See Table 5–7 for the comparison of the relevant indicators before and after optimization.

It can be seen from Table 5 that, after the signal timing parameter optimization of the intersection on Yongle East Road, the number of stops, at the intersection, has been reduced to a certain extent, with an average reduction of

Table 1: Signal phase sequence before optimization.

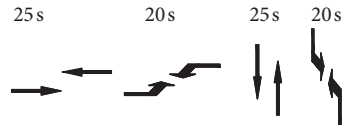
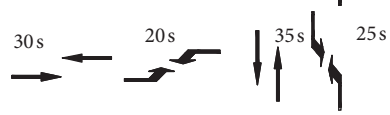
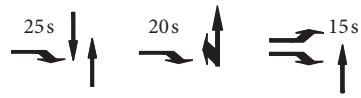
Intersection parameter	Signal phase sequence	Signal period
Intersection of Tongyang Road and Yongle East Road		90 s
Intersection of Tangnan Road and Yongle East Road		110 s
Intersection of Xingyuan Middle Road and Yongle East Road		60 s

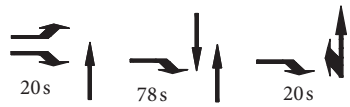
Table 2: Capacity.

Intersection parameter	Turn left	Straight	Turn right
Intersection of Tongyang Road and Yongle East Road	124	726	363
Intersection of Tangnan Road and Yongle East Road	348	830	415
Intersection of Xingyuan Middle Road and Yongle East Road	166	—	526

Table 3: Saturation flow.

Intersection parameter	Turn left	Straight	Turn right
Intersection of Tongyang Road and Yongle East Road	279	653	—
Intersection of Tangnan Road and Yongle East Road	822	653	—
Intersection of Xingyuan Middle Road and Yongle East Road	415	—	658

Table 4: Optimized signal phase sequence.

Intersection parameter	Signal phase sequence	Signal period
Intersection of Tongyang Road and Yongle East Road		122 s
Intersection of Tangnan Road and Yongle East Road		122 s
Intersection of Xingyuan Middle Road and Yongle East Road		122 s

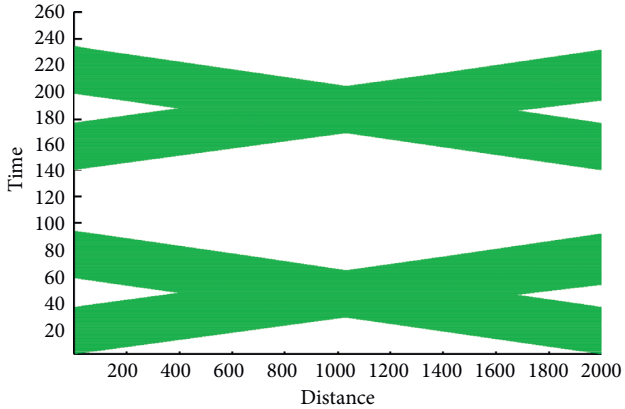


Figure 8: Green wave time interval.

Table 5: Number of stops before and after optimization.

Intersection status	Before optimization	Optimized
Intersection of Tongyang Road and Yongle East Road	163	124
Intersection of Tangnan Road and Yongle East Road	118	79
Intersection of Xingyuan Middle Road and Yongle East Road	56	102

Table 6: Optimized capacity at intersections.

Intersection parameter	Turn left	Straight	Turn right
Intersection of Tongyang Road and Yongle East Road	132	770	385
Intersection of Tangnan Road and Yongle East Road	350	832	416
Intersection of Xingyuan Middle Road and Yongle East Road	169	—	536

Table 7: Comparison of entrance capacity before and after optimization.

Intersection parameter	Before optimization	Optimized
Intersection of Tongyang Road and Yongle East Road	850	902
Intersection of Tangnan Road and Yongle East Road	1011	1182
Intersection of Xingyuan Middle Road and Yongle East Road	692	705

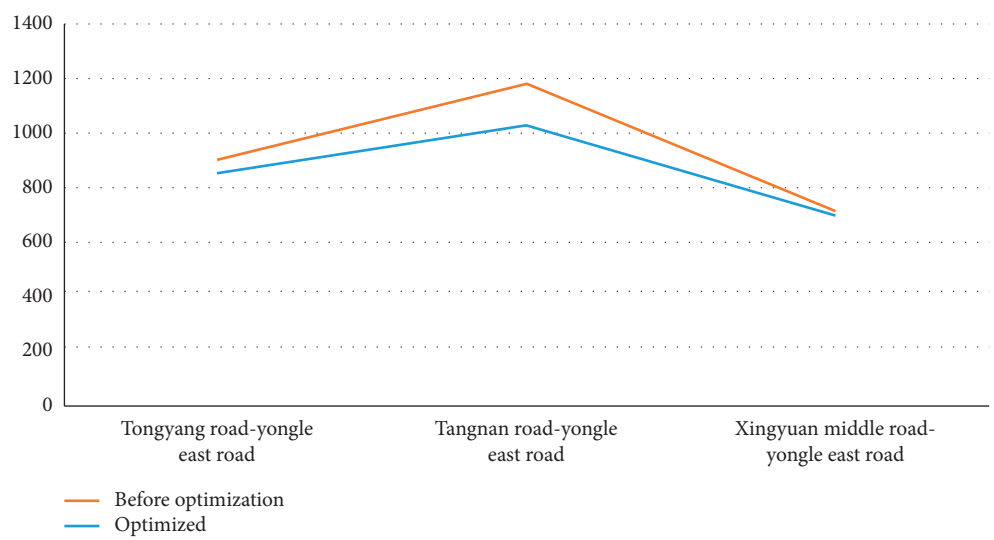


Figure 9: Comparison chart of capacity.

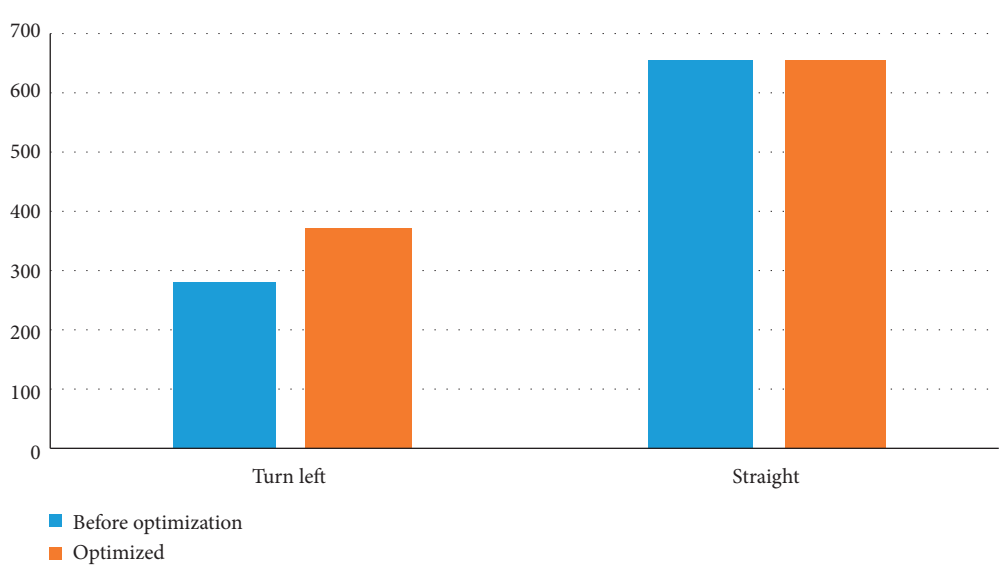


Figure 10: Comparison of saturation flow of each phase of Tongyang Road and Yongle East Road before and after optimization.

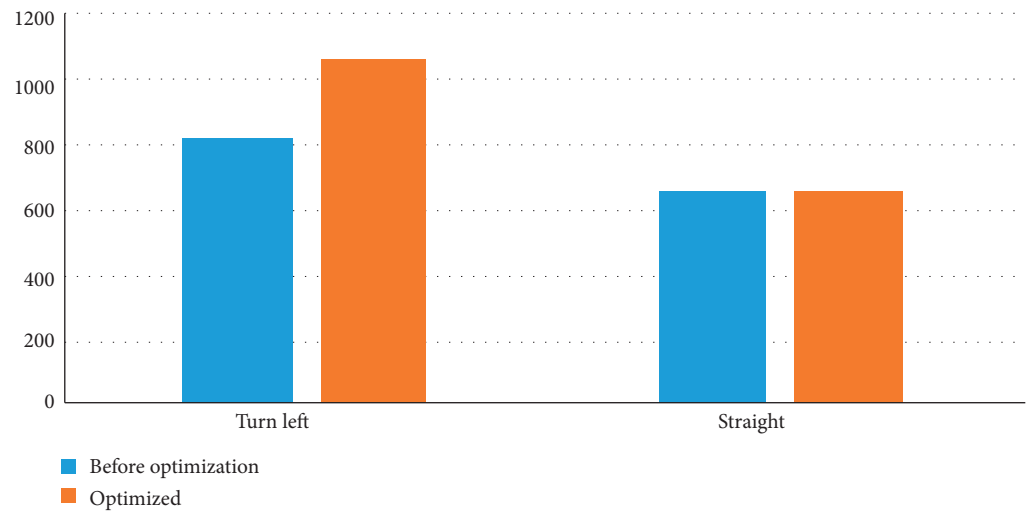


Figure 11: Comparison of saturation of each phase of Tangnan Road and Yongle East Road before and after optimization.

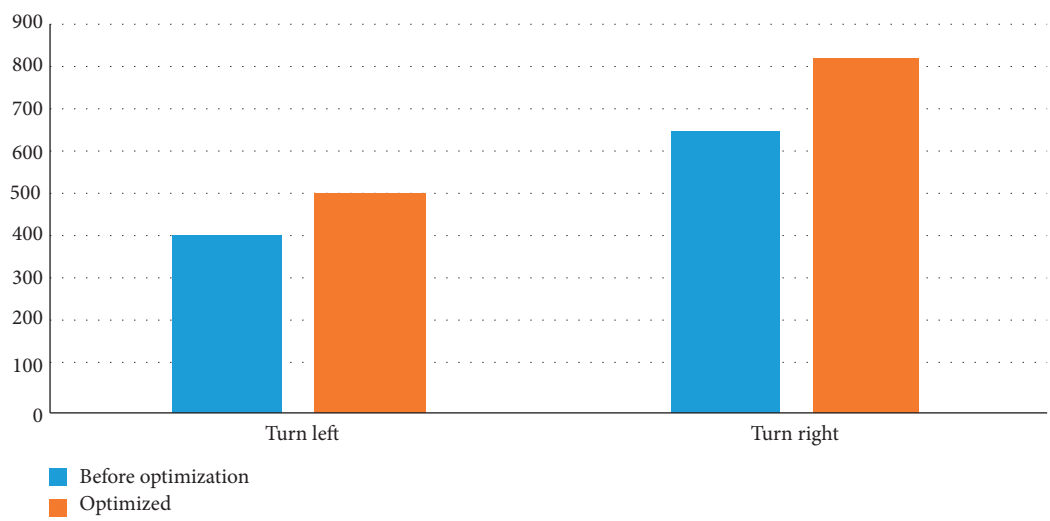


Figure 12: Comparison of saturation flow of each phase of Xingyuan Middle Road and Yongle East Road before and after optimization.

Table 8: Saturation flow rate after intersection optimization.

Intersection parameter	Turn left	Straight	Turn right
Intersection of Tongyang Road and Yongle East Road	366	652	—
Intersection of Tangnan Road and Yongle East Road	1068	651	—
Intersection of Xingyuan Middle Road and Yongle East Road	515	—	817

9.5%. From Figure 8, we can see that the green wave bandwidth is 36 s and the green wave velocity is 37.9 km/h, and the free flow speed in Yongle East Road is 36 km/h, which indicates that, by implementing the green wave control, the speed index could be optimized 5.3%. Tables 6 and 7 and Figure 9 show that, after implementing the green wave control and optimizing signal timing parameters and phase sequence at the three intersections on the Yongle East Road artery, the traffic capacity at the entrance of the intersections has been greatly improved, but from the comparison of Tables 1 and 7, it can be seen that the capacity of some phases at the intersection has not been improved obviously; for example, the improvement of the traffic capacity of the straight-through and left-turn phase at the intersection of Tongyang Road and Yongle East Road is not obvious, and the traffic capacity of the straight-through and left-turn phase at the intersection of Tangnan Road and Yongle East Road has not obviously improved. It can be seen from Figures 10–12 and Table 8 that the left-turn phase saturation flow at the intersection of Tongyang Road and Yongle East Road and the intersection of Tangnan Road and Yongle East Road has significantly improved. The left-turn

phase and right-turn phase saturation flow at the intersection of Xingyuan Middle Road and Yongle East Road has been significantly improved.

5. Conclusion

In this paper, the urban road intersection group was taken as the research object, and the IVICS was used to analyze the capacity of the intersection group under the real-time reversible lanes to determine the calculation model of the capacity after setting the real-time reversible lanes in the IVICS. Based on the real-time reversible lane, the signal phase sequence and signal timing of the intersection group were designed, and the green wave control scheme was designed for the mainline based on the phase difference model. It constructed a phase difference collaborative optimization model between intersections and discussed signal timing parameter design and signal phase sequence design in low and high saturation states. Through the case study of green wave scheme design of Yongle East Road, Wuxi City, Jiangsu Province, the effectiveness of green wave control based on the phase difference model was verified using the number of stops, green wave speed, capacity, and saturation flow as evaluation indicators. The results show that the number of stops was optimized by 9.5%, vehicle speed was improved by 5.3%, and intersection capacity and saturated traffic have been greatly improved.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

Lina Mao and Guiliang Zhou designed and wrote the paper; Yuke Dai, Yao Liu, and Pengsen Hu conducted the model and collected traffic data; Yuke Dai and Xu Bao analyzed the simulation results.

Acknowledgments

This research was supported by the open fund for Jiangsu Key Laboratory of Traffic and Transportation Security (Huaiyin Institute of Technology) (TTS2020-05 and TTS2020-09), Enterprise-University-Research Institute Collaboration Project of Jiangsu Province (BY2020005), Graduate Innovative Projects of Jiangsu Province (KYLX15_0148), the National Natural Science Foundation of China (61573098 and 51308246), University Natural Science Major Basic Project of Jiangsu Province (15KJA580001), and the Natural Science Foundation of Jiangsu Province, China (BK20171426).

References

[1] M. W. Levin and S. D. Boyles, "A cell transmission model for dynamic lane reversal with autonomous vehicles," *Transportation Research Part C*, vol. 68, pp. 126–143, 2016.

[2] K. J. Assi and N. T. Ratrout, "Proposed quick method for applying dynamic lane assignment at signalized intersections," *IATSS Research*, vol. 42, no. 1, p. 7, 2018.

[3] W. K. M. Alhajyaseen, M. Najjar, and N. T. Ratrout, "The effectiveness of applying dynamic lane assignment at all approaches of signalized intersection," *Case Studies on Transport Policy*, vol. 5, no. 2, pp. 1–22, 2017.

[4] S. S. Dey, J. ma, and Y. Haden, "Reversible lane operation for arterial roadways: the Washington, DC, USA Experience," *ITE Journal*, vol. 81, no. 5, pp. 26–35, 2011.

[5] X. Cheng, H. Zhang, and W. Yu, "The construction and application of the phase difference model considering the influence of inbound and outbound traffic flow on the main line coordination," *Science Technology and Engineering*, vol. 19, no. 6, pp. 243–249, 2019.

[6] D. Qu, M. Wan, J. Li, J. Wang, and X. Xu, "Trunk line phase difference optimization and its control method based on traffic wave theory," *Journal of Jilin University (Engineering and Technology Edition)*, vol. 47, no. 2, pp. 429–437, 2017.

[7] D. Qu, J. Yang, B. Qichun, W. Wang, and J. Zhou, "Phase difference optimization model based on queuing characteristics of trunk line traffic," *Journal of Jilin University (Engineering and Technology Edition)*, vol. 48, no. 6, pp. 1685–1693, 2018.

[8] J. Cao, *Cooperative Optimization Control Method for Lane Change Based on Tidal Flow Characteristics*, Qingdao Technological University, Qingdao, China, 2016.

[9] F. Jiao, *Reverse Variable Lane Dynamic Control Method Based on Multi-Source Data*, Shandong University of Technology, Zibo, China, 2018.

[10] Y. Liu, Y. Chang, and S. Mao, "Research on signal timing optimization of intersections with reverse variable lanes," *Journal of Chongqing University of Technology (Natural Science)*, vol. 32, no. 10, pp. 40–46, 2018.

[11] W. Liu, Z. Xie, and K. Chen, "Reverse variable lane signal timing optimization based on NSGA-II algorithm," *Journal of Chongqing Jiaotong University (Natural Science Edition)*, vol. 37, no. 6, pp. 92–97, 2018.

[12] T. Jiang, *Research on Variable Lane Setting Method of Signal-Controlled Level Crossing*, Chang'an University, Xi'an, China, 2019.

[13] X. Zeng, J. Xu, and Z. Wu, "Optimal allocation of time and space resources at intersections based on variable lanes," *Traffic Information and Safety*, vol. 36, no. 06, pp. 81–89, 2018.

[14] H. Xu, *Research on the Pre-signal Relationship of Variable Lanes at Signalized Intersections*, Beijing University of Technology, Beijing, China, 2014.

[15] X. Li, J. Li, and M. He, "A coordinated control method of two-way green waves for trunk lines based on phase optimization," *Journal of Traffic and Transportation Engineering and Information*, vol. 16, no. 1, pp. 115–121, 2018.

[16] W. Wang, *Traffic Engineering*, Southeast University Press, Nanjing, China, 2011.

Hindawi
Journal of Advanced Transportation
Volume 2020, Article ID 8894119, 10 pages
https://doi.org/10.1155/2020/8894119

Research Article

Layout Optimization of Campus Bike-Sharing Parking Spots

Tangyi Guo,[1,2] **Jie Yang**,[1] **Liu He**,[1,2] **and Kun Tang**[1,2]

[1]*School of Automation, Nanjing University of Science & Technology, Nanjing 210094, China*
[2]*MIIT Key Lab of Traffic Information Fusion & System Control, Nanjing 210094, China*

Correspondence should be addressed to Liu He; heliu@njust.edu.cn

Received 31 July 2020; Accepted 13 October 2020; Published 2 November 2020

Academic Editor: Dongfang Ma

Copyright © 2020 Tangyi Guo 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.

The rapid development of bike sharing has posed some challenges to the traffic management on campus. The bike sharing on campus has problems such as messy parking, and some buildings in the peak hours have no bikes to borrow. Thus, alternative parking spots are proposed based on the layout principle of parking spots for bicycles. An optimization model of the layout of campus bike-sharing parking spots with travel time and construction cost as the optimization goal is established, and the branch and bound algorithm is used to solve the model. Finally, the study analysis is carried out by optimizing the layout of the bike-sharing parking spot of Nanjing University of Science and Technology. The results show that, after optimizing the layout of parking spots, the average travel time of users is reduced by 6.0%, and the total construction cost is reduced by 27.3%. While being convenient for campus bike-sharing users, it also provides scientific decision-making support for the campus traffic management.

1. Introduction

Bike sharing not only helps to alleviate urban traffic pressure but also generates tremendous energy in constructing a green transportation trip system. It also has unique advantages in solving the "last mile" problem [1]. Bike sharing helps to alleviate urban traffic pressure in building a green energy traffic system and has a unique advantage in solving the "last kilometer" problem [2, 3]. The dockless bike sharing has got rid of the limitation of fixed parking piles and has the characteristics of small traffic capacity, flexible operation, good accessibility, and less investment [4], which has gradually covered most of the first- and second-tier cities in China, as well as Singapore, Washington, and other overseas cities. However, due to the regular change process of bikes demand, the bike-sharing system does not guarantee self-balancing, resulting in the phenomenon that a large number of shared bikes are idle in some areas and no bikes are available in some areas [5]. Therefore, it is imperative and necessary to set up a reasonable and convenient fixed storage spot for shared bikes.

The main problem of optimizing the number of parking spots and layout of shared bicycles can be solved by selecting the optimal facility location and related resource allocation based on different optimization objectives and constraints. Some scholars consider maximum coverage to optimize the shared bike system to maximize the demand covered by the shared bike parking spot or meet the limits of available budgets [6]. With the overall imbalance between supply and demand for shared bikes, Hu et al. [7] adopted three optimization models based on CMCLP to optimize the system configuration to achieve maximum service coverage [8–10]. Due to the close connection between the parking spots of bike sharing and the surrounding infrastructure, some scholars have considered the influence of the surrounding environment on the layout of shared bikes. In view of the optimization of the layout of shared bikes parking spots in scenic areas, Guo et al. [11] considered the distribution of subway stations and bus stations around the scenic area, by proposing an optimization model based on clustering and greedy algorithms, and solved the problem with an optimization coverage rate

of 89.2%. In establishing the layout optimization model of the shared bikes parking spot, we can also consider the multiobjective optimization model such as travel cost and construction cost [12]. Romero et al. [13] proposed a two-layer mathematical programming model to optimize the location of the base station while minimizing total costs and maximizing the number of system users. Luis Ines Frade et al. [14] proposed the maximum coverage model from the perspective of user demand, with the goal of maximizing demand and the optimized model with the available budget and system benefits as the constraints, so as to get the optimal location and configuration of the parking spot. In addition, George and Xia [15] introduced a queuing theory in the study of the size of the shared bike rental spot. García-Palomares et al. [16] proposed a GIS-based approach to calculate the spatial distribution of potential travel demand, using the location allocation model to determine the location of shared bike parking spots, the capacity of the parking spots, and the demand characteristics of defining the parking spots. At present, the optimization of the bike-sharing layout mainly focuses on scenic spots and cities, which involves the maximum coverage maximization and cost minimization. The model solution involves clustering, particle swarm algorithm [17], genetic algorithm, and ant colony algorithm [18], while there are few studies on the layout optimization of bike-sharing parking spots on campus.

Teachers and students are the main targets of bike-sharing service within the campus. The phenomenon of stopping and parking anywhere is widespread within the campus which can easily lead to the problem of 'difficulty to find a bike' thereby reducing the convenience of the user's travel [19]. According to the characteristics of the parking spot of bike sharing on campus, this paper defines the rules for setting up the parking spots of shared bicycles on campus and establishes the distribution model of the parking spots of bike-sharing on campus with travel time and construction cost as the optimization goal. Finally, the layout plan of the bike-sharing parking spot on campus is designed with specific cases to verify the rationality and feasibility of the model.

2. Problem Description

Considering that the traffic operation status quo is of different areas and the geographical conditions and structure of residents' travel are also different, it goes without saying that users will have different needs for bike sharing. To begin with, we divide the different functional areas of the campus which are mainly the teaching areas, office areas, living areas, and activity areas, to make certain of the demand for each service area; then, for different service areas, taking into account the reasonable number and location of parking spots, the premise of meeting the bike-sharing needs of each parking spot, the total travel time, and the total construction cost are minimized, and the layout optimization model is perfectly built.

2.1. Parking Lots Layout Planning. According to different service properties of different functional buildings, the campus is divided into several areas that are conducive to the layout of bike-sharing parking spots, and in each divided area, alternative parking spots are set according to different building radiation ranges.

The selection of alternative parking spots for bike sharing is in accordance with the following principles:

(1) Maximum service radius of parking spots: researches have shown that most bike sharing use areas within 300 meters from the station [18], and the maximum service radius of bike-sharing parking spots is fixed to 300 m, as shown in Figure 1

(2) Taking into consideration the nature of building services: the needs of people in different functional buildings for bike sharing are unpredictable, and the impact of user needs on the layout of parking spots should be fully considered

2.2. Layout Optimization Model. The aim of this model is to minimize the total travel time of users and spot construction costs on the premise of meeting the bike-sharing needs of each parking spot. The total travel time of users includes the total time of walking to the station, riding time, and time spent walking to the destination after returning the bike. The spot construction cost includes fixed cost of parking spot and cost of bikes. The model parameter symbols description is shown in Table 1.

The model objective functions are as follows:

$$\min T = \sum_{i\in I}\sum_{i'\in I}\left(\frac{\sum_{j\in J} bs_{i,j,i'}\cdot d_{i,j}}{u_1} + \frac{\sum_{j\in J}\sum_{j'\in J} br_{i,j,j',i'}\cdot d_{j,j'}}{u_2} + \frac{\sum_{j'\in J} rs_{i,j',i'}\cdot d_{i',j'}}{u_1}\right), \tag{1}$$

$$\min P = \sum_{j\in J} y_j\cdot c_0 + \sum_{j\in J} y_j\cdot bn_j\cdot c_1, \tag{2}$$

$$\min \psi = k\cdot T + (1-k)\cdot P. \tag{3}$$

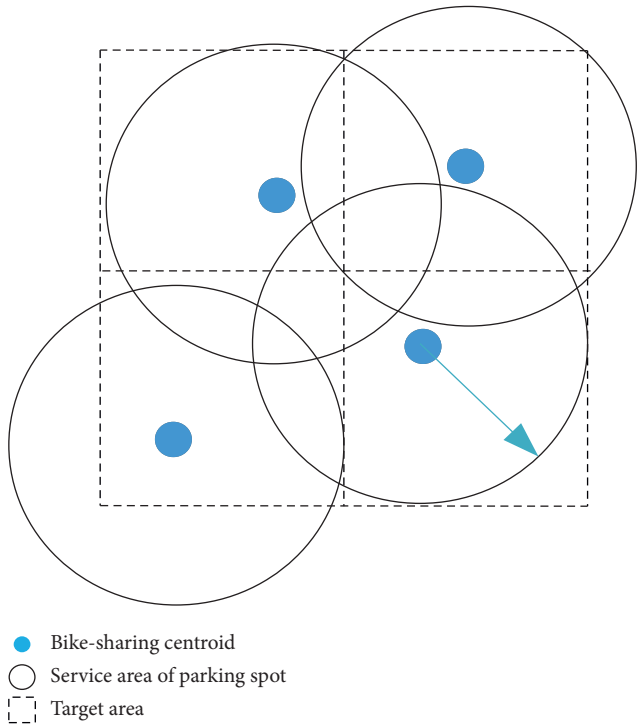


Figure 1: Diagram of the service scope of the bike-sharing parking spots.

Table 1: Model parameter symbols description.

Parameters	
I	Set of target areas, where, $i, i' \in I$
J	Set of alternative bike-sharing parking spots, where, $j, j' \in J$
k	Weight of the objective function
y_j	A binary variable to determine whether the parking spot is optimized, and the value of the optimized parking spot is 1
$bs_{i,j,i'}$	The number of the users depart from the target area i to the target area i', select borrow bicycles at the parking spot j, and the travel route is $i \longrightarrow j \longrightarrow i'$, where, $i, i' \in I$, $j \in J$
bn_j	The number of bikes per spot
$rs_{i,j',i'}$	The number of the users depart from the target area i to the target area i', select return bikes at the parking spot j', and the travel route is$i \longrightarrow j' \longrightarrow i'$, where, $i, i' \in I$, $j' \in J$
$br_{i,j,j',i'}$	The number of the users depart from the target area i to the target area i', select borrow bicycles at the parking spot j, and select return bicycles at the parking spot j'. The travel route is $i \longrightarrow j \longrightarrow j' \longrightarrow i'$, where, $i, i' \in I$, $j \in J$
$d_{i,j}$	Distance from the target area i to the parking spot j, where,$i \in I$, $j \in J$
$d_{j,j'}$	Distance from the parking spot j to the parking spot j',where,$j, j' \in J$
$d_{i',j'}$	Distance from the target area i' to the parking spotj', where, $i' \in I$, $j' \in J$
$S_{\min}, S_{\max}$	The lower limit and upper limit of the number of alternative construction parking spots
C	The service area of the parking spots
$\alpha_{i,j}$	A binary variable determines whether the target area i is within the service range of the optimized parking spot j; when the target area is within the service range of the parking spot, the value is 1, where,$i \in I, j \in J$
u_1, u_2	Walking and cycling speed
M	Maximum service capacity of parking spots
$D_{i,i'}$	The number of vehicles from target area $i$to target area i'

Equation (1) is the objective function, that is, the smallest total travel time of the user, including the walking time, borrowing and returning the bike, and riding time. Equation (2) is the objective function of construction cost. Equation (3) is the synthetic objective function.

Restrictions:

$$S_{\min} \le \sum_{j \in J} y_j \le S_{\max}, \tag{4}$$

$$\alpha_{i,j} = \begin{cases} 1, d_{i,j} \le C \\ 0, d_{i,j} > C \end{cases}, \quad \forall i \in I, \forall j \in J, \tag{5}$$

$$\sum_{j\in J} \alpha_{i,j} \cdot y_j \geq 1, \quad \forall i \in I, \tag{6}$$

$$\sum_{i\in J} \alpha_{i,j} \geq 1, \quad \forall j \in J, \tag{7}$$

$$bs_{i,j,i'} \leq M\alpha_{i,j} y_j, \forall i, i' \in I, \text{and } i \neq i', \quad \forall j \in J, \tag{8}$$

$$rs_{i,j',i'} \leq M\alpha_{i',j'} y_j, \forall i, i' \in I, \text{and } i \neq i', \quad \forall j' \in J, \tag{9}$$

$$bs_{i,j,i'} = \sum_{j'\in J} br_{i,j,j',i'}, \forall i, i' \in I, \text{and } i \neq i', \quad \forall j \in J, \text{and } j \neq j', \tag{10}$$

$$rs_{i,j',i'} = \sum_{j\in J} br_{i,j,j',i'}, \forall i, i' \in I, \text{ and } i \neq i', \quad \forall j' \in J, \text{ and } j \neq j', \tag{11}$$

$$\sum_{j\in J} bs_{i,j,i'} \geq D_{i,i'}, \forall i, i' \in I, \tag{12}$$

$$y_i = \{0, 1\}, \quad \forall j \in J, \tag{13}$$

$$\sum_{j\in J} \alpha_{i,j}, bn_j, y_j \geq D_i. \tag{14}$$

Equation (4) is to optimize the construction of the number of bike-sharing parking spots, to avoid the situation where the number of parking spots is too small or too much, resulting in inefficient use of shared bike systems or resource redundancy; equation (5) is the target area i within the service area of the optimized parking spot j' s scope of services of binary variable, that is, to ensure that the user can find the parking spot within the maximum tolerable walking distance to complete the borrowing and returning of the bike within the given target area; equation (6) provides a bike borrowing and returning service for at least one bike-sharing parking spot in any target area; equation (7) serves at least one target area for a shared bike parking spot; equations (8) and (9) restrict users to borrow and return bikes at only optimized parking spots; equation (10) constrains the number of borrowed bikes at any one parking spot to be equal to the sum of the number of returned bikes from any parking spot to each parking spot; equation (11) constrains the number of bikes returned at any one parking spot to be equal to the sum of the number of vehicles returned from each parking spot to any one of the parking spots; equation (12) constrains the number of vehicles demanded from each target area to travel to other areas; equation (13) is the binary variable to optimize the parking spot; and equation (14) ensures that the number of vehicles in the target area meets the demand in the target area.

3. Model Solution

Owing to the fact that branch and brand is a basic method for solving integer planning (or mixed integer planning) in operations, it is very efficient to seek the optimal solution of integers. In this paper, the model is solved by the branch and bound method, and the solving steps are as follows:

Step 1: first, the integer constraints of the original problem are not considered, to solve the corresponding relaxation problem, and a graph or simplex method is used to obtain the optimal solution Z.

Step 2: if the optimal solution sought is just the integer solution, the integer solution is the optimal solution of the original integer planning problem.

Step 3: branch: based on our understanding of the importance of variables, we select a value that does not meet the integer constraint X_j in the optimal solution, its value is b_j, where $[b_j]$ represents the maximum number which is less than b_j. Two constraints were constructed, $X \leq b_j$ and $X \geq [b_j] + 1$, joining the original LP problem separately which formed two subproblems because $[b_j]$ and $[b_j] + 1$ have no integers between them, so the integer solution within these two subsets must be consistent with the original feasible solution of the whole number solution.

Step 4: bounding: first, we determine whether there is an integer solution to each subproblem. If there is one, we find out the integer solution corresponding to the maximum value of the target function, fixed as Z^*, and then, the integer solution of the problem is the target function $Z \geq Z^*$, which is the bounding. Also, in the branching process, once there is a subproblem, then $Z^* = Z$.

Step 5: if there is a subproblem greater than Z^*, that needs to be branched out. If there is no integer solution in Step 4, it is also necessary to continue to branch to find the integer solution and to branch the subproblem corresponding to the maximum value of the target function.

Step 6: if the target value of all subproblems is less than or equal to Z^*, there is no need to continue branching, and the corresponding integer solution for Z^* is the optimal solution.

The solution process of branch and bound algorithm is shown in Figure 2.

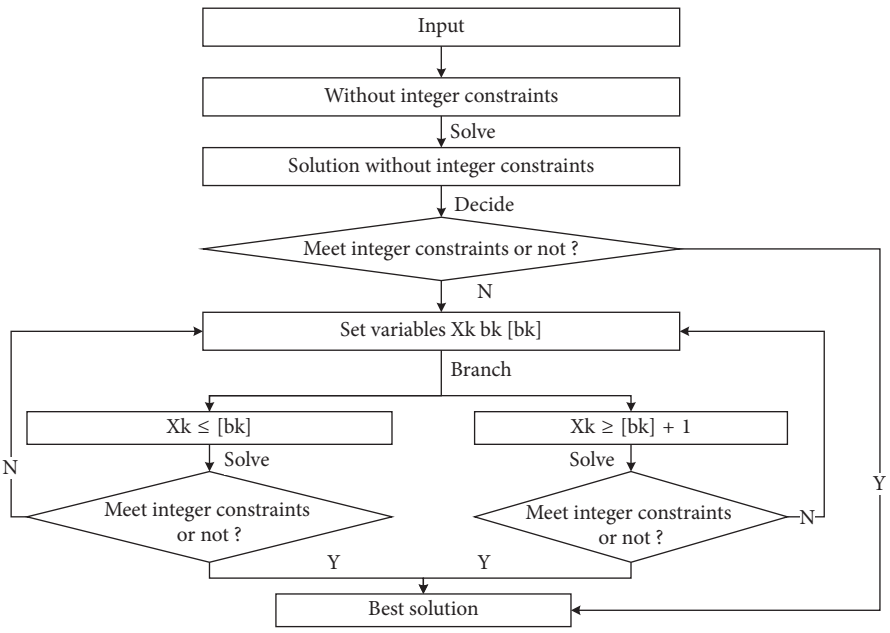


Figure 2: Solution process of branch and bound algorithm.

4. Case Study

In order to verify the accuracy of the layout optimization model of shared bike parking spots and the accuracy of the scheduling optimization model and the effectiveness of solving the genetic algorithm of the abovementioned model, this paper analyzes the genetic algorithm of Nanjing University of Technology as the research object. The school covers an area of 3118 acres, there are more than 30,000 students and more than 3,200 staff, and the travel demand is comparatively greater.

First of all, a statistical assessment was carried out on the number of peopleN_i working in the functional buildings and regions who travel during peak hours. The distribution of alternative parking spots is shown in Figure 3. Different needs for bike sharing for users in different regions are used to calculate the demand for bikes. The percentage of bikes used for different regions is based on the analysis of survey data, as is shown in Table 2.

$$\text{Demand}_i = N_i * b_i, \tag{15}$$

where b_i is the percentage of bikes used for each functional region.

Considering the fact that the teaching area, office area, living area 1 and living area 3, and other space range are large, in order to ensure the reasonable distribution of parking spots and the use of grid layout, taking into account that the service radius of the parking spot is 300 m, the set style size is 300 m × 300 m, to set the target regional center spot coordinates, as shown in Figure 4. Then, the target regional center spots and coordinate values of the alternate parking spots and the shared bike demand between the target areas are entered separately as the parking spot layout optimization model input data.

The parking spot layout optimization model parameter values are as shown in Table 3.

In a global optimal angle, the travel time of all users is the target function, and the layout scheme with the smallest total travel time and the total construction cost is solved. Based on the branch boundary method, this paper uses Matlab to write a program to solve the model, and the optimization results are shown in Figure 5. After optimization, the number of shared bike parking spots has been reduced from 51 to 35, which effectively reduces the operating and maintenance costs of shared bike parking spots.

Because of the convenience of the pile-free shared bikes, when considering the site construction cost, there is no need to consider the cost of building a parking pile, so the site construction cost only considers the fixed cost and the bike operation and maintenance cost.

$$P = \sum_{j \in J} y_j \cdot c_0 + \sum_{j \in J} y_j \cdot bn_j \cdot c_1, \tag{16}$$

where P is the total construction cost, y_j is the binary variable of the alternative site to optimize the site, bn_j represents the number of bikes per spot, c_0, c_1 ,respectively, are the cost of construction of a single site and the cost of operation and maintenance of a single bike, $c_0$equals 200 Yuan/unit, and c_1 equals 500 Yuan/bike.

Compared with the optimization model that only considers the shortest travel time, the model proposed in this paper reduces the average travel time of users by 6.0% and reduces the total construction cost by 27.3%. The comparative analysis of optimization results is presented in Table 4. The number of optimized parking spots and corresponding shared bikes is shown in Table 5. The

Figure 3: Distribution of alternative parking spots at Nanjing University of Science and Technology.

Table 2: Campus zoning planning.

No.	Zone	Spot	Number of people travelling	Bicycle ratio	Demand	Number of spots
1	Entrance	Gate 1	170	0.22	37	8
2		Gate 2	34	0.22	7	
3		Gate 3	272	0.22	60	
4		Gate 7	170	0.22	37	
5		South gate	340	0.22	75	
6		Gate 5	170	0.22	37	
7		Underground tunnel 1	816	0.22	180	
8		Underground tunnel 2	850	0.22	187	
9	Living area 1	Xiyuan District 1	340	0.15	51	6
10		Huayuan District	680	0.15	102	
11		Zhongshan District	680	0.15	102	
12		Campus hospital	272	0.15	41	
13		Zhuyuan District	170	0.15	26	
14		Xiyuan District 2	136	0.15	20	
15	Living area 2	Keyuan District	510	0.13	66	3
16		Zilu hotel	56	0.13	2	
17		Postgraduate canteen	272	0.13	35	
18	Active area	Academic Ex. Center	204	0.11	22	4
19		Art and Culture Museum	68	0.11	7	
20		Sports center	272	0.11	30	
21		School of Foreign Language	306	0.20	61	

Table 2: Continued.

No.	Zone	Spot	Number of people travelling	Bicycle ratio	Demand	Number of spots
22	Office area	Sch. of Computer Sci.	782	0.20	156	8
23		Sch. of Electrical and Optical Sci.	816	0.20	163	
24		Sch. of Chemical Eng.	408	0.20	82	
25		Transient physical state	17	0.20	3	
26		Key labs	17	0.20	3	
27		Civil Explosion building	17	0.20	3	
28		Intelligent building	340	0.20	4	
29		Basic laboratory building 1	374	0.20	4	
30	Teaching area	Basic laboratory building 2	680	0.21	143	13
31		Teaching building 1	782	0.21	164	
32		Teaching building 2	816	0.21	171	
33		Teaching building 3	680	0.21	143	
34		Teaching building 4	340	0.21	71	
35		Yifu building	340	0.21	71	
36		Library	170	0.21	36	
37		Sport Gallery	204	0.21	43	
38		Qian Xuesen College	340	0.21	71	
39		Zhiyuan building	306	0.21	64	
40		Materials research center	170	0.21	36	
41		Printing plant	170	0.21	36	
42		Experiment center	34	0.21	7	
43	Living area3	Zhizhen building	170	0.12	20	9
44		Youth League building	340	0.12	41	
45		2 and 3 canteen	408	0.12	49	
46		Ming Yuan	340	0.12	41	
47		Campus supermarket	306	0.12	37	
48		Gymnasium	306	0.12	37	
49		204 Student Dormitory	340	0.12	41	
50		Int. Students Dormitory	612	0.12	73	
51		Huizhi Pavilion	680	0.12	82	

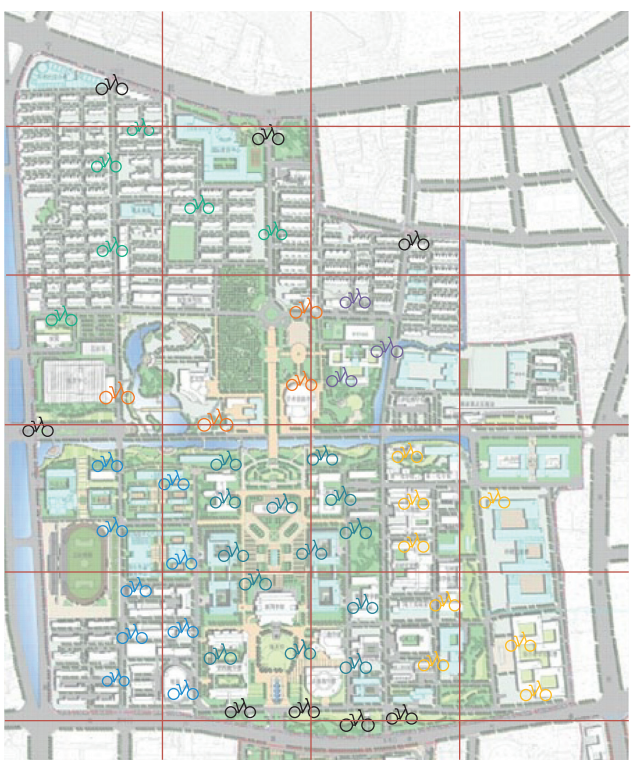


Figure 4: Grid zoning.

Table 3: Parameter values of the optimized model of the parking spot layout.

Parameter	C	u_1	u_2	M	$S_{\min}$	$S_{\max}$
Value	300	1.4	5	1000	25	40

Figure 5: Planning and layout of optimized parking spots.

Table 4: Comparative analysis of optimization results.

Parameter	Research model	Contrast model
Total travel time/s	418489.6	394822.6
Total construction cost/Yuan	671600	923500
Objective function value	545044.8	659161.3
The number of shared bike parking spots	28	35

Table 5: Number of optimized parking spots and corresponding shared bikes.

Parking spot	1	2	3	4	5	6	7	8	9	10
Number of shared bikes	37	7	60	37	75	51	102	26	6	27
Parking spot	11	12	13	14	15	16	17	18	19	20
Number of shared bikes	27	61	163	3	3	37	68	78	82	68
Parking spot	21	22	23	24	25	26	27	28		
Number of shared bikes	34	34	17	17	40	44	40	88		

Table 6: The optimization result when k takes different values.

Parameter	Total travel time/s	Total construction cost/Yuan	The number of shared bike parking spots
$k = 0.1$	483624	644300	24
$k = 0.2$	445960	651900	27
$k = 0.3$	445855	651900	27
$k = 0.4$	436092	656700	26
$k = 0.5$	418490	671600	28
$k = 0.6$	409266	682000	30
$k = 0.7$	409266	682000	30
$k = 0.8$	401180	703900	32
$k = 0.9$	401180	703900	32

planning and layout of optimized parking spots are shown in Figure 5.

This paper also analyzes the influence of the weight k value in the model on the layout optimization results, as shown in Table 6. The results show that the value of k determines the impact of the total travel time and construction cost on the objective function, so the corresponding optimization results are quite different. When $k = 0.1$, the optimized parking point is 24; when $k = 0.9$, the optimized parking point is 32.

5. Conclusions

(1) Considering the influence of the functionality of campus buildings on the demand of bike sharing, the layout rules of the bike-sharing parking spot are proposed.

(2) In view of the optimization of the campus bike-sharing parking spot layout, a model of campus bike-sharing parking spot planning is established with the total travel time of the user and the total construction cost as the optimization goal.

(3) The rationality of the model is verified by using Nanjing University of Science and Technology as the research case study. The user's travel time is reduced by 6.0%, the total construction cost is reduced by 27.3%, and it has a certain reference value for the operation management of bike-sharing on campus.

The results also show that considering different factors will have a greater impact on the optimization results. There are many factors that affect the layout of campus bike-sharing parking spots. In the subsequent research work, the planning of shared bicycle parking spots under multiobjective and multiconstraint conditions can be considered to further improve the model.

Data Availability

On request, the data supporting the results of this study can be provided by the author of the article.

Conflicts of Interest

The authors affirm that there are no conflicts of interest.

Acknowledgments

This research was supported by "the Fundamental Research Funds for the Central Universities" (Nos. 30920010010 and 30919011290).

References

[1] H. Zhang, X. Song, T. Xia et al., "MaaS in bike-sharing: smart phone GPS data based layout optimization and emission reduction potential analysis," *Energy Procedia*, vol. 152, pp. 649–654, 2018.

[2] Z. Zheng, C. Lei, and Y. Ouyang, "Optimal investment and management of shared bikes in a competitive market," *Transportation Research Part B*, vol. 135, 2020.

[3] S. A. Shaheen, A. P. Cohen, and E. W. Martin, "Public bikesharing in north America, early operator understanding and emerging trends," *Transportation Research Record*, vol. 2387, no. 1, 2013.

[4] Y. Liu, W. Y. Szeto, and S. C. Ho, "A static free-floating bike repositioning problem with multiple heterogeneous vehicles, multiple depots, and multiple visits," *Transportation Research Part C*, vol. 92, 2018.

[5] S. A. Shaheen, S. Guzman, and H. Zhang, "Bikesharing in europe, the americas, and asia: past, present, and future," *Transportation Research Record Journal of the Transportation Research Board*, vol. 2143, pp. 159–167, 2019.

[6] S. Davari, M. H. Fazel Zarandi, and I. Burhan Turksen, "A greedy variable neighborhood search heuristic for the maximal covering location problem with fuzzy coverage radii," *Knowledge-Based Systems*, vol. 41, pp. 68–76, 2013.

[7] Y. Hu, Y. Zhang, D. Lamb et al., "Examining and optimizing the BCycle bike-sharing system–a pilot study in Colorado," *US. Applied Energy*, vol. 247, 2019.

[8] H. Kong, T. Scarlett, and D. Z. S. Jin, "Deciphering the relationship between bikesharing and public transit: modal substitution, integration, and complementation," *Transportation Research Part D: Transport and Environment*, vol. 85, 2020.

[9] K. Wang and Y.-J. Chen, "Joint analysis of the impacts of built environment on bikeshare station capacity and trip attractions," *Journal of Transport Geography*, vol. 82, 2020.

[10] Y. Guo and S. Y. He, "Built environment effects on the integration of dockless bike-sharing and the metro," *Transportation Research Part D*, vol. 83, 2020.

[11] T. Guo, P. Zhang, F. Shao et al., "Allocation optimization of bicycle-sharing stations at scenic spots," *Journal of Central South University*, vol. 7, 2014.

[12] M. Benchimol, P. Benchimol, and B. Chappert, "Bike sharing systems: solving the static rebalancing problem," *Discrete Optimization*, vol. 10, no. 2, pp. 120–146, 2013.

[13] J. P. Romero, A. Ibeas, J. L. Moura, and B. Alonso, "A simulation-optimization approach to design efficient systems of bike-sharing," *Procedia-Social and Behavioral Sciences*, vol. 54, pp. 646–655, 2012.

[14] I. Benavente and A. Ribeiro, "Bike-sharing stations: a maximal covering location approach," *Transportation Research Part A: Policy and Practice*, vol. 82, pp. 216–227, 2015.

[15] D. K. George and C. H. Xia, "Fleet-sizing and service availability for a vehicle rental system via closed queueing networks," *European Journal of Operational Research*, vol. 211, no. 1, pp. 198–207, 2011.

[16] J. C. García-Palomares, J. Gutiérrez, and M. Latorre, "Optimizing the location of stations in bike-sharing programs: a gis approach," *Applied Geography*, vol. 35, no. 1-2, pp. 235–246, 2012.

[17] L. Liu, L. Sun, Y. Chen et al., "Optimizing fleet size and scheduling of feeder transit services considering the influence of bike-sharing systems," *Journal of Cleaner Production*, vol. 236, 2019.

[18] C. S. Shui and W. Y. Szeto, "Dynamic green bike repositioning problem - a hybrid rolling horizon artificial bee colony algorithm approach," *Transportation Research Part D: Transport and Environment*, vol. 60, pp. 119–136, 2018.

[19] A. Kabra, E. Belavina, and K. Girotra, "Bike-share systems: accessibility and availability. working paper," *Chicago Booth Research Paper*, vol. 15, 2016.

Hindawi
Journal of Advanced Transportation
Volume 2020, Article ID 8870449, 10 pages
https://doi.org/10.1155/2020/8870449

Research Article

Measuring the Spatial Spillover Effects of Multimodel Transit System in Beijing: A Structural Spatial Vector Autoregressive Approach

Honghai Li,[1] Xiaolei Ma,[1,2] Xian Zhang,[1] Xin Li,[3] and Weihan Xu[3]

[1]*School of Transportation Science and Engineering, Beijing Key Laboratory for Cooperative Vehicle Infrastructure System and Safety Control, Beihang University, Beijing 100191, China*
[2]*Beijing Advanced Innovation Center for Big Data and Brain Computing, Beihang University, Beijing 100191, China*
[3]*College of Transportation Engineering, Dalian Maritime University, Dalian 116026, China*

Correspondence should be addressed to Xin Li; xtopli@dlmu.edu.cn

Received 28 March 2020; Revised 1 September 2020; Accepted 19 October 2020; Published 31 October 2020

Academic Editor: Zhuping Zhou

Copyright © 2020 Honghai Li 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.

Changes in local transit passenger flow may cause a spatial spillover effect across the involved regions and affect traffic patterns in other regions. To identify the affected areas and the traffic patterns, this study develops an enhanced spatial vector autoregressive (SpVAR) model to investigate relations in public transport systems in the case of sudden large passenger flow impact. The proposed model captures the interacted correlation within different transit models in separated regions. Three representative commuting regions in Beijing, namely, Zhongguancun, Guomao, and Huilongguan, are employed for empirical study. Results confirm the existence of spatial spillover effect in the commuter regions and reveal heterogeneous effects of multimodal transit system on regions with different distances.

1. Introduction

In recent years, the transportation infrastructures have been significantly improved in China which strengthens the transfer capabilities between different transit modes. As a result, some types of changes in passenger volume taking one specific transit mode in some areas or regions are highly likely to affect other transit modes in other unconnected regions. Such phenomenon is known as a spatial spillover effect, which may reveal some spatial conduction laws in the multimodal transit system. A better understanding about the spatial spillover effect within multimodal transit network could help us to predict some unexpected events and further to plan the necessary countermeasures.

To this end, numerous scholars have made efforts to explore the contributory factor which may lead to the spatial spillover effect. The results suggest that the interaction among the multimodal traffic flow (passenger flow) is a major factor [1–7]. Then, a variety of statistical models, especially regressive models, have been developed to better observe and capture such interactions.

Although a number of enhanced spatial econometrics models have been proposed to reflect spatial and time features, they typically can only analyze the case of single endogenous variable and lack of sufficient capabilities to investigate the correlation among multiendogenous variables. Such limitation led those enhanced VAR model to lose the capability of analyzing the spatial spillover phenomenon. To address this issue, Conley and Dupor [8] and Neusser [9] further improved the framework of spatial VAR model again for capturing the spatial spillover effects considering temporal and spatial correlations, which later contributes to developing the spatial vector autoregressive (SpVAR) model. The developed SpVAR model is thus capable of capturing the actual impact of regional factors and reflecting the regional spatial relations [10–12]. Although various types of SpVAR models have been developed in recent

years, the application of spatial econometrics in transport fields is limited, particularly to study the passenger flow patterns within multimodal transit system due to the fact that all the existing SpVAR models lack transportation elements, so they are incapable of reflecting the impact of spatial distance on change of interregional transit. Deng [13] developed a spatial vector autoregressive (VAR) model to predict traffic flows within a simulated system. Chen et al. [14] constructed SpVAR models at typical analysis periods for volume and speed forecasting by considering different combinations of upstream and downstream impacts. However, there is a research gap in using SpVAR model to study the traffic flow patterns within multimodal transportation system.

To fill this gap, this study proposed an enhanced SpVAR model to better understand the relationship among different types of passenger flows in multimodal transit system, specifically to explore the spatial spillover effect in multimodal transit system. In detail, a weight matrix of spatial factors is firstly embedded for including linear interdependence among multiple spatial time series variables. Furthermore, another new feature, impulse response, is also designed and included for generating the dynamic response of interregional passenger flow when change of passenger flow is in other regions and other modes. As a result, the enhanced SpVAR model is assigned with spatial and geographical attributes to address the interregional and temporal correlations of passenger flow in the multimodal transit system. In addition, the proposed model can detect multivariable interactions and explore spatial correlations among different transit modes in the multimodal network. Specifically, the model can capture the response of each mode in one specific region in case of the changes of other modes in other regions. To summarize, the improved model with the impulse response is assigned with the following capabilities:

(1) It is able to measure the current value and also to predict passenger volume in other regions after a unit passenger flow impulse is generated in a certain area.

(2) It is able to detect the interactions in multivariables and spillover effect in different regions under the change of passenger flow in the specific transit mode.

The rest of the paper is organized as follows. Section 2 describes the structure and format of the data. Section 3 introduces the structure of the SpVAR model, including identification of parameters and spatial weights, solution of impulse response, and application to the interaction of passenger flow. Section 4 presents the three representative traffic analysis zones (TAZ) in Beijing selected using the average daily passenger flow of bus alighting volume, metro boarding, and alighting volume per 15 minutes.

2. Data Source

To better analyze the spatial spillover of multimodal transit demands, this paper discusses the intraregional and interregional interaction of bus and metro flow by using real integrated circuit card (IC card) data in Beijing.

Beijing is the capital of China and one of the megacities with a population of over 21.71 million.

Figure 1 shows Beijing that is divided into 1911 TAZ according to the socioeconomic data and the principle of homogeneity and uniformity; the red rectangle represents the studied area.

In this paper, the proposed improved SpVAR model is constructed using time-panel passenger flow data in Beijing. This study investigates the interregional spillover effect of passenger flow by bus and metro in the selected areas. In Beijing public transit system, passengers are required to tap in and tap out their smart cards during both boarding and alighting procedure, and thus Metro boarding volume, metro alighting volume, and bus alighting volume are obtained from the Automatic Fare Collection (AFC) installed on the metro station and bus. The proposed model variables are metro boarding volume, metro alighting volume, and bus alighting volume every 15 minutes in each TAZ. Point of Interest (POI) data are also collected to quantify traffic attraction with each TAZ. The typical POI information can be categorized into land use and transport. In land use category, the densities of residential building, employment, hotel, service facility, attraction, and commercial establishment are measured, while in traffic category, bus stop density, metro station density, and road density are calculated in each TAZ.

For the specific research objects, we choose three typical and representative TAZs in Beijing. The criteria of setting up these TAZs include similar census block information (e.g., household and employment densities) within each TAZ. These zones are marked with the red box in Figure 1 and include districts surrounding Zhongguancun (hereinafter referred to as ZGC), Guomao (hereinafter referred to as GM), and Huilongguan (hereinafter referred to as HLG) (Figure 2). The straight-line distance between ZGC and HLG is 16 km and that between GM and HLG is 21 km, and the network distances between ZGC and HLG as well as HLG and GM are 19 km and 32 km, respectively.

GM and ZGC are business office centers and the morning commuter destinations in Beijing. HLG is one of the largest residential centers and the origination of early peak commuters in Beijing. The traffic among the three regions constitutes the main commuter flow in Beijing.

3. Model

3.1. Spatial Vector Autoregressive Model. This study improves SpVAR model to incorporate space and time dimensions as well as the spatial-related disturbances by adding spatial weight matrix and impulse response function.

The SpVAR model introduces the spatial dimension into the parameters of the SVAR model, which is capable of capturing both spatial and temporal dynamics of multimodal passenger flows. However, the spatial weights of SpVAR model commonly rely on distances, which may not hold true to model interregional impulse effect among multiregion transport demands [15].

The model contains N regions and K variables, and the structure of the model for each variable in each region is shown as follows:

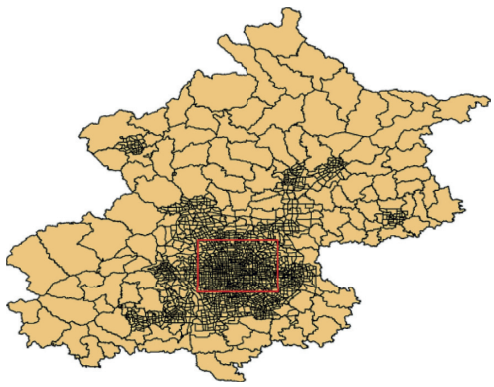


Figure 1: Traffic analysis zones in Beijing.

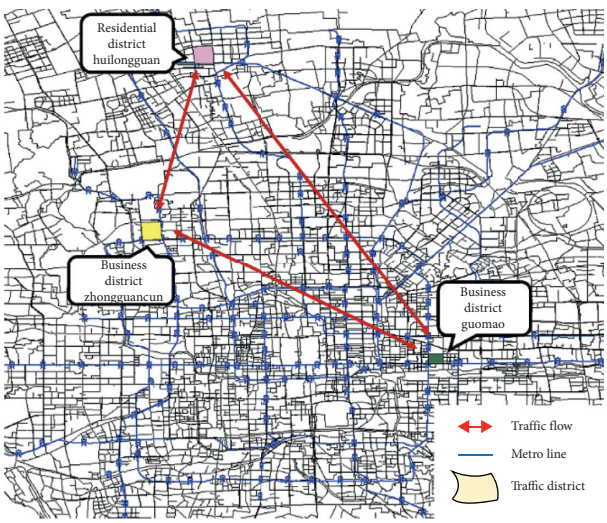


Figure 2: Studied area.

$$y_{nt}^{i} = \alpha_{n}^{i} + \sum_{j=1}^{p}\sum_{k=1}^{K}\beta_{j}^{ik}y_{n,t-j}^{k} + \sum_{k=1}^{K}\lambda_{0}^{ik}\sum_{m=1}^{N}w_{nm}y_{mt}^{k} + \sum_{j=1}^{q}\sum_{k=1}^{K}\lambda_{j}^{ik}\sum_{m=1}^{N}w_{nm}y_{m,t-j}^{k} + u_{nt}^{i}, \quad (1)$$

$$u_{nt}^{i} = \sum_{k=1}^{K}\varphi^{ik}\sum_{m=1}^{N}w_{nm}u_{mt}^{i} + \varepsilon_{nt}^{i},$$

where y_{nt}^{i} refers to the travel demand of mode i in region n and i represents three variables, namely, metro boarding, metro alighting, and bus alighting flow, respectively $(i = 1, 2, 3)$. Traffic zones or regions are labeled as $n = 1, 2, \ldots, N$, and time periods are labeled as $t = 1, 2, \ldots, T$. w_{nm} represents the spatial weights matrix related to passenger flow, and α_{n}^{i} denotes the traffic regional specific effects. Regional specific effects refer to the unique effects determined by certain specific features in each region. For this study, the traffic regional specific effects may be related to the number of traffic facilities, the structure of the network, and so on. For each variable in each region, α_{n}^{i} has a specific value. The parameters of the model include time lag coefficient β_{j}, spatial lag coefficient λ_{0}^{ik}, and spatial lag coefficient with time lag λ_{j}^{ik}. Time lag represents the value of variables in time series delayed from the current time, and spatial lag indicates the value of a variable that is in a different spatial relationship with the variable. Each variable in each region is assumed to be generated by the interaction of the four indicators, namely, fixed effect α_{n}^{i}, time lag values of each variable $y_{n,t-j}^{k}$, spatial lag values of each variable $w_{nm}y_{mt}^{k}$, and time lag values that contain the spatial lag of each variable $w_{nm}y_{m,t-j}^{k}$. Spatial-related error term u_{nt}^{i} exists in case $\varphi^{ik} \neq 0$. We carried out a series of transformations to produce a relatively easy solution from (2). Details are not shown due to length limitations.

$$C_{0}y_{t} = \alpha + C_{1}y_{t-1} + C_{2}y_{t-2} + \cdots + C_{h}y_{t-h} + \cdots + C_{p}y_{t-p} + \varepsilon_{t}, \quad (2)$$

where C_0 is the coefficient matrix that characterizes the contemporaneous correlation of the SpVAR model.

$$C_{0} = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \\ C_{21}^{0} & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & 1 & \vdots & \vdots \\ C_{NK-1,1}^{0} & C_{NK-1,2}^{0} & \cdots & 1 & 0 \\ C_{NK,1}^{0} & C_{NK,2}^{0} & \cdots & C_{NK,NK-1}^{0} & 1 \end{bmatrix}, \quad (3)$$

C_h contains the coefficient matrix with a spatial structure and can be expressed as follows:

$$C_{h} = \begin{bmatrix} A_{11}^{(h)} & A_{12}^{(h)} & \cdots & A_{1K}^{(h)} \\ A_{21}^{(h)} & A_{22}^{(h)} & \cdots & A_{2K}^{(h)} \\ \vdots & \vdots & \cdots & \vdots \\ A_{K1}^{(h)} & A_{K2}^{(h)} & \cdots & A_{KK}^{(h)} \end{bmatrix}, \quad h = 1, 2, \ldots, p, \quad (4)$$

$$A_{kr}^{(h)} = \sum_{l=0}^{s}\Psi_{kr}^{(hl)}\mathbf{W}_{kr}^{(hl)}, \quad (5)$$

$$\begin{aligned} \Psi_{kr}^{(hl)} &= \operatorname{diag}\left\{\left[\psi_{1kr}^{hl}, \psi_{2kr}^{hl}, \ldots, \psi_{Nkr}^{hl}\right]\right\}, \\ k, r &= 1, 2, \ldots, K, \\ h &= 1, 2, \ldots, p, \\ l &= 1, 2, \ldots, s, \end{aligned} \quad (6)$$

$$W_{kr}^{(hl)} = \begin{bmatrix} w_{kr}^{(hl)}(1,1) & w_{kr}^{(hl)}(1,2) & \cdots & w_{kr}^{(hl)}(1,N) \\ w_{kr}^{(hl)}(2,1) & w_{kr}^{(hl)}(2,2) & \cdots & w_{kr}^{(hl)}(2,N) \\ \vdots & \vdots & \ddots & \vdots \\ w_{kr}^{(hl)}(N,1) & w_{kr}^{(hl)}(N,1) & \cdots & w_{kr}^{(hl)}(N,N) \end{bmatrix}, \quad (7)$$

$$C_h = \begin{bmatrix} \Psi_{11}^{(h0)} & \Psi_{12}^{(h0)} & \cdots & \Psi_{1K}^{(h0)} \\ \Psi_{21}^{(h0)} & \Psi_{22}^{(h0)} & \cdots & \Psi_{2K}^{(h0)} \\ \vdots & \vdots & \ddots & \vdots \\ \Psi_{K1}^{(h0)} & \Psi_{K2}^{(h0)} & \cdots & \Psi_{KK}^{(h0)} \end{bmatrix} + \begin{bmatrix} \Psi_{11}^{(h1)}\mathbf{W}_{11}^{(h1)} & \Psi_{12}^{(h1)}\mathbf{W}_{12}^{(h1)} & \cdots & \Psi_{1K}^{(h1)}\mathbf{W}_{1K}^{(h1)} \\ \Psi_{21}^{(h1)}\mathbf{W}_{21}^{(h1)} & \Psi_{22}^{(h1)}\mathbf{W}_{22}^{(h1)} & \cdots & \Psi_{2K}^{(h1)}\mathbf{W}_{2K}^{(h1)} \\ \vdots & \vdots & \ddots & \vdots \\ \Psi_{KK}^{(h1)}\mathbf{W}_{KK}^{(h1)} & \Psi_{K2}^{(h1)}\mathbf{W}_{K2}^{(h1)} & \cdots & \Psi_{KK}^{(h1)}\mathbf{W}_{KK}^{(h1)} \end{bmatrix} + \cdots + \begin{bmatrix} \Psi_{11}^{(hl)}\mathbf{W}_{11}^{(hl)} & \Psi_{12}^{(hl)}\mathbf{W}_{12}^{(hl)} & \cdots & \Psi_{1K}^{(hl)}\mathbf{W}_{1K}^{(hl)} \\ \Psi_{21}^{(hl)}\mathbf{W}_{21}^{(hl)} & \Psi_{22}^{(hl)}\mathbf{W}_{22}^{(hl)} & \cdots & \Psi_{2K}^{(hl)}\mathbf{W}_{2K}^{(hl)} \\ \vdots & \vdots & \ddots & \vdots \\ \Psi_{KK}^{(hl)}\mathbf{W}_{KK}^{(hl)} & \Psi_{K2}^{(hl)}\mathbf{W}_{K2}^{(hl)} & \cdots & \Psi_{KK}^{(hl)}\mathbf{W}_{KK}^{(hl)} \end{bmatrix}, \quad h = 1, 2, \ldots, p. \tag{8}$$

Expand C_h to yield (8), where h is the time lag order, l is the spatial lag order, coefficient matrix $\Psi_{kr}^{(hl)}$ represents the extent of the influence of variable r on variable k in the h order time lag and l order time lag, and $A_{kr}^{(h)}$ denotes the influence degree of the $r-$th variable on the $k-$th variable in the h time lag.

Full Information Maximum Likelihood (FIML) method is adopted to estimate unknown parameters. This method is suitable for the entire system parameters and, to a certain extent, can deal with all the parameters and variance at the same time. If we can establish the likelihood function accurately, then FIML can estimate all structural parameters by solving the likelihood function of simultaneous equations of the model based on sample observations. FILM estimation can be proved to be asymptotically normal and unbiased if sufficient restrictions are imposed on the parameters.

The coefficients of the model do not have a definite traffic meaning. Therefore, the impulse response is needed for deep analysis. Usually, the estimated SpVAR model mainly analyzes the transmission effect of variables across regions by calculating the impulse response. The impulse response refers to the impact on each explanatory variable when a random disturbance item of the explanatory variable produces a unit of standard deviation shock while other disturbances remain unchanged. The impulse response is used to measure the current value of passenger flow volume in certain area. It is also used to predict the value of passenger volume in other regions when a unit passenger flow impulse is generated in a certain area, allowing us to analyze the degree of interaction between regional transit modes and dynamic equilibrium characteristics. In the SpVAR model of this study, the impulse response mainly refers to the impact of the traffic flow of a transit mode on the passenger flow in a certain area under the following conditions: (1) the impact of the occurrence of regional impact mode; (2) other transit modes in the area of effective impact; (3) impact modes in other areas; (4) other modes in other areas.

The impulse response in the SpVAR model includes general dynamic effect in conventional VAR model and the spillover effect of variables caused by the travel spatial characteristics. In this case, the spillover effect is caused by the spatial hysteresis structure and the spatial autoregressive structure in the model. When the spillover effect is positive, it indicates that the response is the stimulating effect and the spillover effect is negative, indicating that the response is the inhibitory effect.

3.2. Spatial Weight. In the proposed model, each mode is assumed to be related to other modes in each studied traffic zone. Ruan et al. pointed that urban road traffic system is a time-evolving, directed weighted network [16, 17]. The autoregressive coefficients in the SpVAR model are assumed to vary across locations, thereby allowing spatially heterogeneous model dynamics. Spatial weight W is added to the SpVAR model to enhance the SVAR model. The regional comprehensive spillover effect value can be decomposed into each region by spatial weight matrix $W_{kr}^{(hl)}$. In this study, spatial weight $w_{kr}^{(hl)}(i, j)$ denotes the extent of the influence of k transit mode in region i on r transit mode in region j. We set the spatial weights as fixed over time. In the traditional spatial VAR model, the weight matrix is set as expression (9), where d_{ij} denotes the distance between regions i and j. Researchers tend to express d_{ij} by calculating the distance between two geographical center points. Z_j and Z_i represent the spatial scale effects of regions j and region i, respectively. The spatial scale effect above refers to the specific effect value of each area and is related to some specific space, architecture, economy, and other factors in the region. In this regard, we improve the regional scale effect function by incorporating the characteristics of passenger flow in the model.

$$w_{kr}^{(h1)}(i, j) = \frac{1}{d_{ij}} \frac{Z_i}{Z_j + Z_i} \quad (i \neq j) \tag{9}$$

$$w_{kr}^{(h1)}(i, i) = 0 \tag{10}$$

The performance of the transport system is related to the general distribution of land. Land use has greatly affected the degree of public traffic collection and distribution, which is the most important factor that determines the attraction of passenger flow [18, 19]. Therefore, we aim to comprehensively characterize the volume of regional public collection and distribution by the function in which the number of POIs in each region is an independent variable. We then fit the number of POIs with actual bus and metro passenger flow by the following logistic regression function:

$$Z_i = \frac{1}{1 + e^{-\omega_i^T X_i}} \tag{11}$$

where X_i is a vector of six variables and represents the number of POIs in each region that has been normalized to 0-1. The POI data used in this paper mainly include residential, hotel, entertainment, service, business, and tourist facilities. ω_i represents the weights of POIs. The value of Z_i changes not linearly with increasing X_i but shows a smooth change. When the value of X_i is relatively large or small, Z_i

changes slowly, which meets the actual situation in this research.

4. Empirical Analysis

4.1. Stationarity Testing. The primary step of applying SpVAR techniques is conducting unit root test to examine the stationarity properties for each variable. If the variables are not stable, the sequence needs to be differentiated, which may lead to the loss of original information. As a basic criterion of constructing the dynamic regressive model, the cointegration test is then adopted to verify the randomness between nonstationary series and the long-equilibrium relationship in linear combinations.

To implement unit root test, the Augmented Dicky–Fuller (ADF) test is activated in which "the variable has a unit root" is the null hypothesis. After ADF test, the Engle–Granger test is further utilized to examine the cointegration properties of the variables. The results suggest that the residual term of time series is stable as the t-statistic is significant at 1%. Hence, the time series of variables conform to the 0-order cointegration relationship, which satisfies the basic requirement of implementing SpVAR techniques.

4.2. Regression. We develop a SpVAR model for analyzing the metro boarding and alighting volume as well as bus alighting volume in three selected traffic zones.

A number of different types of POIs are used to obtain the spatial weight parameters, as POIs are able to represent as the influential factor to traffic volume. The number of different types of POI, specifically including residential, hotel, entertainment, service, business, and tourist facilities, is aggregated at TAZ.

4.3. Impulse Response Analysis. This section analyzes the impulse response of the SpVAR model for analyzing the transit volume in ZGC, GM, and HLG. The impulse function illustrates the effect of both direct and indirect responses of the change of passenger volume to a shock on the involved transit modes in studied regions. Specifically, a positive response implicates that the shock phenomenon would result in a ridership increase for affected transit mode, and the value of response equals the number of increased passengers. Such ridership increase may be due to a significant positive correlation of travel patterns between two involved regions, in which an increase in one region would easily lead to a correspondent rise to another one. Under such a circumstance, the generated conclusions are important reference for transit operators to take effective measures in advance to prevent unexpected events, for example, a shortage of transit supply, and further to maintain transit system efficiency. On the contrary, a negative impulse response indicates that the ridership in affected region would be reduced due to an appearance of shock in connected region. This decrease may result from a plunge in the transport system efficiency, or the mode shift, or the rerouting behavior caused by a sudden rise of traffic congestion in study area. In this case, the managers are able to dynamic adjust transit supply or allocate more resource to bottleneck for alleviating the problems.

4.3.1. Empirical Results of Impulse Response for Intraregional Transit Modes in Morning Rush Hour. The impact of impulse response on the intraregional transit system is analyzed in this section. In Figure 3, x-axis is the time stamp and y-axis is the impulse response.

Figure 3 shows the response of transit modes, including metro boarding/alighting, and bus alighting volumes when the metro boarding volume has a positive shock at 7 : 00 in involved regions. Figure 3 shows that when the metro boarding volume generates a large passenger flow at morning rush hour, the influence of the intraregional metro alighting volume and bus alighting volume is not significant and will not cause major fluctuation. For the response of metro boarding volume, the impulse responses in ZGC and GM are similar, whereas that in HLG is slightly different. This finding may due to the reason that the first two regions are both the business districts, whereas HLG is the residential area. The metro boarding volume fluctuates over y-axis in the morning and has a small peak at 13 : 00 at noon in HLG which may be because a great number of commuters in residential areas take metro to work in the morning and some commuters will return home to relax and then go to work after noon. In business district, appearance of shock in metro boarding volume at morning rush hours does not influence any other mode and will gradually disappear.

4.3.2. Empirical Results of Impulse Response for Intraregional Transit Modes in Evening Rush Hour. Figure 4 shows the response of the transit system within the regions when the metro boarding volume reflects a positive unit shock at 17 : 00. The response in the evening rush hour is similar to that of the morning peak rush hour. In comparison to the metro boarding volume, the response of metro alighting volume in HLG is relatively larger. The impulse response is positive at the beginning of the first half hour, and then Figure 5 presents the interregional impulse response of MA, MB, and BA to MB in the morning rush hour suggesting the spatial spillover effects of metro boarding volume in each region.

Figure 5 shows the SpVAR impulse responses of the transit system in ZGC and GM to the metro boarding flow in HLG at 7 : 00.

Figures 5(a) and 5(d) suggest that a positive shock of metro boarding volume in HLG will bring a positive impact on the metro boarding volume in two regions, and the volume reaches the highest points, 2.12 and 1.44, at 8 : 00 and 8 : 15, respectively.

4.3.3. Empirical Results of Impulse Response for Interregional Transit Modes in Morning Rush Hour. For the response of MA to MB, Figures 5(b) and 5(e) show that a positive shock of HLG metro boarding volume in morning rush hour has a positive impact on metro alighting volume in both regions. This finding may be due to the fact that ZGC and GM are job

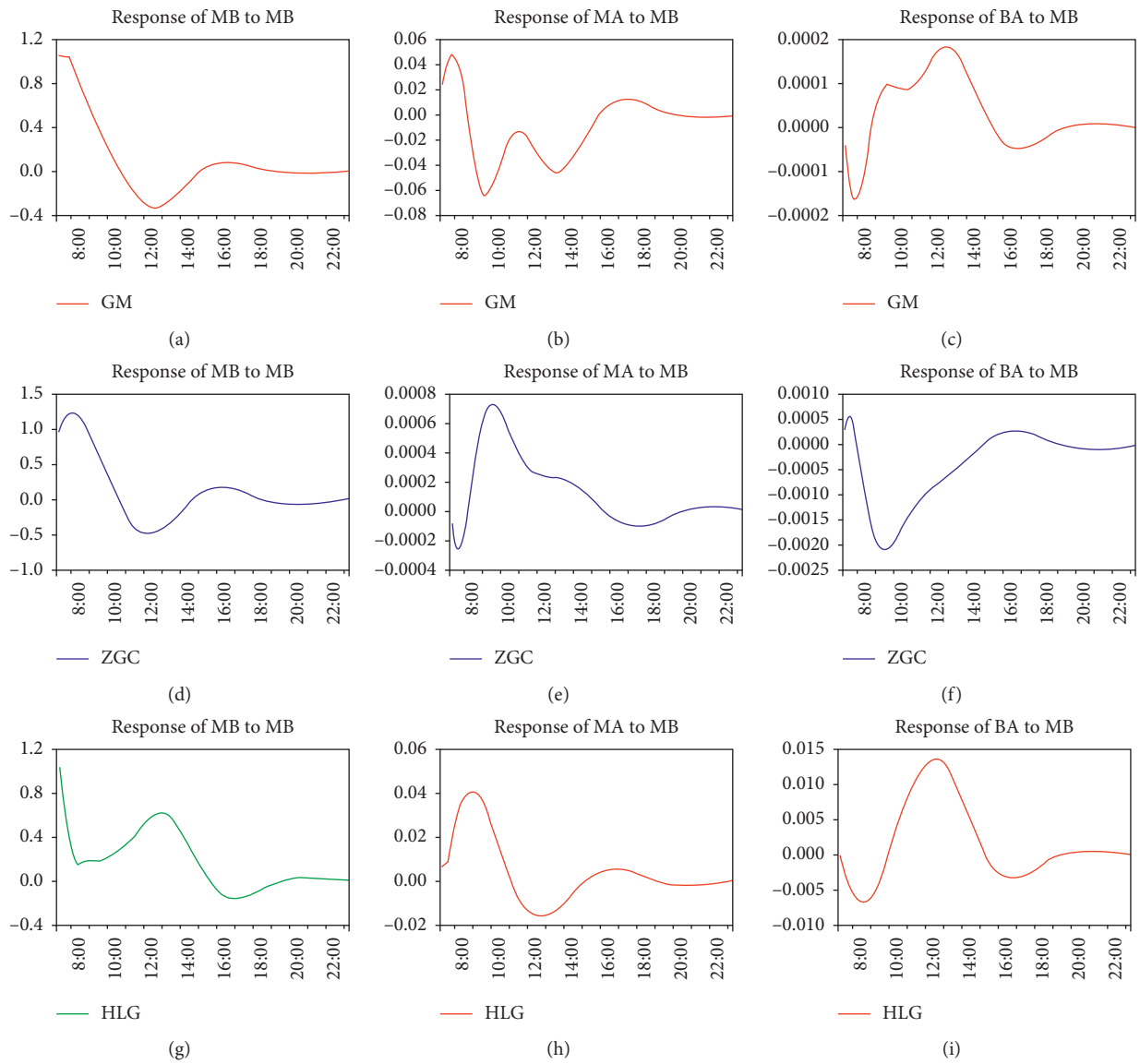


Figure 3: Intraregional passenger flow impulse response at 7 : 00.

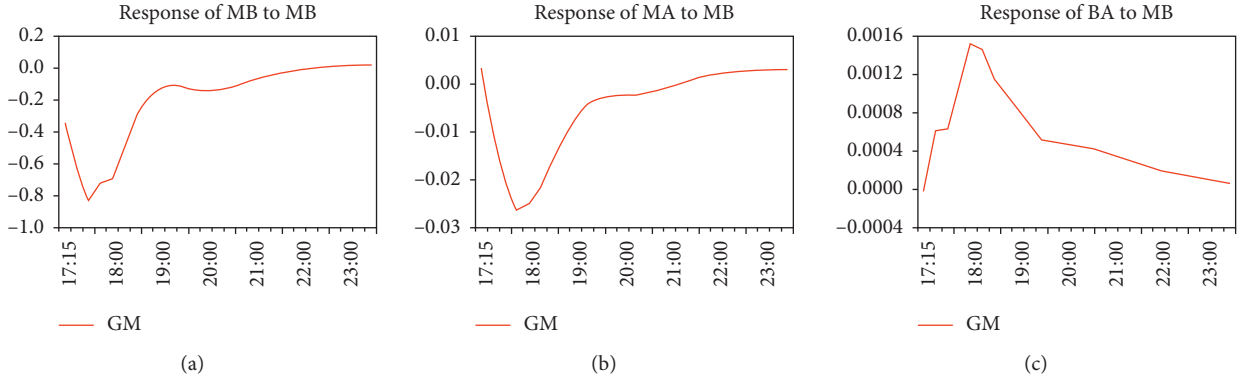


Figure 4: Continued.

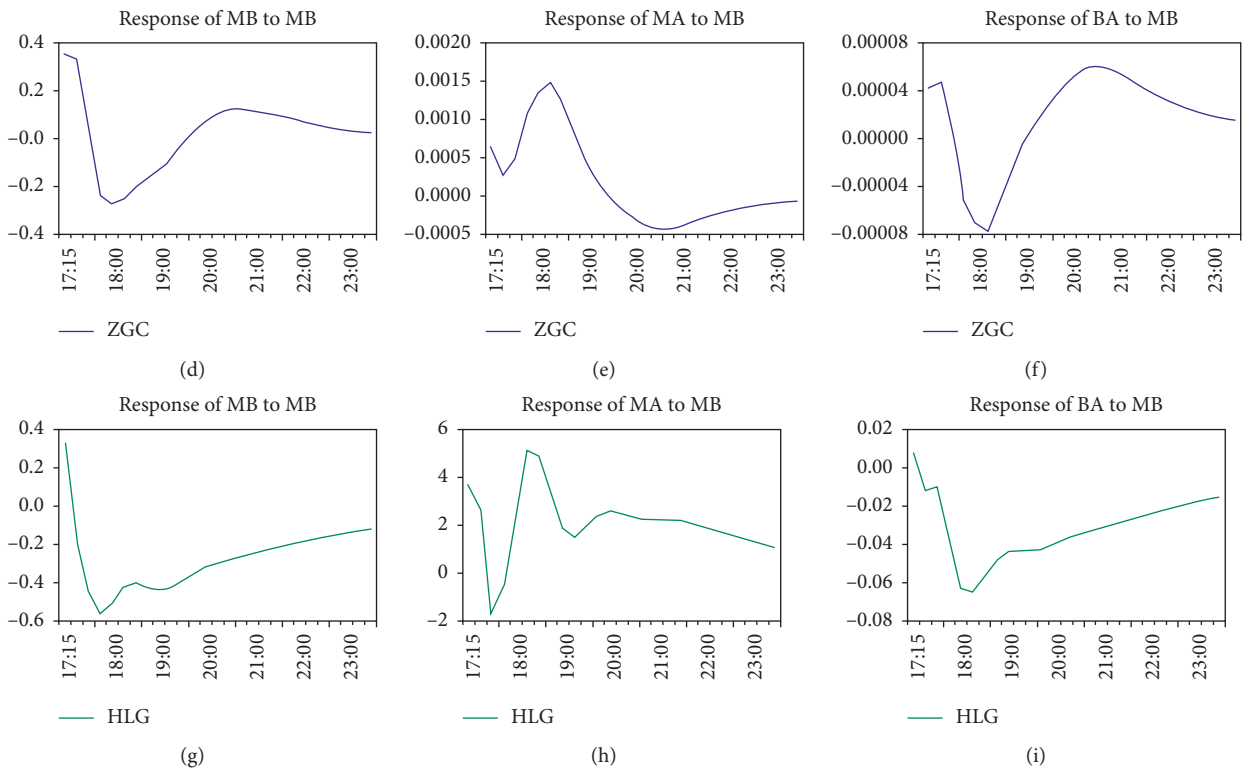


Figure 4: Intraregional passenger flow impulse response at 17 : 00.

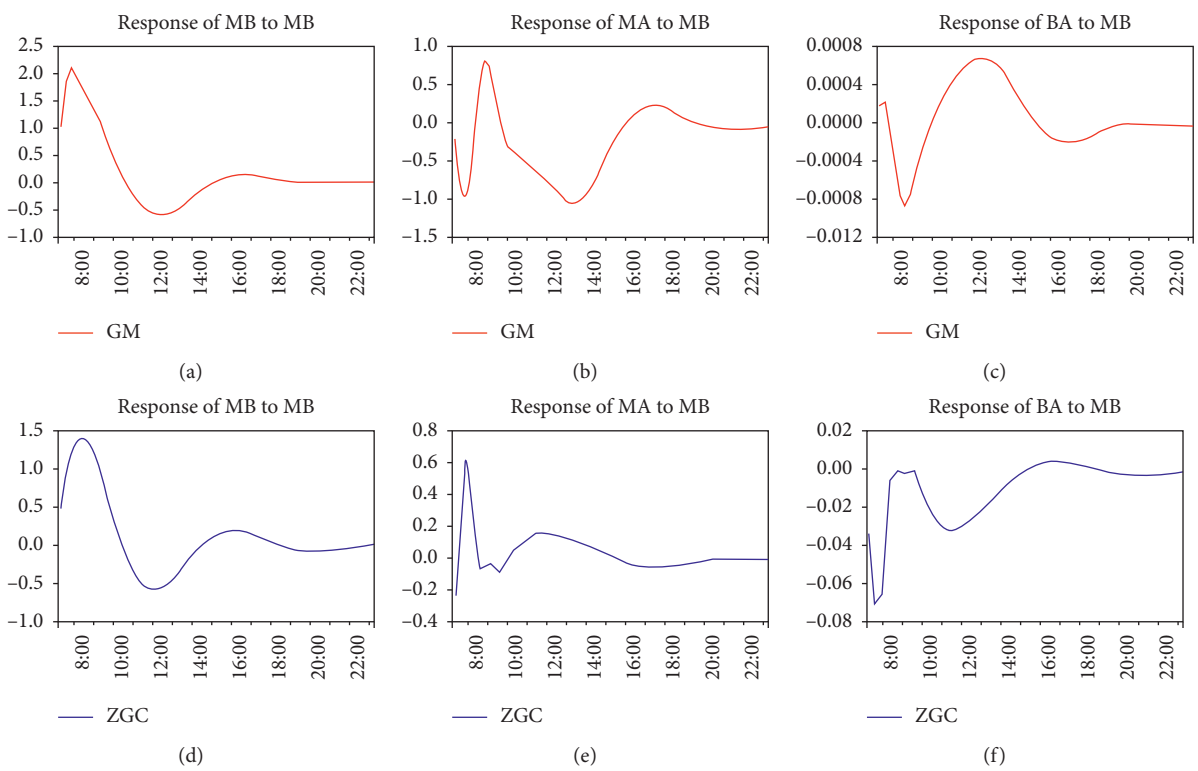


Figure 5: Response to the change in HLG metro boarding volume at 7 : 00.

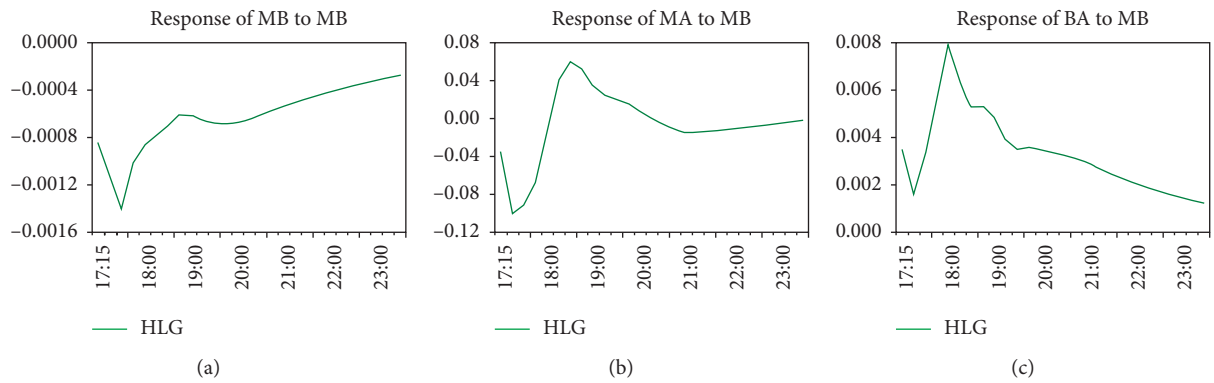


Figure 6: Response to the impact of GM metro boarding volume at 17 : 00. (a) Response of MB to MB. (b) Response of MA to MB. (c) Response of BA to MB.

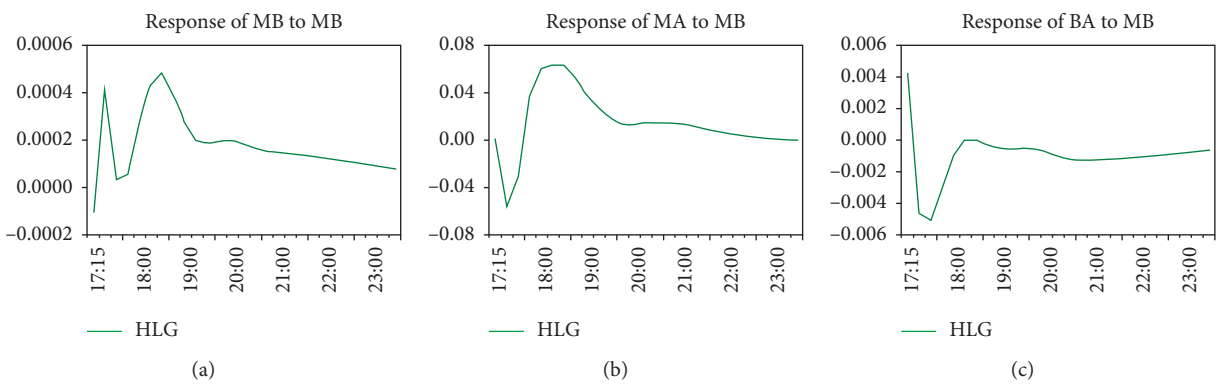


Figure 7: Response to the change in ZGC metro boarding volume at 17 : 00. (a) Response of MB to MB. (b) Response of MA to MB. (c) Response of BA to MB.

areas, which would attract a large number of commuters from the residential zones, likely HLG in the morning rush hour. The influence of metro alighting volume in ZGC and GM reaches the maximum at 7 : 45 and 8 : 45, respectively. Notably, the travel times from HLG to ZGC and GM are around 45 mins and 105 mins, respectively, when the trip is started at 7 : 00 AM. These travel times will match the time difference between the peak hours of ZGC and GM (7 : 45 vs. 8 : 45) in the morning peak hour. Such observation is able to be a convincing evidence for proving the spatial spillover effect within three study zones. As such, the transit operators could predict the traffic conditions in ZGC and GM with referring to HLG's impulse response in advance and further to take effective measures to better contend with any unexpected scenario.

Figures 5(c) and 5(f) show that a positive shock to HLG metro boarding volume has a small negative impact on bus alighting volume in ZGC and little impact on GM. Given that the distance between ZGC and HLG is not too far, bus and metro are competitive with each other, and a part of bus users may shift to taking metro, which then results in a decrease of bus ridership.

Figure 5 shows that the metro boarding flow in region HLG has a strong spatial spillover effect on other modes in ZGC and GM.

4.3.4. Empirical Results of Impulse Response for Interregional Transit Modes in Evening Rush Hour. Figures 6 to 7 present the interregional impulse response form MA, MB, and BA to MB in the evening rush hour. Generally, the spatial spillover effect of the evening rush hour is not significant as that observed in the morning.

Figures 6 and 7 depict the response of the metro boarding, metro alighting, and bus alighting in HLG to a change in the metro boarding volume in GM and ZGC. In terms of the response from MA to MB, the impulse responses are negative at the beginning and then turn positive, reaching the highest point at 18 : 15 and 18 : 45 in ZGC and GM, respectively. Such observation is similar to the case of morning peak. Specifically, after a short period of the metro boarding volume increases in the first hour, the bus alighting volume accordingly increases in GM and decreases in ZGC which lasts for a long time. This finding is also relevant with the property of study zones since the commuters tend to be back home in the evening rather than generating trips from home addresses in the evening.

5. Conclusions

This study proposes an enhanced SpVAR model to explore the time-varying relationships and capture spatial spillover

effects between bus passenger volume and metro passenger volume among different regions. The proposed model features in taking spatial weights and the impulse response into account so that it is able to study the propagation mechanism of interregional passenger flow under impacts, such as emergencies. Three representative zones in Beijing, Huilongguan (HLG), Zhongguancun (ZGC), and Guomao (GM), are employed to study and capture the spatial spillover effect and the its mechanism in the multimodel transit system in Beijing. The statistical results confirm the existence of the spatial spillover effect in these selected Beijing commuter regions in which the changes in transit system in one region will generate some significant impacts on other modes in another two regions. Those findings can be applied to predict unexpected events and improve the decision-making process for traffic management departments in terms of warning, guiding, or limiting the passenger flow in case of a sudden large passenger flow burst. For example, public transit agencies will be suggested to increase the dispatching frequency when the model result shows a positive impulse response in the bus boarding patron. The metro company will be warned to take actions to better manage mass metro patron in case of a positive impulse response in the metro system. The main contributions of the paper are as follows: (1) The enhanced spatial VAR (SpVAR) model is proposed and then firstly introduced into analyzing the spatial interactions among the multimodal transit systems. Specifically, a spatial weight matrix is developed to be embedded into the proposed model, while the model is able to explore the impulse response that is used to analyze the impact of the change of passenger volume in one mode and in region on multitransit modes in other separated regions. (2) This paper empirically analyzes the multimodal spatial spillover effects within three separated traffic districts during morning and evening peaks as well as the off-peak periods, to explore the interactions in multimodal transit system.

Although the proposed model is capable of analyzing the time-varying relationships within multimodal system in different regions, some critical issues still need to be further investigated: (1) the accuracy of the proposed model highly depends on the size of test data. As a result, how to improve the capability of estimation techniques under small dataset is one of next research directions; (2) the proposed model is unable to include the features and elements of physical links between regions. Since the spatial weights have been proved to be significantly correlated with physical link between different regions, adding physical link or travel impedance into model and further to verify its effectiveness will be another main topic in the future research.

The future research directions may include (a) the extension of the SpVAR model to leverage dynamic panel data models and capture prior knowledge of some parameters; (b) modeling the nonlinear relationships and interactions of the transit modes and TAZs; and (c) enhancing the SpVAR considering the error correction, time-varying patterns, and the factor augmentation.

Data Availability

All types of data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This paper was supported by National Key R&D Program of China (2016YFE0125000) and the National Natural Science Foundation of China (61903058 and 61773036).

References

[1] H. Peng and H. Lu, "Study on the impacts of urban density on the travel demand using GIS spatial analysis," *Journal of Transportation Systems Engineering and Information Technology*, vol. 7, no. 4, pp. 90–95, 2007.

[2] J. C. Cardona, R. Stanojevic, and R. Cuevas, *On Weather and Internet Traffic Demand*, Imdea Networks Institute, Madrid, Spain, 2012.

[3] J. Chai, Y. Yang, Q. Y. Lu, and L. M. Xing, "Analysis of influence factors of road transportation demand based on bayesian structural equation model," *Chinese Journal of Management Science*, vol. 23, pp. 286–390, 2015.

[4] G. R. Patil and B. K. Bhavathrathan, "Effect of traffic demand variation on road network resilience," *Advances in Complex Systems*, vol. 19, no. 01n02, Article ID 1650003, 2016.

[5] L. Sun, K. W. Axhausen, D.-H. Lee, and X. Huang, "Understanding metropolitan patterns of daily encounters," *Proceedings of the National Academy of Sciences*, vol. 110, no. 34, pp. 13774–13779, 2013.

[6] L. Sun, K. W. Axhausen, D. Lee, and M. Cebrian, "Efficient detection of contagious outbreaks in massive metropolitan encounter networks," *Scientific Reports*, vol. 4, p. 5099, 2014.

[7] L. Sun and K. W. Axhausen, "Understanding urban mobility patterns with a probabilistic tensor factorization framework," *Transportation Research Part B: Methodological*, vol. 91, pp. 511–524, 2016.

[8] T. G. Conley and B. Dupor, "A spatial analysis of sectoral complementarity," *Journal of Political Economy*, vol. 111, no. 2, pp. 311–352, 2003.

[9] K. Neusser, "Interdependencies of US manufacturing sectoral TFPs: a spatial VAR approach," *Journal of Macroeconomics*, vol. 30, no. 3, pp. 991–1004, 2008.

[10] M. Beenstock and D. Felsenstein, "Spatial vector autoregressions," *Spatial Economic Analysis*, vol. 2, no. 2, pp. 167–196, 2007.

[11] V. D. Giacinto, "On vector autoregressive modeling in space and time," *Journal of Geographical Systems*, vol. 12, no. 2, pp. 125–154, 2010.

[12] J. Ramajo, M. A. Márquez, and G. J. D. Hewings, "Spatio-temporal analysis of regional systems: a multiregional spatial vector autoregressive model for Spain," *International Regional Science Review*, vol. 40, pp. 1–22, 2013.

[13] M. Deng, "Bayesian variable selection in a large vector autoregression for origin-destination traffic flow modelling," in *Spatial Econometric Interaction Modelling*, pp. 199–223, Springer, Cham, Switzerland, 2016.

[14] P. Chen, C. Ding, G. Lu, and Y. Wang, "Short-term traffic states forecasting considering spatial-temporal impact on an

urban expressway," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2594, no. 1, pp. 61–72, 2016.

[15] M. Beenstock and D. Felsenstein, "Spatial vector autoregressions," in *The Econometric Analysis of Non-stationary Spatial Panel Data*, pp. 129–161, Springer, Cham, Switzerland, 2019.

[16] Z. Ruan, C. Song, X.-h. Yang, G. Shen, and Z. Liu, "Empirical analysis of urban road traffic network: a case study in Hangzhou city, China," *Physica A: Statistical Mechanics and Its Applications*, vol. 527, Article ID 121287, 2019.

[17] S. Yang, J. Wu, Y. Xu, and T. Yang, "Revealing heterogeneous spatiotemporal traffic flow patterns of urban road network via tensor decomposition-based clustering approach," *Physica A: Statistical Mechanics and Its Applications*, vol. 526, Article ID 120688, 2019.

[18] S. Tanuja, I. W. H. Ho, and K. T. Chi, "Spatial analysis of bus transport networks using network theory," *Physica A Statistical Mechanics & Its Applications*, vol. 502, 2018.

[19] Y. Wang, J. Cao, W. Li, T. Gu, and W. Shi, "Exploring traffic congestion correlation from multiple data sources," *Pervasive & Mobile Computing*, vol. 41, pp. 470–483, 2017.

Hindawi
Journal of Advanced Transportation
Volume 2020, Article ID 8820680, 17 pages
https://doi.org/10.1155/2020/8820680

Research Article

Assessing Urban Road Network Capacity considering Parking Supply and Parking Pricing

Xiaowei Jiang,[1] Muqing Du ,[2] and Haisheng Liu[2]

[1]*School of Business, Jinling Institute Technology, Nanjing 211100, China*
[2]*College of Civil and Transportation Engineering, Hohai University, Nanjing 210098, China*

Correspondence should be addressed to Muqing Du; dumuqing@gmail.com

Received 7 April 2020; Revised 11 August 2020; Accepted 15 October 2020; Published 29 October 2020

Academic Editor: Zhao Xing

Copyright © 2020 Xiaowei Jiang 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 enhance the assessment of the network capacity for a given urban road system, the effects of the parking management strategies at destination areas are supposed to be considered in the network capacity assessment model. This study provides an extended road network capacity model which takes into consideration both the parking supply and parking pricing at each traffic zone. The network capacity model is formulated as a bilevel programming problem, with the maximization of total trip generation in the upper level and the combined trip distribution and traffic assignment (CTDTA) problem in the lower level. To reasonably characterize the impacts of the parking pricing and parking delay due to the congestion effect, two classes of travel demand are involved in the CTDTA model. An efficient and practical algorithm is provided for the solution of the bilevel network capacity model. Numerical experiments show the advantages of the proposed model and also demonstrate the effect of the parking supply and parking pricing on the assessment results of the road network capacity.

1. Introduction

The rapid growth of motor vehicle travel demand leads to most cities suffering from traffic congestion and parking space shortage. The road network capacity assessment is helpful to reflect the capability of the urban road system for allocating the maximum travel demand by motorists. Hence, to properly assess the capacity of the urban road network has long been an important goal for transportation project planning and evaluation. In the literature, most research explores the transportation network capacity problem by only considering the constraints from the link capacities (e.g., [1, 2]). However, a few studies indicate that involving the impact of parking management strategies can help to obtain more practical and reasonable results for the urban road network [3, 4]. On the other hand, assessing the network capacity with the parking management will be meaningful to properly coordinate the parking resource and motor travel demand in the level of the road network. The network-wide origin-destination (O-D) demand could redistribute in space due to the implementation of the new parking management strategies. Thereby, to incorporate the parking supply and/or parking pricing into the transportation network capacity problem will be of theoretical and practical significance for the decision making in urban parking management.

The concept of network capacity is defined as the maximum travel demand that can be allocated to the transportation network without violating the capacity of any individual component (e.g., link and intersection) [5]. The network capacity has been recognized as an important measure to assess the network-wide performance in the urban transportation system. This concept has been widely used in many applications, such as capacity reliability analysis [6], network design problem [7], capacity flexibility analysis [8], and capacity vulnerability analysis [9]. Early studies assessed road network capacity by considering the capacity constraint on each road segment [10]. The maximum capacity was estimated by assigning the total O-D demand gradually, in which every slice of the O-D demand is loaded with an equilibrium traffic assignment [11]. Following that, the traffic signal splits at intersections were

incorporated into the network capacity problem. The concept of reserve capacity is proposed [1] and extended in the study of the signal-controlled network [12–14]. Since the reserve capacity model assumes the O-D demand distribution to be unchanged, which is not practical for the networks in the developing areas, variable O-D demand was involved in the network capacity assessment [2, 5, 15]. On the other hand, as the distribution of the total O-D travel demand (compared with that of the motor O-D demand) is relatively stable, the multimodal choice was extended based on the reserve capacity concept [16, 17], in which the total demand with a fixed O-D distribution splits into several variable O-D tables which are separately associated with the different travel modes. Besides, other studies explored the impact of market penetration of the advanced traveler information system (ATIS) on the road network capacity [18]. Only limited research can be found for the impact of the parking supply or parking pricing on the transportation network capacity [3, 4, 19].

In the assessment of the urban road network capacity, the parking constraints are twofold: (1) the parking supply, which determines both the upper bound of the parking demand and the parking time due to the level of congestion in each trip destination; (2) the parking pricing, which decides the parking fee at each destination area. Note that the upper limit of the parking demand can be regarded as the parking capacity that limits the growth of the travel demand by motorists, which is the same as the role of the link capacity. The parking time and fee at destinations are supposed to affect some travelers' travel choices, which leads to the redistribution of the O-D travel demand network wide. According to the existing studies, Asakura and Kashiwadani [19] and Leng and Yan [4] only consider the effect of the parking capacity on the network-wide capacity, and thus, the change on O-D demand pattern was not considered. Although Tam and Lam [3] involved both parking capacity and parking charge at destinations, the entire O-D trip distribution was assumed to only depend on the travel cost (specifically the accessibility measures) between the O-D pair, i.e., the level of destination congestion was not considered. Nevertheless, the parking time due to the level of destination congestion has been studied in the context of parking choice problems [20–23]. The parking (search) time is often defined as an increased function in terms of the parking demand dividing the parking capacity at the destination area. Xie et al. [24] once employed the parking time function in their study on the network capacity. They assumed all travel demands are variable under the impact of the travel cost at destinations, which could be loss of generality because the commuting demand may not willing to change their destination even the destination cost is very high.

In summary, the existing research on the road network capacity considering parking supply and parking pricing has the following inadequacies: (1) only some of the factors were involved in the network capacity assessment, so the effect of the urban parking management strategies cannot be evaluated comprehensively [19]; (2) incomplete to characterize the impact of destination congestion on the network-wide travel demand pattern [3], which may result in impractical O-D demand pattern for the maximum network capacity; (3) not consider the difference among the motor travelers and, thus, causes the underestimate [4] or overestimate [24] to the total network capacity. In this study, we will characterize both the parking supply and parking pricing and incorporate them into a combined trip distribution and traffic assignment (CTDTA) model [25] which involves two classes of motor travel demand, i.e., the fixed demand and the variable demand. We refer the fixed demand to be the commuting motorists which will hold a stable O-D pattern under the impact of destination parking cost and refer the variable demand to be the non-commuting motorists which show a variable demand pattern with respect to the level of congestion in both the road network and the destination areas. The new model is expected to better capture the effect of both parking supply and parking pricing on the urban road network. The application of the new model will be more practical and helpful for assessing the capacity of the given road network, especially for the demonstration of strategies in urban parking management.

The remaining of this paper is organized as follows. The next section presents the new urban road network capacity model with parking supply and parking demand. Then, section 3 provides an efficient solution algorithm for our proposed model formulated as bilevel programming. Numerical experiments are conducted in section 4 to demonstrate the feature of the new model and the effect of parking supply and parking demand on the network capacity assessment. Finally, conclusions are summarized in section 5.

2. The Extended Road Network Capacity Model

2.1. Model Assumptions. The following assumptions are made to facilitate the presentation of the essential ideas in this study:

(a) Two classes of travel demand by motorists are involved, i.e., the fixed demand and variable demand. The fixed demand is related to the daily travel demand that has predetermined origins and destinations, such as the commuting traffic. The pattern (trips and spatial distribution) of the fixed demand will be stable regardless of the change of the travel impedance or cost from the origin to destination. On the contrary, the destinations of the variable demand are exchangeable with respect to the crowdedness and costs at the destinations, and thus, such kind of demand (such as shopping and recreation) shows a flexible demand pattern.

(b) The travelers associated with the fixed demand can only change the paths to minimize the individual travel cost, which is characterized by the user equilibrium principle. The travelers associated with the variable demand can change both the paths and destinations to minimize the total travel cost, which will be characterized by the combined trip distribution and traffic assignment model.

(c) The parking demand is a ratio of the total destination demand at a destination area, and it is restrained by the space of the parking facilities.

(d) Travelers have sufficient network information to choose travel paths in a user equilibrium manner.

(e) Due to the static nature of the model, the traffic flow is assumed in a steady state, and the study period is assumed to be one hour, such as the morning peak hour.

2.2. Model Formulation. We consider a directed network $G = (N, A)$, where N is the set of all nodes and A is the set of all directed links. Let I and J be the set of origins and the set of destinations, respectively. Let R_{ij} be the set of paths between the O-D pair (i, j), $\forall i \in I$, $j \in J$. The traffic flow on link $a \in A$ is denoted by v_a. The travel time on link a is an increasing function of the traffic flow through itself, i.e., $t_a = t_a(v_a)$. The variable travel demand between the O-D pair (i, j) is denoted by q_{ij}, and the fixed demand is denoted by e_{ij}. Let f_r^{ij} and h_r^{ij} represent the traffic flow on path r corresponding to the variable demand and fixed demand, respectively.

Since both the level and distribution of the variable demand are changeable with respect to the traffic situation in the transport network, the utility between origins and destinations will be utilized to derive the flow pattern of the variable demand. Hence, we define the utility from origin i to destination j by U_{ij}, which consists of the generalized destination cost M_j and the minimum travel cost τ_{ij} from i to j, i.e., $U_{ij} = -\tau_{ij} - M_j$. Specifically, M_j is defined as an increasing function of the total travel demand, D_j, at the destination j. As the noncommuting travelers at each origin i may have certain randomness on the destination choice, we employ the logit choice model to produce the conditional probability that a traveler from origin i will choose destination j, which is given by

$$P(j|i) = \frac{\exp\{-\theta(\tau_{ij} + M_j)\}}{\sum_{n\in J}\exp\{-\theta(\tau_{in} + M_n)\}}, \quad \forall i \in I, \tag{1}$$

where θ is the dispersion parameter that reflects the traveler's sensitivity to the travel utility.

In equation (1), the minimum travel cost τ_{ij} is determined by the network-wide traffic flow pattern in the context of the congestion. Generally, it can be conducted from the network equilibrium models. In this study, we obtain it by using an equilibrium combined trip distribution and traffic assignment model which can cope with the mixture of the fixed and variable demand. Besides, the generalized destination cost M_j will be incorporated with the pricing and the searching time in the parking area. The details will be presented in the remainder of this section. Our network capacity model will be formulated in a bilevel framework, where the upper-level problem maximizes the throughput of the network and the lower-level problem regulates the travel choice behaviors of the network users (for both the variable and fixed demand).

2.2.1. Parking Supply and Parking Pricing. Let q_j be the total occupancy of parking $j \in J$ under equilibrium, and let k_j be the capacity of parking j. Note that parking occupancy q_j is obtained by multiplying the total travel demand, D_j, at area j with the parking rate ρ_j, i.e.,

$$q_j = \rho_j \cdot D_j, \quad \forall j \in J, \tag{2}$$

where the total travel demand at destination j will be given by $D_j = \sum_{i\in I}(q_{ij} + e_{ij})$.

Furthermore, we assume the parking search time is a convex function of parking occupancy q_j and parking capacity k_j [26]. Thus, the parking search time function $S_j(q_j)$ can be formulated as

$$S_j(q_j) = \frac{\mu_j s_j}{1 - q_j/k_j}, \quad \forall j \in J, \tag{3}$$

where s_j is the average searching time in destination area j when the parking occupancy is low, referred to as the low-occupancy searching time. μ_j is a constant denoting how drivers' attitudes to parking occupancy information. If drivers are unaware of the searching time, set $\mu_j = 0$; if drivers are completely aware of searching time, set $\mu_j = 1$. Note that the searching time reaches infinity as parking occupancy q_j approaches parking capacity k_j. This indicates that drivers arriving at a full occupancy parking area will never find a parking space.

Although the parking time function in equation (3) nicely captures the effect of parking demand versus parking supply, the calculation of this function can lead to infeasible solutions if the parking occupancy is equal or greater than the parking capacity. Alternatively, the BPR-type functions were often employed [21, 23] to avoid the discontinuous phenomenon when the parking demand is approaching the capacity. We recommend the following formulation in practice:

$$S_j(q_j) = s_j\left[1 + \varphi\left(\frac{q_j}{k_j}\right)^{\omega}\right], \quad \forall j \in J, \tag{4}$$

where φ and ω are calibration parameters. When $q_j \longrightarrow k_j$, the parking search time increases to the $(1 + \varphi)$ times of the low-occupancy searching time s_j.

The hourly parking price at area j is denoted by π_j. Thus, the drivers who choose area j as a travel destination will pay π_j dollars per hour. Therefore, the generalized destination cost, M_j, for travelers who choose area j as their destination, is given by

$$M_j = \pi_j + \eta \cdot S_j(q_j), \tag{5}$$

where η is the value of time for travelers. Generally, the values of π_j are various for different destination areas, which depends on the industrial structure, land use, and locations. To reflect the differences among the destination areas, a more detailed traffic zone division will be implemented in practice.

From the aspect of parking supply constraint, the parking occupancy should not exceed the parking capacity in area i, that is,

$$q_j \le k_j, \quad \forall j \in J. \tag{6}$$

2.2.2. Combined Trip Distribution and Traffic Assignment Model. In this section, the lower-level model of the network capacity problem will be formulated by considering the parking cost at destinations. Following the assumptions in Section 2.1, we consider the travelers can choose the destinations and paths to minimize the travel cost accordingly. The combined trip distribution and traffic assignment (CTDTA) model [25] will be employed to capture travelers' behavior. Specifically, both the fixed demand and variable demand are considered in the road network. The travelers with fixed demand will be free to choose their travel paths, while the travelers with variable demand will be free to choose either the paths or the destinations. Both types of travelers aim to minimize their own travel cost for the whole trip. Furthermore, the CTDTA model will be extended to involve the parking supply and parking pricing at destinations, which will impact the destination choice of the variable travel demand. Thus, the extended CTDTA model is given as follows:

$$\min \quad Z_L(\mathbf{q}, \mathbf{f}, \mathbf{h}) = \sum_{a \in A} \int_0^{v_a} t_a(x)\mathrm{d}x + \frac{1}{\theta} \sum_{i \in I} \sum_{j \in J} q_{ij}\left(\ln q_{ij} - 1\right) + \sum_{j \in J} q_j \pi_j + \eta \sum_{j \in J} \int_0^{q_j} S_j(y)\mathrm{d}y, \tag{7}$$

$$s.t. \quad \sum_{j \in J} q_{ij} = O_i, \quad \forall i \in I, \tag{8}$$

$$\sum_{r \in R_{ij}} h_r^{ij} = e_{ij}, \quad \forall i \in I,\ j \in J, \tag{9}$$

$$\sum_{r \in R_{ij}} f_r^{ij} = q_{ij}, \quad \forall i \in I,\ j \in J, \tag{10}$$

$$v_a = \sum_{i \in I} \sum_{j \in J} \sum_{r \in R_{ij}} \left(f_r^{ij} + h_r^{ij}\right)\delta_{a,r}^{ij}, \quad \forall a \in A, \tag{11}$$

$$f_r^{ij} \ge 0, \quad \forall r \in R_{ij},\ i \in I,\ j \in J, \tag{12}$$

$$h_r^{ij} \ge 0, \quad \forall r \in R_{ij},\ i \in I,\ j \in J, \tag{13}$$

$$q_{ij} \ge 0, \quad \forall i \in I,\ j \in J, \tag{14}$$

where $\delta_{a,r}^{ij}$ is the link/path incidence indicator, which equals to 1 if link a is on path r between the O-D pair (i, j) and 0 otherwise. In the objective function, the first term indicates the user equilibrium of all travelers. The second term indicates the destination choice behavior of the variable demand only. The third and the fourth terms indicate the parking charging and the parking searching time at the destination area, respectively. Equations (8)–(10) define the flow conservation of the variable and fixed demand. Equation (11) is the relationship between link flows and path flows. Equations (12)–(14) are nonnegative constraints. Besides, according to the first-order necessary condition, the O-D flows of the variable demand is produced by the following logit share model:

$$q_{ij} = O_i \cdot \frac{\exp\left\{-\theta\left(\tau_{ij} + \pi_j + S_j\left(D_j\right)\right)\right\}}{\sum_{n \in J} \exp\left\{-\theta\left(\tau_{in} + \pi_n + S_n\left(D_n\right)\right)\right\}}, \quad \forall i \in I,\ j \in J, \tag{15}$$

where the parking cost function, $S_j(D_j)$, is defined in equation (4).

2.2.3. Total Trip Maximization Model. The upper-level problem defines a maximum trip production model, where the limits of the individual link capacity and zonal parking supply are considered. Thus, the upper-level problem is given by

$$\max Z_U(\widetilde{O}) = \sum_{i \in I} \widetilde{O}_i, \tag{16}$$

$$s.t. v_a(\widetilde{O}) \le C_a, \quad \forall a \in A, \tag{17}$$

$$\rho_j \cdot \sum_{i \in I} \left(q_{ij}(\widetilde{O}) + e_{ij}\right) \le k_j, \quad \forall j \in J, \tag{18}$$

$$\widetilde{O}_i \ge 0, \quad \forall i \in I. \tag{19}$$

where $v_a(\widetilde{O})$ and $q_{ij}(\widetilde{O})$ are obtained by solving the lower-level model. The objective is to maximize the summation of the trip production of the variable demand, $\widetilde{O}_i$, from each origin. Equation (17) represents that the traffic flow on each link should not exceed its capacity. Equation (18) represents that the parking demand at each traffic zone should not be over its parking space.

3. Solution Algorithm

In the previous section, the network capacity model considering parking pricing and supply was formulated as a bilevel model. Moreover, the lower-level model integrates two classes of travel demand associated with different travel choice behaviors. The solution of the proposed network capacity model is not trivial. Therefore, in order to apply the proposed network capacity model in practical applications, we design an improved sensitivity analysis-based (SAB) algorithm that enables to solve the bilevel problem efficiently. The SAB algorithm was widely used for solving the bilevel programming model [27]. The sensitivity analysis is specific to the anlaytical senstivity analysis of the lower-level model, which is used to produce the derivatives of the upper-level decision variables with respect to the lower-level one. The derivatives will be used to explicitly reflect the relationship between the upper level and lower level and help to solve the bilevel model iteratively. In practice, the SAB algorithm usually performs a fast convergence which makes the algorithm terminate within a considerable time. Given the nonconvexity of the bilevel problem, the SAB algorithm will converge to a local optimal point. However, as shown by Yang et al. [2], for the bilevel model, if the upper-level objective function is a linear function

of its decision variables, the SAB algorithm can, at least, find a noninferior optimal solution. Therefore, the SAB approach is capable to find a satisfying solution for the proposed network capacity problem in this study. The accuracy of the solution is sufficient to be used for transportation project forecasting or evaluation.

3.1. Path-Based Algorithm for Solving the Lower-Level Model. According to the typical process of the SAB algorithm [27], the lower-level model should be resolved in every iteration to obtain the equilibrium traffic flow pattern with respect to the current upper-level solution. Note that there are two classes of travel demand with different O-D distributions involved in the extended CTDTA model, while their path choice principles are the same. It is general to suppose that the path flows of both the fixed and the variable demand have the same proportion on each equilibrated path. Thus, we used the total path flow, F_r^{ij}, as the main variable in the traffic assignment stage. The ratio of the fixed demand to the variable demand on each path is obtained by e_{ij}/q_{ij}. Moreover, in this section, we embed the gradient projection (GP) method [28] in Evans' algorithm [29] for solving the extended CTDTA model, in which the O-D flow is updated in an additional stage by dealing with a one-dimension search problem. Hence, we define the *path flow proportion*, γ_{ijr}, to denote the portion of the total demand between O-D (i, j) that is assigned to path r. This definition has the merit that the value of γ_{ijr} does not need to be rescaled when the O-D demand is updated. In addition, the GP method is capable to quickly converge to sufficiently high accuracy. The GP method is classified as the path-based algorithm, which enables to quickly extract the set of all equilibrated paths from the path-based optimal solution. The path set will be used as a precondition to in the next stage for the analytical sensitivity analysis approach of the lower-level model.

The implementation of modified Evans' algorithm consists of two stages: the traffic assignment stage and the trip distribution stage. The former is implemented to equilibrate path flows while fixing the current O-D demand; the latter performed to update variable O-D demand while restraining the current path flow proportion. We carry out the traffic assignment stage as an inner loop with certain times, so as to accelerate the convergence of the whole algorithm. Besides, when producing a new (auxiliary) O-D flow, the parking search time is derived in advance based on the O-D flow pattern in the previous iteration, i.e., $S_j^n := S_j(q_j^{n-1})$.

The solution procedure for the extended CTDTA model is presented as follows:

Initialization: we set $n := 0$ and determine the initial link cost $\{t_a^0\}$ and parking time $\{S_j^0\}$ by setting link flows $v_a = 0, \forall a \in A$ and variable O-D demand $q_{ij} = 0, \forall i \in I, j \in J$, respectively. For each origin $i \in I$,

(1) find the minimum-cost paths starting from origin i. For each minimum cost path $\bar{r}$ to destination $j \in J$, we denote the minimum cost by τ_{ij}^0, and let $R_{ij}^0 := R_{ij}^0 \cup \{\bar{r}\}$.

(2) compute the initial variable OD demands by

$$q_{ij}^0 = O_i \cdot \frac{\exp\left\{-\theta\left(\tau_{ij}^0 + \pi_j + S_j^0\right)\right\}}{\sum_{n\in J}\exp\left\{-\theta\left(\tau_{in}^0 + \pi_n + S_n^0\right)\right\}}, \quad \forall i \in I,\ j \in J. \tag{20}$$

$$\Delta F_r = \frac{c_{ijr} - \tau_{ij}}{\sum_{a\in A(r,\bar{r})}\partial t_a/\partial v_a}, \quad \forall r \in R_{ij},\ r \neq \bar{r}, \tag{21}$$

$$\gamma_{ijr}^n = \frac{F_r^{ij}}{e_{ij} + q_{ij}^n}, \quad \forall r \in R_{ij},\ i \in I,\ j \in J. \tag{22}$$

Step 2: we update variable O-D flows, retaining the path flow proportions, given the path flow proportions $\{\gamma_{ijr}^n\}$, link flows $\{v_a^n\}$, and path costs $\{c_{ijr}^n\}$ obtained in Step 1.

Step 2.1: for each destination j, we compute the parking search time $S_j^n(\cdot)$ based on the O-D demand $\{e_{ij}\}$ and $\{q_{ij}^{n-1}\}$.

Step 2.2: for each O-D pair, we optimize the variable O-D demand:

(1) The set of auxiliary trip demands $\{\hat{q}_{ij}^n\}$ is found by the following logit share model:

$$\hat{q}_{ij}^n = O_i \cdot \frac{\exp\left\{-\theta\left(\tau_{ij}^n + \pi_j + S_j^n\right)\right\}}{\sum_{k\in J}\exp\left\{-\theta\left(\tau_{ik}^n + \pi_k + S_k^n\right)\right\}}, \quad \forall i \in I,\ j \in J. \tag{23}$$

$$\max_{0\le\lambda\le 1} Z_L(\lambda) = \sum_{\lambda}\int_0^{v(\lambda/a)} t_a(x)\mathrm{d}x + \frac{1}{\theta}\sum_{i\in I}\sum_{j\in J} q_{ij}^{\lambda}\left(\ln q_{ij}^{\lambda} - 1\right) + \sum_{j\in J} q_j^{\lambda}\cdot\pi_j + \sum_{j\in J}\int_0^{q(\lambda/j)} S_j(y)\mathrm{d}y, \tag{24}$$

$$\begin{aligned} v_a^{n+1} &:= v_a^{\lambda^*}, \quad \forall a \in A, \\ q_{ij}^{n+1} &:= q_{ij}^{\lambda^*}, \quad \forall i \in I,\ j \in J. \end{aligned} \tag{25}$$

$$F_r^{ij} = \gamma_{ijr}\left(e_{ij} + q_{ij}^{n+1}\right), \quad \forall r \in R_{ij},\ i \in I,\ j \in J. \tag{26}$$

(3) For each destination $j \in J$ associated with origin i, we assign both the fixed demand e_{ij} and the variable demand q_{ij}^0 to the minimum cost path $\bar{r}$ from i to j and, thereby, initialize the path flow $\{F_r^{ij}\}$ and path flow proportions $\{\gamma_{ijr}^0 = 1\}$, as well as the link flows $\{v_a^0\}$.

(4) update the link costs.

Main Loop. For $n = 1$ to number of main iterations (L_{Main}):

Step 1: we equilibrate path flows, retaining the variable demands, given the current variable O-D demand $\{q_{ij}^n\}$ and path flows $\{F_r^{ij}\}$, obtained from

the $(n\text{-}1)^{\text{th}}$-iteration. For $m=1$ to number of inner iterations (L_{Inner}):

For each origin $i \in I$, we generate the minimum path tree rooted from i. For each destination $j \in J$ associated with origin i, we retrieve the minimum path $\bar{r}$ with the minimum cost τ_{ij} and equilibrate the path set R_{ij} between the O-D pair (i, j) as follows:

Step 1.1: column generation: updating the costs of paths, c_{ijr}, $\forall r \in R_{ij}$.

If $\bar{r} \notin R_{ij}$, let $R_{ij}: = R_{ij} \cup \{\bar{r}\}$; otherwise, label the minimum path in R_{ij} as $\bar{r}$.

Step 1.2: shift flow from nonminimum paths to the minimum path:

(1) The path flow shift is computed as follows:

where $A(r, \bar{r})$ is the set of links that belong either to nonminimum path r or to minimum path $\bar{r}$ but not to both of them.

(2) The path flows are updated and projected onto the feasible set:

Nonminimum paths: $F_r^{ij}: = F_r^{ij} - \min\left\{\Delta F_r, F_r^{ij}\right\}$, $\forall r \in R_{ij}$, $r \neq \bar{r}$

Minimum path: $F_{\bar{r}}^{ij}: = (q_{ij} + e_{ij}) - \sum_{r \in R_{ij}, r \neq \bar{r}} F_r^{ij}$

If $F_r^{ij} = 0$, then eliminate this path r from the path set, that is, $R_{ij}: = R_{ij}/r$.

(3) The link flows $\{v_a\}$, the link costs $\{t_a\}$, and the derivative costs $\{\{\partial t_a/\partial v_a\}\}$ are updated.

Step 1.3: updating the path flow proportions:

(2) The auxiliary link flows $\{\hat{v}_a^n\}$ are obtained with the given path flow proportions $\left\{\gamma_{ijr}^n\right\}$.

(3) Let $v_a^\lambda = (1-\lambda)v_a^n + \lambda\hat{v}_a^n$, $\forall a \in A$; $q_{ij}^\lambda = (1-\lambda)q_{ij}^n + \lambda\hat{q}_{ij}^n$, $\forall i \in I$, $j \in J$, and we solve the one-dimensional search problem defined as follows to obtain the optimal step size $\lambda^* \in [0, 1]$:

where $q_j^\lambda = \rho_j D_j^\lambda = \sum_{j \in J}(e_{ij} + q_{ij}^\lambda)$.

(4) The link flows and the variable O-D demands are updated as follows:

Step 2.3: updating the path flows:

Step 2.4: the link costs $\{t_a\}$ and the derivative costs $\{\partial t_a/\partial v_a\}$ are updated.

Step 3: checking convergence: it is terminated if the convergence criterion is satisfied; otherwise, a new iteration of the main loop is started, and set $n: = n + 1$.

3.2. Sensitivity Analysis of the CTDTA Model. In every iteration, the SAB algorithm evaluates the derivatives of the lower-level decision variables with respect to the upper-level ones (i.e., $\nabla_{\tilde{O}}\mathbf{v}$ and $\nabla_{\tilde{O}}\mathbf{q}$) based on the current solution to the lower-level model. The derivatives will be used to approximate the bilevel model as a linear programming problem which is easier to tackle. For the original CTDTA model, the approach for deriving derivatives can follow the restriction sensitivity analysis approach for the combined model in Du et al. [30]. The restriction approach reduces the original problem involving all path variables into a restricted one that only consists of those independent equilibrated paths, so the problem of nonuniqueness of the path flows can be handled. Also, using the sensitivity results of the independent paths in the restricted problem, the derivatives of the link flow in the original one will be restored. The necessary results of the analytical sensitivity analysis of the extended CTDTA model are presented in the following.

To simplify the calculation of the sensitivity analysis approach, the extended CTDTA model in equations (7)–(14) will be reformulated as follows (by including the perturbation ε):

$$\min Z_L'(\mathbf{f}, \mathbf{h}, \boldsymbol{\varepsilon}) = \sum_{a \in A}\int_0^{v_a} t_a(x, \varepsilon)\mathrm{d}x + \frac{1}{\theta}\sum_{i \in I}\sum_{j \in J}\sum_{r \in R_{ij}} f_r^{ij}$$
$$\left(\ln \sum_{r \in R_{ij}} f_r^{ij} - 1\right) + \sum_{j \in J} q_j \pi_j + \eta \sum_{j \in J}\int_0^{q_j} S_j(y, \varepsilon)\mathrm{d}y, \quad (27)$$

$$s.t. \sum_{j \in J}\sum_{r \in R_{ij}} f_r^{ij} = O_i(\varepsilon), \quad \forall i \in I, \quad (28)$$

$$\sum_{r \in R_{ij}} h_r^{ij} = e_{ij}(\varepsilon), \quad \forall i \in I, j \in J, \quad (29)$$

$$v_a = \sum_{i \in I}\sum_{j \in J}\sum_{r \in R_{ij}}\left(f_r^{ij} + h_r^{ij}\right)\delta_{a,r}^{ij}, \quad \forall a \in A, \quad (30)$$

$$f_r^{ij} \geq 0, \quad \forall r \in R_{ij}, i \in I, j \in J, \quad (31)$$

$$h_r^{ij} \geq 0, \quad \forall r \in R_{ij}, i \in I, j \in J. \quad (32)$$

In the equations given above, the decision variables are reduced to the path flows $\mathbf{f}$ and $\mathbf{h}$ only. After that, by substituting equation (30) into the upper limit of the integral in the first term of the objective function, we can obtain the following Lagrange function of the abovementioned problem.

$$\begin{aligned} L(\mathbf{f}, \mathbf{h}, \boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\varepsilon}) = {} & \sum_{a \in A}\int_0^{\sum_{i \in I}\sum_{j \in J}\sum_{r \in R_{ij}}\left(f_r^{ij} + h_r^{ij}\right)\delta_{a,r}^{ij}} t_a(x, \varepsilon)dx \\ & + \frac{1}{\theta}\sum_{i \in I}\sum_{j \in J}\sum_{r \in R_{ij}} f_r^{ij}\left(\ln \sum_{r \in R_{ij}} f_r^{ij} - 1\right) \\ & + \sum_{j \in J} q_j \pi_j + \eta \sum_{j \in J}\int_0^{q_j} S_j(y, \varepsilon)dy \\ & + \lambda_i\left(O_i(\varepsilon) - \sum_{j \in J}\sum_{r \in R_{ij}} f_r^{ij}\right) \\ & + \mu_{ij}\left(e_{ij}(\varepsilon) - \sum_{r \in R_{ij}} h_r^{ij}\right), \end{aligned} \quad (33)$$

where $\lambda = \{\lambda_i\}$ and $\mu = \{\mu_{ij}\}$ are the Lagrange multipliers associated with the conservation constraints (28) and (29). Then, the Kuhn–Tucker condition can be conducted as follows (set $\varepsilon = 0$):

$$\begin{aligned}
\mathbf{c}(\mathbf{f}^*, \mathbf{h}^*, 0) + \frac{1}{\theta}\Lambda_f^T \ln\left(\Lambda_f \mathbf{f}^*\right) + \Lambda_f^T \Psi^T (\boldsymbol{\rho} + \eta \mathbf{S}(\cdot, 0)) - \Lambda_f^T \Phi^T \boldsymbol{\lambda} &= 0, \\
\mathbf{c}(\mathbf{f}^*, \mathbf{h}^*, 0) + \Lambda_h^T \Psi^T (\boldsymbol{\rho} + \eta \mathbf{S}(\cdot, 0)) - \Lambda_h^T \boldsymbol{\mu} &= 0, \\
\Phi \Lambda_f \mathbf{f}^* - \mathbf{O}(0) &= 0, \\
\Lambda_h \mathbf{h}^* - \mathbf{e}(0) &= 0, \\
\mathbf{u}_f^T \mathbf{f}^* &= 0, \\
\mathbf{u}_h^T \mathbf{h}^* &= 0, \\
\mathbf{u}_f \geq 0,\ \mathbf{u}_h \geq 0, \mathbf{f}^* \geq 0, \mathbf{h}^* &\geq 0.
\end{aligned} \tag{34}$$

By only retaining the used paths whose flows are positive, the abovementioned equation system will be reduced to

$$\begin{aligned}
\mathbf{c}(\mathbf{f}^*, \mathbf{h}^*, 0) + \frac{1}{\theta}\Lambda_f^T \ln\left(\Lambda_f \mathbf{f}^{0*}\right) + \Lambda_f^T \Psi^T (\boldsymbol{\rho} + \eta \mathbf{S}(\cdot, 0)) - \Lambda_f^T \Phi^T \boldsymbol{\lambda} &= 0, \\
\mathbf{c}(\mathbf{f}^*, \mathbf{h}^*, 0) + \Lambda_h^T \Psi^T (\boldsymbol{\rho} + \eta \mathbf{S}(\cdot, 0)) - \Lambda_h^T \boldsymbol{\mu} &= 0, \\
\Phi \Lambda_f \mathbf{f}^{0*} - \mathbf{O}(0) &= 0, \\
\Lambda_h \mathbf{h}^{0*} - \mathbf{e}(0) &= 0.
\end{aligned} \tag{35}$$

Furthermore, let $[\mathbf{f}^{0*}, \mathbf{h}^{0*}]$ be partitioned as $[(\mathbf{f}^B, \mathbf{f}^{\mathrm{NB}})^T, (\mathbf{h}^B, \mathbf{h}^{\mathrm{NB}})^T]$. $\mathbf{f}^{\mathrm{B}}$ and $\mathbf{h}^{\mathrm{B}}$ are the basic variables of path flows associated with the variable and fixed demand, respectively; $\mathbf{f}^{\mathrm{NB}}$ and $\mathbf{h}^{\mathrm{NB}}$ are the corresponding nonbasic variables. Note that $\mathbf{f}^{\mathrm{B}}$ and $\mathbf{h}^{\mathrm{B}}$ are the flows on the independent equilibrated paths according to the restriction sensitivity analysis approach. Then, we can restrict the equation system by holding the nonbasic variable to be $[\mathbf{f}^{\mathrm{NB}}, \mathbf{h}^{\mathrm{NB}}] = 0$. Thereby, the Jacobian matrix $\mathbf{J}_{\mathbf{x}}$ with respect to the solution variables $\mathbf{x} = (\mathbf{f}^B, \mathbf{h}^B, \lambda, \mu)$, and the Jacobian matrix $\mathbf{J}_{\tilde{O}}$ with respect to the production of the variable demand from origins can be derived as

$$\begin{aligned}
\mathbf{J}_{\mathbf{x}} &= \begin{bmatrix} \nabla_f^2 L & \nabla_{f,h} L & -\Lambda_f^T \Phi^T \boldsymbol{\lambda} & 0 \\ \nabla_{h,f} L & \nabla_h^2 L & 0 & -\Lambda_h^T \\ \Phi \Lambda_f & 0 & 0 & 0 \\ 0 & \Lambda_h & 0 & 0 \end{bmatrix}, \\
\mathbf{J}_{\tilde{O}} &= \begin{bmatrix} 0 & 0 & -I & 0 \end{bmatrix}^T,
\end{aligned} \tag{36}$$

where

$$\begin{aligned}
\nabla_f^2 L &= \Delta_f^T \nabla_{\mathbf{v}} \mathbf{t}(\mathbf{v}, 0) \Delta_f + \left(\frac{1}{0}\right) \Lambda_f^T \mathrm{diag}\left(\frac{1}{f_r^{ij}}\right) \Lambda_f + \Lambda_f^T \Psi^T \left(\eta \nabla_q S(\cdot)\right) \Psi \Lambda_f, \\
\nabla_{h,f} L &= \Delta_h^T \nabla_{\mathbf{v}} \mathbf{t}(\mathbf{v}, \varepsilon) \Delta_f + \Lambda_h^T \Psi^T \left(\eta \nabla_q S(\cdot)\right) \Psi \Lambda_f, \\
\nabla_h^2 L &= \Delta_h^T \nabla_{\mathbf{v}} \mathbf{t}(\mathbf{v}, \varepsilon) \Delta_h + \Lambda_h^T \Psi^T \left(\eta \nabla_q S(\cdot)\right) \Psi \Lambda_h, \\
\nabla_{f,h} L &= \Delta_f^T \nabla_{\mathbf{v}} \mathbf{t}(\mathbf{v}, \varepsilon) \Delta_h + \Lambda_f^T \Psi^T \left(\eta \nabla_q S(\cdot)\right) \Psi \Lambda_h.
\end{aligned} \tag{37}$$

Note that diag $(1/f_r^{ij})$ is a diagonal matrix with $1/f_r^{ij}$ as the diagonal element. Here, the perturbation ε corresponds to the original production $\widetilde{O}$ in this problem. Δ_f and Δ_h are the link/path incidence matrix corresponding to independent equilibrated path flow $\mathbf{f}^{\mathrm{B}}$ and $\mathbf{h}^{\mathrm{B}}$, respectively. Λ_f and Λ_h are the O-D/path incidence matrix corresponding to $\mathbf{f}^{\mathrm{B}}$ and $\mathbf{h}^{\mathrm{B}}$, separately. Φ is the origin/O-D incidence matrix. Ψ is the destination/O-D incidence matrix. The superscript 'T' represents the transposed matrix. 'I' denotes the identity matrix. The link cost function, $\mathbf{t}(\cdot)$, is strongly monotonically defined in $\mathbf{v}$, which guarantees the uniqueness of the equilibrium solution.

Therefore, in the restricted problem, the derivatives of the model solutions with respect to the input parameters are produced by the following equation:

$$\nabla_{\widetilde{O}}\mathbf{x} = \left[\mathbf{J}_{\mathbf{x}}\right]^{-1}\left[-\mathbf{J}_{\widetilde{O}}\right], \tag{38}$$

where $\mathbf{J}_{\mathbf{x}}$ is an invertible Jacobian matrix. According to the restriction approach, the derivative, $\nabla_{\widetilde{O}}\mathbf{x}(\varepsilon)$, is first conducted in the restricted problem, and then, $\nabla_{\widetilde{O}}\mathbf{v}$, which is defined in the original problem, will be obtained by

$$\nabla_{\widetilde{O}}\mathbf{v} = \Delta_f \nabla_{\widetilde{O}}\mathbf{f}^B + \Delta_f \nabla_{\widetilde{O}}\mathbf{h}^B. \tag{39}$$

Equation (39) indicates that the variations on the link flows can be just represented by the changes in the basic path flows, which is the rationale of the restriction sensitivity analysis approach. In addition, $\nabla_{\widetilde{O}}\mathbf{q}$ is also obtained by

$$\nabla_{\widetilde{O}}\mathbf{q} = \Lambda_f \nabla_{\widetilde{O}}\mathbf{f}^B + \Lambda_f \nabla_{\widetilde{O}}\mathbf{h}^B. \tag{40}$$

In addition, the inverse of matrix $\mathbf{J}_{\mathbf{x}}$ in equation (38) is required to be worked out. To avoid the disadvantages of deriving the whole inverse matrix directly, it is usually to solve the corresponding linear equations as an alternative. Hence, equation (38) is rewritten as

$$\mathbf{J}_{\mathbf{x}} \cdot \nabla_{\widetilde{O}}\mathbf{x} = -\mathbf{J}_{\widetilde{O}}. \tag{41}$$

The abovementioned system of linear equations can be solved efficiently by using the exiting solvers, such as the Cplex or Matlab built-in functions. Besides, the well-known block inversion formula can be used to incorporate with iterative solution techniques for the system of linear equations [31], ensuring that no matrix needs to be inverted with large dimension.

3.3. An Adaptive Step-Size Method for SAB Algorithm. The analytical sensitivity expressions derived in the previous section are essential to uncover the implicit relationships between lower-level decision variables, ($\mathbf{v}$, $\mathbf{q}$), and upper-level variables, $\widetilde{O}$. With the derivatives, the bilevel model is approximated as a single-level model using first-order Taylor's expansion at a solution point, i.e., $\widetilde{O}^* = f^{-1}(\mathbf{v}^*, \mathbf{q}^*)$. Let $\mathbf{v}(\widetilde{O}^*)$ and $\mathbf{q}(\widetilde{O}^*)$ denote the solutions to the lower-level model at $\widetilde{O}^*$, respectively. The implicit relationships can be represented by using the Taylor expansion:

$$\begin{aligned} \mathbf{v}(\widetilde{O}) &\approx \mathbf{v}\left(\widetilde{O}^*\right) + \nabla_{\widetilde{O}}\mathbf{v} \cdot \left(\widetilde{O} - t\widetilde{O}^*\right), \\ \mathbf{q}(\widetilde{O}) &\approx \mathbf{q}\left(\widetilde{O}^*\right) + \nabla_{\widetilde{O}}\mathbf{q} \cdot \left(\widetilde{O} - t\widetilde{O}^*\right), \end{aligned} \tag{42}$$

where the derivatives $\nabla_{\widetilde{O}}\mathbf{v}$ and $\nabla_{\widetilde{O}}\mathbf{q}$ are obtained from the sensitivity analysis of the CTDTA model. Therefore, the original bilevel problem will be approximated as

$$\begin{aligned} &\text{Max} \quad \sum_{j\in J} \widetilde{O}_j \\ &s.t. \quad \begin{bmatrix} \nabla_{\widetilde{O}}\mathbf{v} \\ \rho\Psi \cdot \nabla_{\widetilde{O}}\mathbf{q} \end{bmatrix} \cdot \widetilde{O} \le \begin{bmatrix} C \\ \mathbf{k} - \rho\Psi \cdot \mathbf{e} \end{bmatrix} - \begin{bmatrix} \mathbf{v}^* \\ \rho\Psi \cdot \mathbf{q}^* \end{bmatrix} + \begin{bmatrix} \nabla_{\widetilde{O}}\mathbf{v} \\ \rho\Psi \cdot \nabla_{\widetilde{O}}\mathbf{q} \end{bmatrix} \cdot \widetilde{O}^* \\ &\quad\quad \widetilde{O} \ge 0. \end{aligned} \tag{43}$$

Let $\widehat{O}^n$ denote the solution of the abovementioned linear programming problem. Since the auxiliary solution $\widehat{O}^n$ does not always satisfy the feasibility of the original problem, so the solution is updated using the convex combination of the solution in the last iteration, $\widetilde{O}^n$ (i.e., the start point in this iteration), and the auxiliary solution, $\widehat{O}^n$, which is given by

$$\widetilde{O}^{n+1} := \widetilde{O}^n + \min\left\{\frac{\zeta_{\max}}{\left|\widehat{O}^n - \widetilde{O}^n\right|}, 1\right\} \cdot \left(\widehat{O}^n - \widetilde{O}^n\right), \tag{44}$$

where $\zeta_{\max} \in (0, 1]$ is a predetermined maximum step size that prevents the updated solution from moving too long. $(\widehat{O}^n - \widetilde{O}^n)$ is the descent direction. The value of $\zeta_{\max}$ could be flexible. An adaptive scheme for $\zeta_{\max}$ will be suggested in the large-scale problem for providing a step size sequence that ensures the convergence.

4. Numerical Experiments

Numerical experiments are conducted in this section. In the first example, we employ a small network, composed of 7 links and 6 nodes to demonstrate the proposed network capacity model and, then, analyze the effect of parking pricing and parking supply on the network capacity of the road system. The example network is shown in Figure 1. There are 4 O-D pairs from the origin nodes 1 and 2 to the

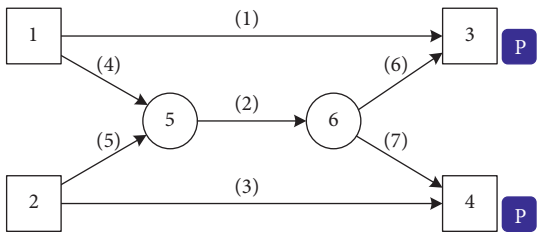


Figure 1: An example network.

destination nodes 3 and 4. The BPR function is used as the link performance function, given by $t_a(v_a) = t_a^0[1+0.15(v_a/C_a)^4]$. The characteristics of all links in the network are listed in Table 1. Table 2 gives the fixed demand in the network. The parameters related parking strategies at destinations are listed in Table 3. Besides, the value of time η is set to 1.0, and the dispersion parameter θ for destination choice is set to 0.5.

4.1. Results of the New Model. To show the advantages of the proposed model, the result of the proposed model is compared with the other network capacity models in the literature [3, 4, 19, 24]. The experiments are carried out on the example in Figure 1. As the models in the literature on this topic were conducted for different problems under various backgrounds, it is inappropriate to compare the results of these models directly. Therefore, we conclude three situations from the literature: (1) without variable O-D demand (i.e., the reserve capacity model in this paper); (2) without variable destination cost; (3) without fixed O-D demand.

For the situation without the variable O-D demand, the reserve network capacity model [1] is modified by adding the parking supply constraint. The O-D flows are assumed to grow following a fixed O-D distribution, and the effect of traffic congestion and parking pricing at destination areas is not involved. The reserve capacity is the most classical concept that has been widely applied for the assessment of transportation network capacity. Such concept was used in Asakura and Kashiwadani [19] and Leng and Yan [4]. The results of the reserve capacity model and the proposed model are reported in Tables 4 and 5, respectively.

From the results in Tables 4 and 5, the network capacity is to be 137.5 by the reserve capacity model, while it is evaluated to be 222.35 by the proposed model. In both models, flows on link 5 and link 6 reach the capacity, and the links are the bottlenecks limiting the growth of the total travel demand in the network. Besides, in the results of the reserve capacity model, no parking facility is saturated; however, in the results of the proposed model, the parking capacity at destination 3 is fully used. The network capacity from the reserve capacity model is evidently lower than that from the proposed model. It means the reserve capacity tends to produce too conservative results [8], since it assumes an unchangeable travel demand pattern. The network-wide travel demand will be restricted to continue to grow if the flow on any individual link meets its capacity. By contrast, the proposed model with the CTDTA model as the lower level produces a changeable O-D demand pattern, which is more practical. Thus, a portion of travel demand can adjust their travel destination according to the parking cost and available space at the destination area. In this example, the limits to the total demand come from the capacities on link 5 and link 6, the parking capacity at destination 3, and the restraint from the O-D travel cost (soft constraint). Note that the variable travel demand does not change the O-D distribution in a completely free manner. The O-D travel distribution should follow the logit share model in equation (15), by which the destinations with lower disutility will be assigned more travel demand for any origin area, and vice versa.

Table 1: Link characteristics of the example network in Figure 1.

Links	1	2	3	4	5	6	7
t_a^0	10	5	12	4	4	5	4
C_a	100	120	80	80	50	50	50

Table 2: Fixed travel demand in the network.

3	4	$\overline{O}_i$	
1	30	20	50
2	40	20	60
$\overline{D}_j$	70	40	Total demand = 110

Table 3: The parameters of parking strategy at destinations.

Destinations	Low-occupancy searching time s_j	γ	ω	Parking pricing π_j	Parking capacity k_j	Parking rate ρ_j
3	2.0	1.0	2.0	4.0	100	0.75
4	3.0	1.0	2.0	5.0	80	0.75

For the situation without variable destination cost, the O-D flows are assumed to grow with a variable O-D distribution, while the effect of traffic congestion at destinations is not considered. Tam and Lam [3] employed this behavior pattern for the maximum car ownership problem. For the situation without the fixed O-D demand, all motorists are considered to change their trip destination under the influence of traffic congestion (both from O-D travel cost and destination cost) and parking management strategy. Tam and Lam [3] and Xie et al. [24] used this assumption.

Figure 2 further compares the maximum demand patterns of different network capacity models. The above-mentioned three situations are labeled as "Reserve Capacity," "Without Destination Cost," and "Without Fixed Demand." The O-D demand patterns of the models are significantly different. Specifically, the differences are mainly on the demands between O-D (1–3) and O-D (2–4). The maximum demand pattern from the "Reserve Capacity" model is rather conservative, as it clearly follows the fixed O-D demand (as a prescribed O-D demand) existed in the network. Although O-D (1–3) and O-D (2–4) can be allocated more travel demand, limited by the capacity on link 5 and link 6 (the vital links for O-D (2–3)), the total demand

Table 4: Results of the reserve network capacity model.

O-D	Demand types		Destination demand			Links	Link flow	V/C ratio
	Fixed demand	Additional demand	Destination	Total trip attraction	Parking demand			
1–3	30	7.50	3	87.50	65.6	1	37.50	0.375
1–4	20	5.00	4	50.00	37.5	2	75.00	0.625
2–3	40	10.00				3	25.00	0.312
2–4	20	5.00				4	25.00	0.312
Total	110	27.50				5	50.00	1.000
						6	50.00	1.000
						7	25.00	0.500
	Network capacity			137.50 (demand multiplier = 1.25)				

Table 5: Results of the proposed road network capacity model.

O-D	Demand types		Destination demand			Links	Link flow	V/C ratio
	Fixed demand	Additional demand	Destination	Total trip attraction	Parking demand			
1–3	30	53.33	3	133.33	100.0	1	83.33	0.833
1–4	20	6.91	4	65.13	48.85	2	26.91	0.336
2–3	40	10.00				3	38.21	0.478
2–4	20	18.21				4	50.00	1.000
Total	110	88.46				5	76.91	0.641
						6	50.00	1.000
						7	26.91	0.538
	Network capacity				198.46			

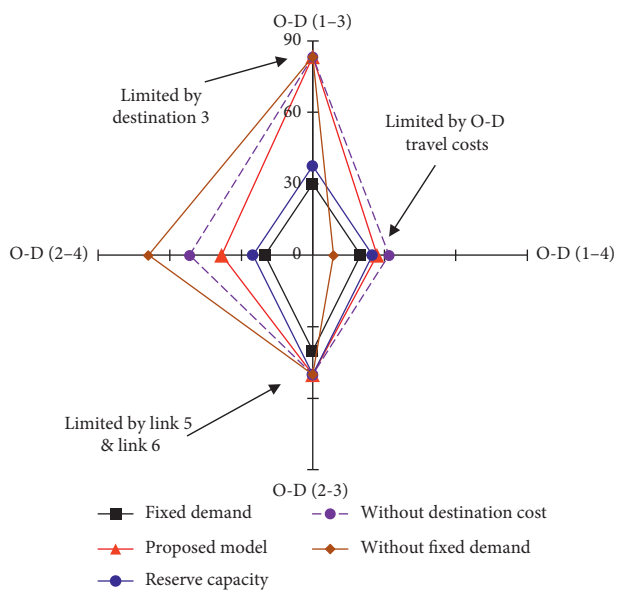


Figure 2: O-D demand pattern of maximum flow from different network capacity models.

cannot increase further. On the other hand, without considering the effect of destination congestions (i.e., the parking search time increases as the destination demand), the network capacity is overestimated in the situation "Without Destination Cost". Note that the travel demands between O-D (1–4) and O-D (2–4) are higher than those from the results of the proposed model. This reflects that, without variable destination cost, the congestion effect at destination 4 cannot be captured. Travelers will choose destination 4 even though the congestion in this area is at a high level (i.e., the parking search time could be very long), which is unrealistic in real-life. Moreover, the situation "Without Fixed Demand" assumes an extreme demand pattern of maximum flow, in which the travel demand between O-D (2–4) is much higher while the demand of O-D (1–4) is lower compared with the other models. Such a demand pattern means the commuters between O-D (1–4) should be asked to give up their trips to increase the demand between the other O-D pairs, which is also not realistic for the urban transportation system.

In Figure 2, the proposed model produces a moderate result for network capacity assessment. Compared to the "Reserve Capacity" model, the potential of the network capacity by our model will be further developed between O-D (1–3) and O-D (2–4) after link 5 and link 6 reach their capacities. The total demand will keep growing until the parking capacity at destination 3 is violated. Besides, note that the demand between O-D (1–4) does not increase much. Because the travel cost between O-D (1–4) is significantly larger than that between O-D (1–3), most variable travel demands from origin 1 would like to choose node 3 as the destination for noncommuting activities. As the demand between O-D (1–3) cannot increase (limited by parking capacity), the demand for O-D (1–4) will not grow either. We find such restraint is caused by the travel cost difference among O-D pairs, which is referred to be the soft constraint as mentioned before. In conclusion, the proposed model can capture a more comprehensive situation compared with the models in the literature and, thus, is capable to better assess the road network capacity problem.

4.2. The Effect of considering Parking Supply and Parking Pricing. Based on the proposed network capacity model incorporated with parking strategies, we analyze the effect of parking supply and parking pricing on the network-wide capacity in this section. For this purpose, we configure 6 scenarios listed in Table 6. The base scenario is the same as the example in Section 4.1. Scenarios 1 to 3 are associated with the changes in parking supply strategies. Scenarios 4 to 6 are corresponding to the adjustment of the parking pricing strategies at the destination areas.

4.2.1. Effect of Parking Supply. With the same parking pricing strategy, the effect of the parking supply on the network-wide capacity is analyzed. For Scenario 1 to 3, the settings of the parking capacities and the results of the network capacity model are reported in Table 7.

In Scenario 1, only the parking capacity at destination 3 is expanded (from 100 to 1000). Compared with the base scenario, the total network capacity is increased from 198.46 to 221.87. One of the binding (or active) constraints changes from the destination 3 to link 1. This indicates that the total capacity in the base scenario is limited by the parking capacity at destination 3. Expanding the parking supply at such destination areas can effectively enhance the network-wide capacity.

In Scenario 2, only the parking capacity at destination 4 is expanded (from 80 to 1000). However, the total network capacity is not increased (it could appear as unobvious changes in practice). It is because that the parking supply at the destination constraint is still active. Expanding the capacities on the other destinations will have a small effect on the network-wide capacity.

In Scenario 3, both the parking capacities at destination 4 and destination 5 are increased to 1000. It results in the same total network capacity as Scenario 1. This means if the network-wide capacity should be enhanced, it only needs to deal with the destinations whose capacity constraints are binding to the maximum total demand. The investment on the nonbinding parking supplies will have a marginal contribution to the total network capacity. Note that all of the binding constraints change to link capacities if the parking capacities are enlarged sufficiently. There is no need to expand parking capacities too much. On the other hand, ignoring the parking supply capacity might overestimate the capacity of the transportation network.

Table 6: Descriptions of scenarios.

Scenarios	Description
0	Base scenario
1	Only expand parking capacity at destination 3
2	Only expand parking capacity at destination 4
3	Expand parking capacity at both destinations
4	Increase parking pricing at one destination only
5	Increase parking pricing at both destinations in the same proportion
6	Increase parking pricing at both destinations by the same amount

4.2.2. Effect of Parking Pricing. By holding the parking supply the same as the base scenario, the effect of the parking supply on the network-wide capacity can be investigated. The settings of parking capacities, as well as the model results, through Scenario 4 to 6 are reported in Table 8.

In Scenario 4, we only raise the parking pricing at destination 3 (from 4.0 to 8.0). However, the total network capacity is enhanced significantly (increasing to 240.0). Both destinations are saturated so the total travel demand cannot increase anymore. Specifically, the parking costs (consists of parking pricing and parking search time) for destination 3 and destination 4 are changed from 8.00 to 9.12 to 12.00 and 11.00, respectively. This results in the growth in the proportion of the travelers who choose node 4 as their destination. Hence, the demand to destination 4 can further grow, until the parking demand reaches the capacity.

In Scenario 4, we raise the parking pricing to 200% for both destinations. The total network capacity is decreased slightly. The change of the parking pricing leads to the redistribution of the O-D demand pattern. According to the parking cost, a large proportion of travelers (for variable demand only) tend to choose destination 3. Therefore, the demand for destination 4 will not reach that level as the base scenario due to the soft constraint on O-D (1–4) and O-D (2–4).

In Scenario 6, we increase the parking pricing with the same amount (by 4.0). Based on our model, the network capacity is unchanged despite the parking costs change a lot. In practice, one may regard such

Table 7: Results of network capacity under different parking capacities.

	Parking capacity (k_3, k_4)							
	Base scenario: (100, 80)		Scenario 1: (1000, 80)		Scenario 2: (100, 1000)		Scenario 3: (1000, 1000)	
Destinations	Parking demand	Total trip attraction	Parking demand	Total trip attraction	Parking demand	Total trip attraction	Parking demand	Total trip attraction
3	100.00	133.33	112.50	150.00	100.00	133.33	112.50	150.00
4	48.85	65.13	53.90	71.87	48.85	65.13	53.90	71.87
Network capacity	198.46		221.87		198.46		221.87	
Binding constraints*	Capacities on link 5 and link 6**							
	Parking capacity at destination 3		Capacity on link 1		Parking capacity at destination 3		Capacity on link 1	

*The capacity limits of the maximum network throughput. **The capacity constraints on link 5 and link 6 are active in all scenarios.

Table 8: Results of network capacity under different parking pricing strategies.

	Parking pricing (π_3, π_4)							
	Base scenario: (4.0, 5.0)		Scenario 4: <(8.0, 5.0)		Scenario 5: (8.0, 10.0)		Scenario 6: (8.0, 9.0)	
Destinations	Parking demand	Parking cost	Parking demand	Parking cost	Parking demand	Parking cost	Parking demand	Parking cost
3	100.00	8.00	100.00	12.00	100.00	12.00	100.00	12.00
4	48.85	9.12	80.00	11.00	42.98	13.87	48.85	13.12
Network capacity	198.46		240.00		190.65		198.46	

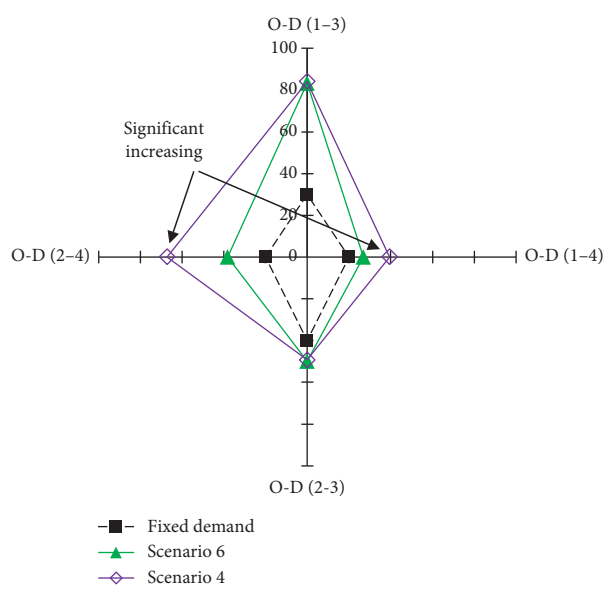


Figure 3: O-D demand pattern of maximum flow for Scenario 4 and Scenario 6.

adjustment of the parking pricing has little impact on the network-wide capacity.

In summary, adjusting the parking pricing strategy for selected destination areas will have an obvious influence on the network capacity of the transportation system. However, increase the parking pricing at all destinations together may result in a small effect for enhancing network capacity. Note that the O-D pattern (or spatial distribution) is the key to obtain a high network capacity. Figure 3 shows the maximal O-D demand pattern for Scenario 4 and Scenario 6. The travel demand increases significantly on O-D (1–4) and O-D (2–4). It indicates that Scenario 4

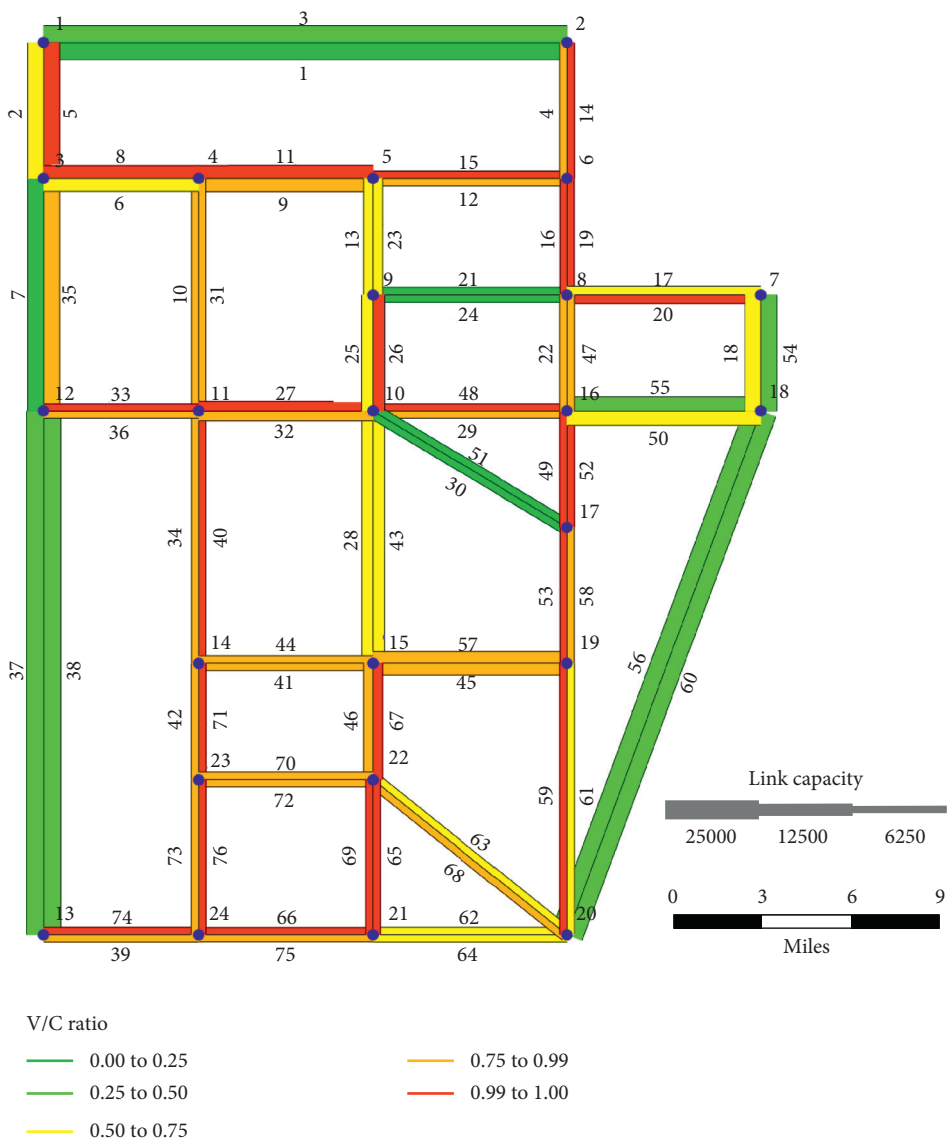


Figure 4: Link utilization pattern of maximum flow with an unlimited parking space.

results in a more proper O-D pattern to match the given network with higher utilization, and thus, a better effect is obtained on the network capacity compared with Scenario 6 (as well as Scenario 5).

4.3. Case Study in the Sioux-Falls Network. In this section, we numerically test the network capacity model with parking management strategies using the Sioux-Falls network (https://github.com/bstabler/TransportationNetworks). The Sioux-Falls network is a highway network with 24 nodes, 76 links, and 528 O-D pairs. All 24 nodes are origin/destination nodes. In order to test the network capacity model, we assumed the fixed travel demand does not exceed the network capacity. Therefore, the demand of all O-D pairs is rescaled as 1/5 of the original data.

We considered three different parking management strategies in the Sioux-Falls network: (1) unlimited parking space and unified parking pricing for all destinations; (2) limited parking space and unified parking pricing at the destinations; and (3) limited parking space and adjusted parking pricing at the destinations. For the unlimited parking space situation, all destination areas are assumed to have infinite parking capacity, so there is no parking capacity constraint. For the limited parking space situation, we considered each destination is assigned with limited parking capacity. The capacity constraints are chosen according to the fixed demand in the network. Specifically, destination 10 (with the most trip attraction of the fixed demand) is assigned with 20,000 unit parking capacity; destinations 11, 15, 16, 17, 20, and 22 are assigned with 12,000 unit parking capacity; and the capacity of the other destinations is set to 10,000. For the adjusted parking pricing situation, the parking fee at selected destinations (destinations 3, 4, 10, 12, 16, 17, 20, and 22) is raised from 5 to 7 (Dollars).

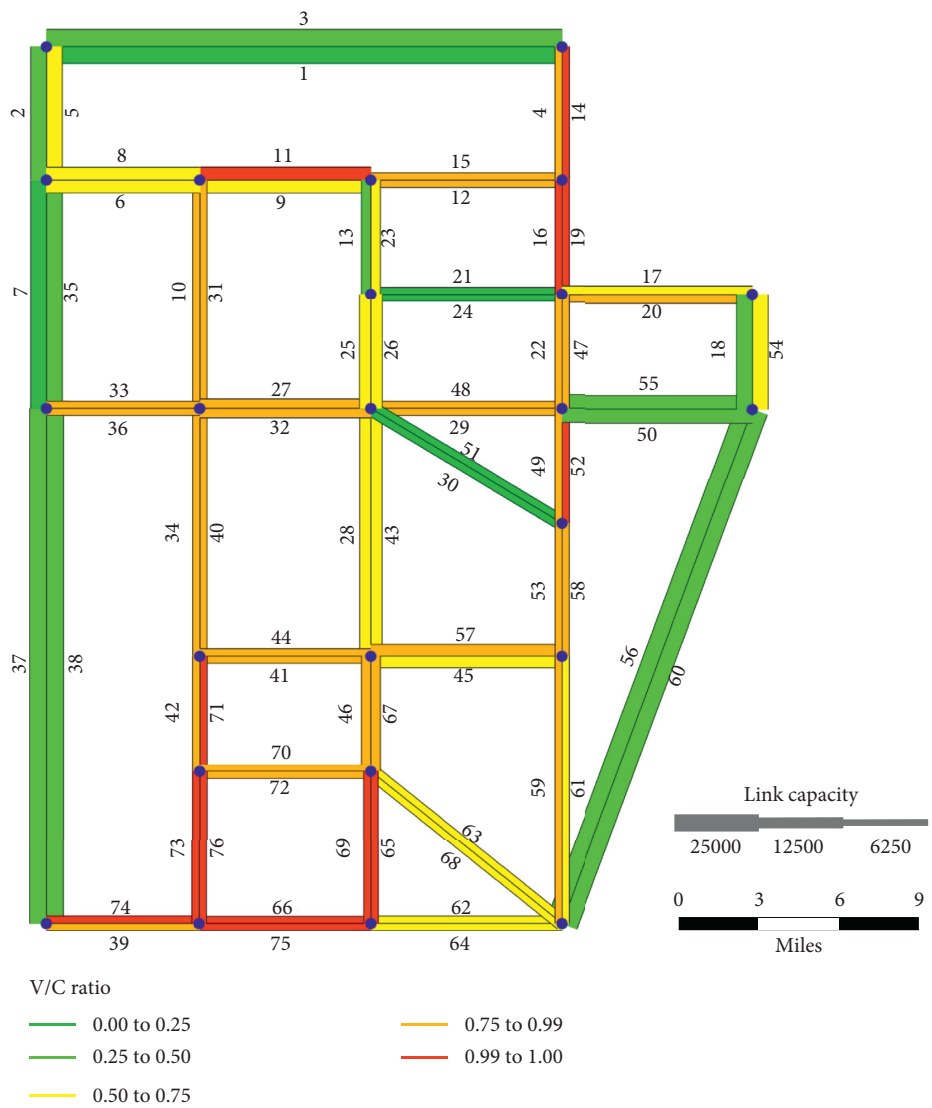


Figure 5: Link utilization pattern of maximum flow with a limited parking space.

The network capacities are assessed by using the proposed model associated with the abovementioned three parking management strategies. The link flow patterns of the maximum flow are shown in Figures 4–6, from which the link utilization patterns are illustrated. The width of a link represents its capacity. If a link is fully utilized (when link capacity is reached), it will be marked in red; otherwise, it will be in green. The summarized results of the network capacity assessment are presented in Table 9. Figure 7 illustrates the parking demand at each destination area corresponding to the three parking management strategies.

According to the numerical results in Table 9, the network capacity could be significantly overestimated when the capacity constraints of the parking supply are ignored. In the case of an unlimited parking space, as the parking capacity is not considered, the total travel demand is derived by only regarding the restraint of the link capacity. The trip attraction at some destinations may go unreasonably high. After involving the effect of the parking capacity, the network-wide capacity has been reduced much. As Figure 5 demonstrates, the number of saturated links (colored in red) decreases, compared to Figure 4, which indicates that the overall utilization of the network goes down. On the other hand, there are many links with low utilization (colored in green). These results can be regarded as the realistic capacity of this network, by considering the effect of the parking supply. Furthermore, if the traffic managers want to enhance the network-wide capacity with the limited road and parking supply resources, the adjusted parking pricing strategies will be recommended. By raising the parking fee at the selected

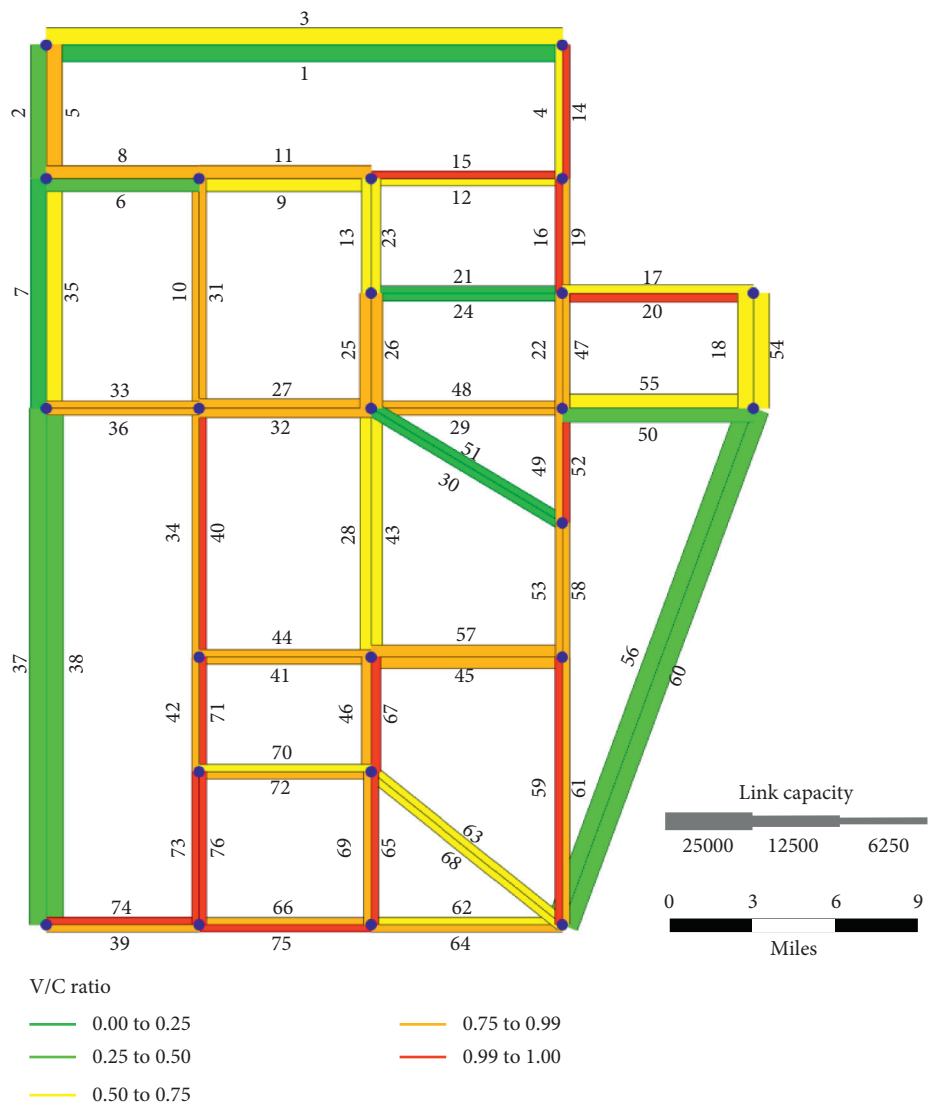


Figure 6: Link utilization pattern of maximum flow with limited parking space and adjusted parking pricing strategy.

Table 9: Summarized results under different parking management strategies in Sioux-Falls.

Parking management strategies	Network capacity	Binding capacity constraints	
		Link #	Destination #
Unlimited parking space and unified parking pricing	347,221.5 pcu/h	2, 4, 6, 9, 12, 16, 17, 19, 25, 29, 32, 34, 36, 39, 42, 46, 49, 52, 58, 61, 65, 69, 73, 75 (24 links)	None
Limited parking space and unified parking pricing	289,707.4 pcu/h	4, 9, 16, 19, 39, 42, 49, 65, 66, 69, 73, 75, 76 (13 links)	3, 4, 10, 12, 16, 17, 22 (7 destinations)
Limited parking space and adjusted parking pricing	306,230.4 pcu/h	4, 12, 17, 19, 34, 39, 42, 46, 49, 61, 66, 69, 73, 76 (14 links)	1, 2, 5, 8, 10, 15, 16 (7 destinations)

destinations whose parking spaces have been in a saturated situation, the network capacity will be increased further. This effect is consistent with the experimental results in section 4.2.1. Figure 6 shows that the network utilization will be increased compared to Figure 5, as the portion of the green links is reduced. Therefore, adjusting parking pricing

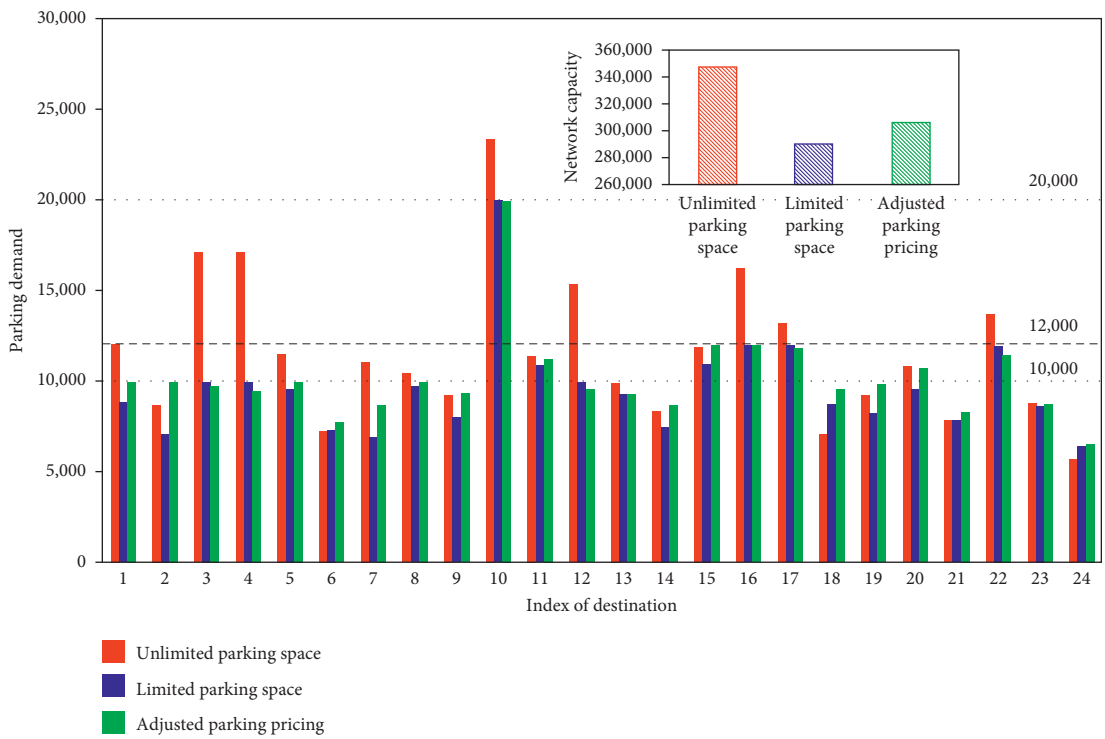


Figure 7: Maximum parking demand under different parking management strategies.

at the destination will be an effective strategy to promote the road network capacity.

5. Conclusions

This study proposed an extended road network capacity model based on the transportation network capacity model in [2]. In the extended model with a bilevel programming formulation, the constraint and impact of parking supply and parking pricing at each traffic zone (corresponding to the travel destination) are involved to enhance the assessment of the road network capacity for obtaining a more reasonable result. The parking supply is embodied as the capacity constraint in the upper level and is employed in the parking search time function for the CTDTA model in the lower level. The parking pricing is integrated into the objective function of the lower-level model. For solving the capacity model efficiently, we provided a sensitivity analysis based (SAB) solution algorithm, in which the lower-level CTDTA model is solved by a path-based solution algorithm. From the numerical examples, we showed that the proposed network capacity model is capable to capture the effect of parking strategies on the O-D travel demand pattern, which improves the reasonability of the network capacity assessment result. The experiments also demonstrated that the parking supply is significant to restrain the network-wide capacity. To expand the network-wide capacity, it is efficient to deal with the destinations whose parking capacity is binding to the maximum travel demand. Also, implementing different parking pricing adjustments at destination areas will be more effective by raising the parking pricing at all destinations with a unified standard. Future works will apply our proposed model to real case studies. Also, it will be of interest to find the optimal parking management scheme for expanding the capacity of the given transportation network to the maximum level.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was funded by the Natural Science Foundation of China (grant no. 71801079), Jiangsu University Philosophy and Social Science Research Project (grant no. 2019SJA0471), and the Fundamental Research Funds for the Central Universities (grant no. B200202079).

References

[1] S. C. Wong and H. Yang, "Reserve capacity of a signal-controlled road network," *Transportation Research Part B: Methodological*, vol. 31, no. 5, pp. 397–402, 1997.

[2] H. Yang, M. G. H. Bell, and Q. Meng, "Modeling the capacity and level of service of urban transportation networks," *Transportation Research Part B: Methodological*, vol. 34, no. 4, pp. 255–275, 2000.

[3] M. L. Tam and W. H. K. Lam, "Maximum car ownership under constraints of road capacity and parking space," *Transportation Research Part A: Policy and Practice*, vol. 34, no. 3, pp. 145–170, 2000.

[4] J. Leng and K. Yan, "Calculating model of road network capacity under limit of parking establishment capacity," *Journal of Tongji University*, vol. 35, no. 2, pp. 204–207, 2007.

[5] J. Wang, M. Du, L. Lu, and X. He, "Maximizing network throughput under stochastic user equilibrium with elastic demand," *Networks and Spatial Economics*, vol. 18, no. 1, pp. 115–143, 2018.

[6] A. Chen, H. Yang, H. K. Lo, and W. H. Tang, "Capacity reliability of a road network: an assessment methodology and numerical results," *Transportation Research Part B: Methodological*, vol. 36, no. 3, pp. 225–252, 2002.

[7] A. Sumalee, P. Luathep, W. H. K. Lam, and R. D. Connors, "Transport network capacity evaluation and design under demand uncertainty," *Transportation Research Record*, vol. 2090, pp. 93–101, 2009.

[8] A. Chen and P. Kasikitwiwat, "Modeling capacity flexibility of transportation networks," *Transportation Research Part A: Policy and Practice*, vol. 45, no. 2, pp. 105–117, 2011.

[9] M. Du, X. Jiang, and L. Cheng, "Alternative network robustness measure using system-wide transportation capacity for identifying critical links in road networks," *Advances in Mechanical Engineering*, vol. 9, no. 4, pp. 1–12, 2017.

[10] Y. Asakura, "Maximum capacity of road network constrained by user equilibrium conditions," in *Proceedings of the 24th Annual Conference of the UTSG*, University of Newcastle Upon Tyne, Newcastle Upon Tyne, UK, 1992.

[11] T. Akamatsu and O. Miyawaki, "Maximum network capacity problem under the transportation equilibrium assignment," *Infrastructure Planning Review*, vol. 12, pp. 719–729, 1995.

[12] H. Ceylan and M. G. H. Bell, "Reserve capacity for a road network under optimized fixed time traffic signal control," *Journal of Intelligent Transportation Systems*, vol. 8, no. 2, pp. 87–99, 2004.

[13] S.-W. Chiou, "Reserve capacity of signal-controlled road network," *Applied Mathematics and Computation*, vol. 190, no. 2, pp. 1602–1611, 2007.

[14] J. Wang, W. Deng, and J. Zhao, "Road network reserve capacity with stochastic user equilibrium," *Transport*, vol. 30, no. 1, pp. 103–116, 2015.

[15] Z. Y. Gao and Y. F. Song, "A reserve capacity model of optimal signal control with user-equilibrium route choice," *Transportation Research Part B-Methodological*, vol. 36, no. 4, pp. 313–323, 2002.

[16] L. Cheng, M. Du, X. Jiang, and H. Rakha, "Modeling and estimating the capacity of urban transportation network with rapid transit," *Transport*, vol. 29, no. 2, pp. 165–174, 2014.

[17] X. Xu, A. Chen, S. Jansuwan, C. Yang, and S. Ryu, "Transportation network redundancy: complementary measures and computational methods," *Transportation Research Part B: Methodological*, vol. 114, pp. 68–85, 2018.

[18] L. Shan, Z. Huang, and R. Luo, "Impact of market penetration of ATIS on road network capacity," in *Proceedings of the 2014 Sixth International Conference on Measuring Technology and Mechatronics Automation*, Zhangjiajie, China, January 2014.

[19] Y. Asakura and M. Kashiwadani, "Estimation model of maximum road network capacity with parking constraints and its application," *Infrastructure Planning Review*, vol. 11, pp. 129–136, 1993.

[20] R. García and A. Marín, "Parking capacity and pricing in park'n ride trips: a continuous equilibrium network design problem," *Annals of Operations Research*, vol. 116, no. 1–4, pp. 153–178, 2002.

[21] Z.-C. Li, H.-J. Huang, W. H. K. Lam, and S. C. Wong, "A model for evaluation of transport policies in multimodal networks with road and parking capacity constraints," *Journal of Mathematical Modelling and Algorithms*, vol. 6, no. 2, pp. 239–257, 2007.

[22] F. Leurent and H. Boujnah, "Traffic equilibrium in a network model of parking and route choice, with search circuits and cruising flows," *Procedia-Social and Behavioral Sciences*, vol. 54, pp. 808–821, 2012.

[23] M. Nourinejad and M. J. Roorda, "Impact of hourly parking pricing on travel demand," *Transportation Research Part A: Policy and Practice*, vol. 98, pp. 28–45, 2017.

[24] H. Xie, X. Yu, and K. Yan, "Evaluation model of urban road network system capacity," *China Journal of Highway and Transport*, vol. 25, no. 3, pp. 129–134, 2012.

[25] N. Oppenheim, "Equilibrium trip distribution/assignment with variable destination costs," *Transportation Research Part B: Methodological*, vol. 27, no. 3, pp. 207–217, 1993.

[26] K. W. Axhausen, J. W. Polak, M. Boltze, and J. Puzicha, "Effectiveness of the parking guidance information system in Frankfurt am Main," *Traffic Engineering & Control*, vol. 35, no. 5, pp. 304–309, 1994.

[27] T. L. Friesz, R. L. Tobin, H. J. Cho, and N. J. Mehta, "Sensitivity analysis based heuristic algorithms for mathematical programs with Variational Inequality constraints," *Mathematical Programming*, vol. 48, no. 2, pp. 265–284, 1990.

[28] R. Jayakrishnan, W. T. Tsai, J. N. Prashker, and S. Rajadhyaksha, *A Faster Path-Based Algorithm for Traffic Assignment*, University of California Transportation Center, Berkeley, CA, USA, 1994.

[29] S. P. Evans, "Derivation and analysis of some models for combining trip distribution and assignment," *Transportation Research*, vol. 10, no. 1, pp. 37–57, 1976.

[30] M. Du, L. Cheng, and H. Rakha, "Sensitivity Analysis of combined distribution-assignment model with applications," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2284, no. 1, pp. 10–20, 2012.

[31] S. D. Boyles, "Bush-based sensitivity analysis for approximating subnetwork diversion," *Transportation Research Part B: Methodological*, vol. 46, no. 1, pp. 139–155, 2012.

Hindawi
Journal of Advanced Transportation
Volume 2020, Article ID 8892859, 12 pages
https://doi.org/10.1155/2020/8892859

Research Article

Research on HOV Lane Priority Dynamic Control under Connected Vehicle Environment

Guiliang Zhou,[1,2] **Lina Mao**,[1,3] **Pengsen Hu**,[4] **Feng Sun**,[5] **and Xu Bao**[1]

[1]*Jiangsu Key Laboratory of Traffic and Transportation Security, Huaiyin Institute of Technology, Huaian 223003, China*
[2]*School of Automotive and Traffic Engineering, Jiangsu University, Zhenjiang 212013, China*
[3]*School of Transportation, Southeast University, Nanjing 210096, China*
[4]*Department of Civil and Environmental Engineering, The Pensylvania State University, University Park, PA 16802, USA*
[5]*School of Transportation and Vehicle Engineering, Shandong University of Technology, Zibo, Shandong 255049, China*

Correspondence should be addressed to Guiliang Zhou; zglpaper@qq.com and Lina Mao; 30721409@qq.com

Received 5 May 2020; Revised 16 July 2020; Accepted 23 July 2020; Published 8 August 2020

Academic Editor: Zhuping Zhou

Copyright © 2020 Guiliang Zhou 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.

The optimization of high-occupancy vehicle (HOV) lane management can better improve the efficiency of road resources. This paper first summarized the current research on HOV lane implementation and analyzed and identifies the threshold of setting road HOV lane dynamic control under the connected vehicle environment. Then, the HOV lane priority dynamic control process was determined, and the operating efficiency and energy consumption evaluation method was proposed. Moreover, a case study in Wuxi City, China, was carried out. The results showed that, after implementing the HOV lane priority dynamic control, the total mileage of road network vehicles was saved by 4.93%, the average travel time per capita was reduced by 4.27%, and the total energy-saving rate of road network travel was 21.96%.

1. Introduction

The setting of the HOV lane can provide more space resources for high-occupancy vehicles to improve traffic operation efficiency. However, it met with great controversy at the beginning [1–3] because when the HOV lane was not fully utilized at the time, road space resources were wasted [4]; Pravin and Han [5] made a comprehensive comparison between the HOV lane and the ordinary lane set in California in the United States. After investigating the HOV lane capacity, traffic operating condition, and carpooling percentage, it was believed that, under the appropriate road and traffic conditions, HOV lanes can serve to improve the traffic capacity, but when a certain saturation is reached, frequent lane changes may cause a decline in capacity. Dahlgren [6] claimed that the HOV lane setting has a significant effect on reducing carbon dioxide and other greenhouse gas emissions. Gutierrez et al. [7–10] made a detailed comparative analysis of the implementation effects of toll lanes and HOV lanes and concluded different applicable conditions for toll lanes and HOV lanes. Chen [11] discussed the experience and lessons of HOV priority planning and application worldwide in detail and proposed the necessity and feasibility of introducing HOV priority into urban transportation planning in China. The conclusion shows that the development level and urban residential characteristics of intelligent transportation technology in China have provided the necessary conditions for the implementation of HOV lanes.

Wang [12] put forward conceptual plans for carpooling priority based on the design experience of HOV schemes and pointed out that road, traffic, and urban land use functions are the key influencing factors of HOV lane setting in large cities. Bi [13] conducted an in-depth study on the carpooling rate model after the HOV lanes were set up and clarified the change law of the carpooling rate under the conditions of different passenger numbers for buses and HOV vehicles, thus proving the necessity and feasibility of setting HOV lanes in urban cities. Wu and Pan [14] analyzed and studied the relationship between the BRT lanes and

HOV lanes on urban roads and the implementation strategy; some scholars also studied aspects such as operating efficiency and traffic conditions. The feasibility of HOV lane setting has been systematically studied [15–21], which provides a theoretical basis for the scientific and rational setting of HOV lanes. In summary, most of the previous studies focused on the feasibility of HOV lane setting, road conditions, and HOV lane static management. However, the dynamic control of HOV lanes based on real-time data is ignored. With the rapid development and improvement of information technology and Internet of Vehicles (IoV) technology, it is feasible to realize HOV lane dynamic control using real-time data for carpooling priority.

2. Threshold Determination for Setting HOV Lanes Priority Dynamic Control under Connected Vehicle Environment

HOV lanes provide the right of way for high-occupancy vehicles. It improves the operating efficiency of highways and attracts more people to carpool, thus reducing the number of trips by private cars with low occupancy, and alleviates the urban traffic congestion. At present, the use of HOV lanes mostly adopts the fixed scheme. With the development of data acquisition methods, it will be feasible to implement dynamic intelligent control over the HOV lanes under connected vehicle environment, which will realize the optimal management of infrastructure resources.

In this section, in order to provide a basis for the dynamic control of HOV lanes with carpool priority, the threshold determination of introducing HOV lanes under connected vehicle environment was analyzed.

2.1. Speed-Volume Model. The BPR (Bureau of Public Roads) model shows the functional relationship between the travel time of road segments and the traffic load:

$$T(q) = T_0\left[1 + \alpha\left(\frac{q}{c}\right)^{\beta}\right], \tag{1}$$

where $T(q)$ refers to the travel time at flow q, T_0 is the travel time when traffic volume $q = 0$, C is the traffic capacity, and α and β are model parameters.

For the same traveler, the travel distance is a constant; therefore, equation (1) can be converted to the following equation:

$$V(q) = \frac{V_0}{\left[1 + \alpha(q/c)^{\beta}\right]}, \tag{2}$$

where $V(q)$ refers to the traffic speed and V_0 refers to the free-flow speed under the condition of mixed traffic of vehicles with different occupancy rates.

The free-flow speed can be determined by the road grade, and its recommended value is shown in Table 1.

In terms of the urban roads, V_0 is mainly related to intersection spacing, stop time of stations, signal period duration, and green signal ratio, which can be calculated by the following equation:

Table 1: Recommended value of free-flow speed for different grades of roads (km/h).

Road grade	Expressway	Arterial road	Collector road
Free-flow speed	60~80	50~60	40~50

$$V_0 = \frac{L}{\left(\left(L/V_f\right) + d\right)}, \tag{3}$$

where d refers to the average delay at intersections, L is the length of road segments, and V_f refers to the free-flow speed.

2.2. Speed-Volume Model Calibration. The least square method was used to calibrate the speed-volume model under the condition of mixed traffic and having HOV lane is as follows:

$$V(q) = \frac{31}{\left[1 + 1.03(q/c)^{3.01}\right]}. \tag{4}$$

2.3. Threshold for Introducing HOV Lane. For traffic travelers, the important indicator for evaluating travel quality is travel time, which heavily depends on the traffic condition. The traffic condition could be divided into two states: the ideal state and the actual state. Then, travel time is divided into ideal travel time and actual travel time as well. Actual travel time usually includes two parts: ideal travel time and ideal travel delay. The ideal travel time is only related to the travel distance and travel mode; that is, when the travel distance and mode are fixed, the ideal travel time is a fixed value, but the ideal travel delay will be determined by the traffic condition and will continue to increase as the traffic condition changes. The improvement and reduction of the traffic condition can effectively reflect the operating efficiency of the road network. This paper takes per capita delay as the evaluation index of road network operation efficiency. The threshold of introducing HOV lanes was investigated by minimizing per capita delay index.

The per capita delay index d' could be calculated by using the following equation:

$$d' = \frac{\left(q_b \cdot n_b \cdot d_b + q_c \cdot n_c \cdot d_C\right)}{\left(q_b \cdot n_b + q_c \cdot n_c\right)}, \tag{5}$$

where q_b refers to the traffic volume of high-occupancy vehicles, n_b represents the number of people in each occupancy vehicle, d_b refers to the delay for each high-occupancy vehicle, q_c refers to the traffic volume of nonhigh-occupancy vehicles, n_c represents the number of people in each nonoccupancy vehicle, and d_c refers to the delay for each nonoccupancy vehicle.

Per vehicle delay d can be calculated as follows:

$$d = \left(\frac{L}{V}\right)(q) - \left(\frac{L}{V_0}\right). \tag{6}$$

According to the previous survey and data analysis, about 2/3 of the people who own private cars are willing to take carpool, and about 1/3 of the people owning private cars show reluctance to carpooling, as shown in Figure 1.

It can be seen from Figure 2 that most (50.3%) high-occupancy vehicles have two passengers, followed by one passenger (24.56%) and three passengers (24.26%). Only 0.89% high-occupancy vehicles have four passengers.

Taking the bidirectional 6-lane road as an example, assume $q_b = (160veh/h)$ and $n_b = 3$, the threshold of setting HOV lanes priority dynamic control was determined as follows: firstly, substitute the traffic volumes into equations (3) and (4) to get the speed of high-occupancy and nonhigh-occupancy vehicles. Then, substitute the speeds into equation (6) to get the per vehicle delay for high-occupancy and nonhigh-occupancy vehicles. Finally, the per capita delay could be obtained by substituting the per vehicle delays into equation (5). Figure 3 shows the relationship between the per capita delay and the traffic volume before and after introducing HOV lane.

If we change the q_b from 160 to 180 vehicle per hour while other conditions remain the same, then the relationship between per capita delay and traffic volume is shown in Figure 4:

When the traffic volume of high-occupancy vehicles equals 160 vehicles per hour, it can be seen from Figure 3 that the normal mixed traffic has less per capita delay than the traffic with HOV lane when traffic volume is less than 600 vehicles per hour. It means that it is unsuitable to set a HOV lane. However, when the traffic volume is greater than 600, a HOV lane should be implemented to reduce the per capita delay. In this case, traffic volume of 600 vehicles per hour is the threshold for setting the HOV lane. Similarly, it can be seen from Figure 4 that when the volume of high-occupancy vehicles equals 180 vehicles per hour, the threshold is 640 vehicles per hour. The thresholds for different traffic volumes of high-occupancy vehicles are presented in Table 2.

2.4. HOV Lane Priority Dynamic Control Process with Carpooling under Connected Vehicle Environment. After determining the threshold of setting HOV lanes, this paper selected carpooling ratio, degree of saturation (v/c), and average traffic speed as the indictors for HOV lane priority dynamic control.

Taking a cycle at the downstream intersection of the road section as the time scale, the dynamic control flow chart is shown in Figure 5.

Step 1: using GPS, RFID, information transmission network, and other information collection technologies under the connected vehicle environment to collect the basic traffic information needed.

Step 2: performing data preprocessing, data cleaning, data repair, and screening on the data collected in step 1.

Step 3: uploading the processed data to the traffic information center.

Step 4: displaying the traffic information and theoretically analyzing the demand for setting HOV lanes based on the carpooling ratio, degree of saturation (v/c), and average traffic speed.

Step 5: setting the time scale as the signal cycle of the downstream intersection.

Step 6: determining the number of passengers on high-occupancy vehicles (carpooling).

Step 7: judging if the normal mixed traffic volume if greater than the threshold value. If yes, then HOV lanes only allow high-occupancy vehicles to access (setting HOV lanes). If no, HOV lanes open to all vehicles with different occupancy rates (removing HOV lanes).

Step 8: repeating step 4 to step 7.

2.5. Evaluation of Implementing HOV Lane Priority Dynamic Control. Real-time traffic information such as number of passengers, travel time, vehicle location, and vehicle type can be obtained through connected vehicle technologies. In this paper, the reduction in per capita travel time and total travel mileage is selected as the evaluation indicator of the implementation of the HOV lane priority dynamic control.

The main ideas are as follows: firstly, the real-time traffic operation condition of the road network in the mixed traffic state (before setting HOV lanes) is obtained: the number of passengers, travel time, and travel mileage. Secondly, based on the information above, the total travel mileage and travel time in the mixed traffic state can be calculated. Thirdly, theoretical analysis will be conducted to estimate the traffic operation condition assuming the HOV lanes were introduced and the total travel mileage and travel time of the new traffic state (after setting the HOV lanes) could be calculated as well. Finally, evaluate the effectiveness of implementing HOV lane priority dynamic control by vehicle mileage saving rate and reduction rate of per capita travel time consumption. The flow chart is shown in Figure 6.

2.5.1. Traffic Information Acquisition in Mixed Traffic. Assume there are m sections in the road network and n cars in each section, then the vehicle mileage of the road network in the most initial operating state (mixed traffic) can be expressed as follows:

$$S_0 = \sum_{j=0}^{m} \sum_{i=0}^{n} S_{ij0}, \tag{7}$$

where S_{ij0} is the mileage of the ith vehicle in the jth section of the road network and S_0 is the total mileage of the vehicles.

Then, the total travel time in the initial operating state could be calculated as follows:

$$PI_0 = P_{a0} * T_{a0} + P_{b0} * T_{b0} + P_{c0} * T_{c0}, \tag{8}$$

where PI_0 is the total travel time, P_{a0} is the number of noncarpooling travelers, T_{a0} is the average travel time of noncarpooling travelers, P_{b0} is the number of carpooling travelers, T_{b0} is the average travel time of carpooling travelers, P_{c0} is the number of public transportation travelers,

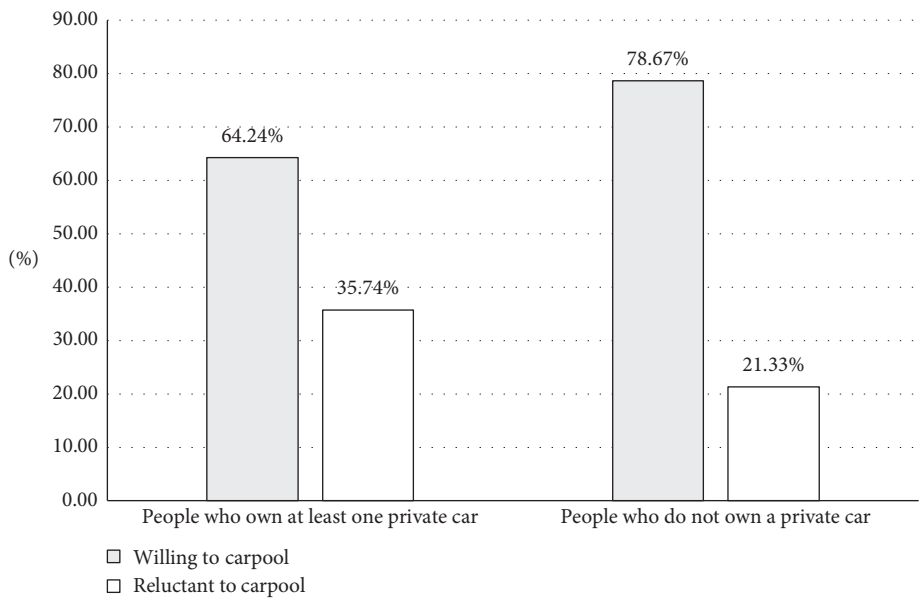


Figure 1: Bar graph of the carpooling willingness.

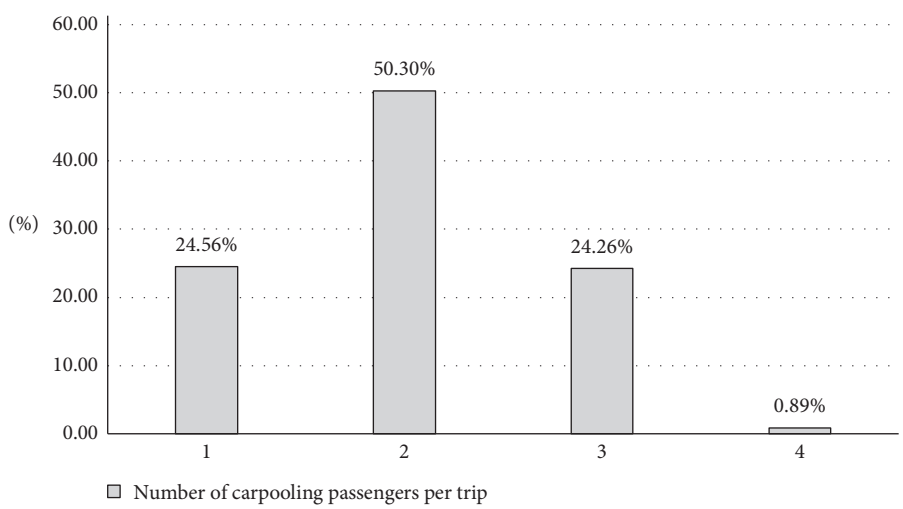


Figure 2: Bar graph of different carpooling passenger numbers.

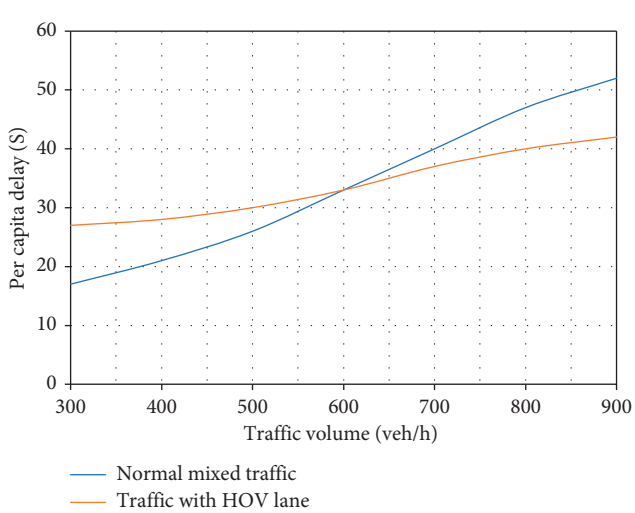


Figure 3: Per capita delay under different traffic conditions at $q_b = 160$ and $n_b = 3$.

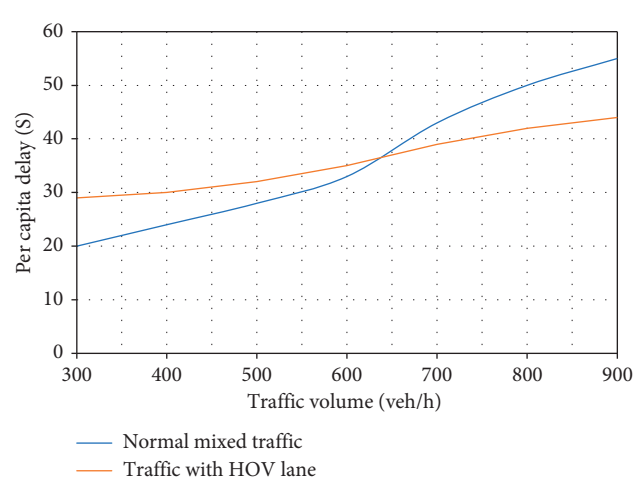


Figure 4: Per capita delay under different driving conditions at $q_b = 180$ and $n_b = 3$.

Table 2: Thresholds for different traffic volumes of high-occupancy vehicles.

Traffic volumes of high-occupancy vehicles (veh/h)	Number of passengers per high-occupancy vehicle (p/veh)	Threshold (veh/h)
150	2	592
	3	580
	4	566
	>4	560
160	2	615
	3	600
	4	582
	>4	575
180	2	650
	3	640
	4	628
	>4	614
200	2	693
	3	675
	4	662
	>4	648

and T_{a0} is the average travel time of public transportation travelers.

2.5.2. Transportation Mode Transfer. After the implementation of the HOV lane priority dynamic control, the trips of low-occupancy vehicles (private cars) in the road network will shift to high-occupancy vehicles (carpool). Through presurvey, the travel mode transfer function under different states is obtained, as shown in Table 3.

2.5.3. Traffic Information Acquisition in Traffic with HOV Lanes. Assume there are m sections in the road network and n cars in each section, and then, the total vehicle mileage of the road network in the posterior operating state (after setting HOV lanes) can be expressed as follows:

$$S_1 = \sum_{j=0}^{m} \sum_{i=0}^{n} S_{ij1}, \tag{9}$$

where S_{ij1} is the mileage of the i-th vehicle in the j-th section of the road network and S_1 is the total mileage of the vehicles.

Then, the total travel time in the v could be calculated as follows:

$$PI_1 = P_{a1} * T_{a1} + P_{b1} * T_{b1} + P_{c1} * T_{c1}, \tag{10}$$

where PI_1 is the total travel time, P_{a1} is the number of noncarpooling travelers, T_{a1} is the average travel time of noncarpooling travelers, P_{b1} is the number of carpooling travelers, T_{b1} is the average travel time of carpooling travelers, P_{c1} is the number of public transportation travelers, and T_{a1} is the average travel time of public transportation travelers.

2.5.4. Operating Efficiency Evaluation. The percentage reduction in total travel mileage can be expressed as follows:

$$M = \frac{\sum_{j=0}^{m} \sum_{i=0}^{n} S_{ij0} - \sum_{j=0}^{m} \sum_{i=0}^{n} S_{ij1}}{\sum_{j=0}^{m} \sum_{i=0}^{n} S_{ij0}}. \tag{11}$$

Similarly, the percentage reduction in per capita travel time can be expressed as follows:

$$p = \frac{PI_0/(P_{ao} + P_{bo} + P_{co}) - PI/(P_{a1} + P_{b1} + P_{c1})}{PI_0/(P_{ao} + P_{bo} + P_{co})}. \tag{12}$$

2.5.5. Energy Consumption Evaluation. Evaluation and comparison of road network energy consumption before and after HOV lane priority dynamic control is of great significance for the promotion and application of HOV lanes. According to the literature [22, 23], the energy consumption of different transportation modes in cities is summarized in Tables 4 and 5.

According to the actual situation of the investigated city, the energy consumption of noncarpooling private car trips, carpooling private car trips, and public bus trips with single compartment was selected as 10.17 (25%), 3.67 (75%), and 1.24 (50%), respectively.

The total energy-saving rate of road network can be calculated by the following equation:

$$H = \frac{\left(\alpha_m \cdot P_{a0} \cdot J_{a0} \cdot Q_{a0} + \beta_n \cdot P_{b0} \cdot J_{b0} \cdot Q_{b0} + \gamma_g \cdot P_{c0} \cdot J_{c0} \cdot Q_{c0}\right) - \left(\alpha_m \cdot P_{a1} \cdot J_{a1} \cdot Q_{a1} + \beta_n \cdot P_{b1} \cdot J_{b1} \cdot Q_{b1} + \gamma_g \cdot P_{c1} \cdot J_{c1} \cdot Q_{c1}\right)}{\alpha_m \cdot P_{a0} \cdot J_{a0} \cdot Q_{a0} + \beta_n \cdot P_{b0} \cdot J_{b0} \cdot Q_{b0} + \gamma_g \cdot P_{c0} \cdot J_{c0} \cdot Q_{c0}}, \tag{13}$$

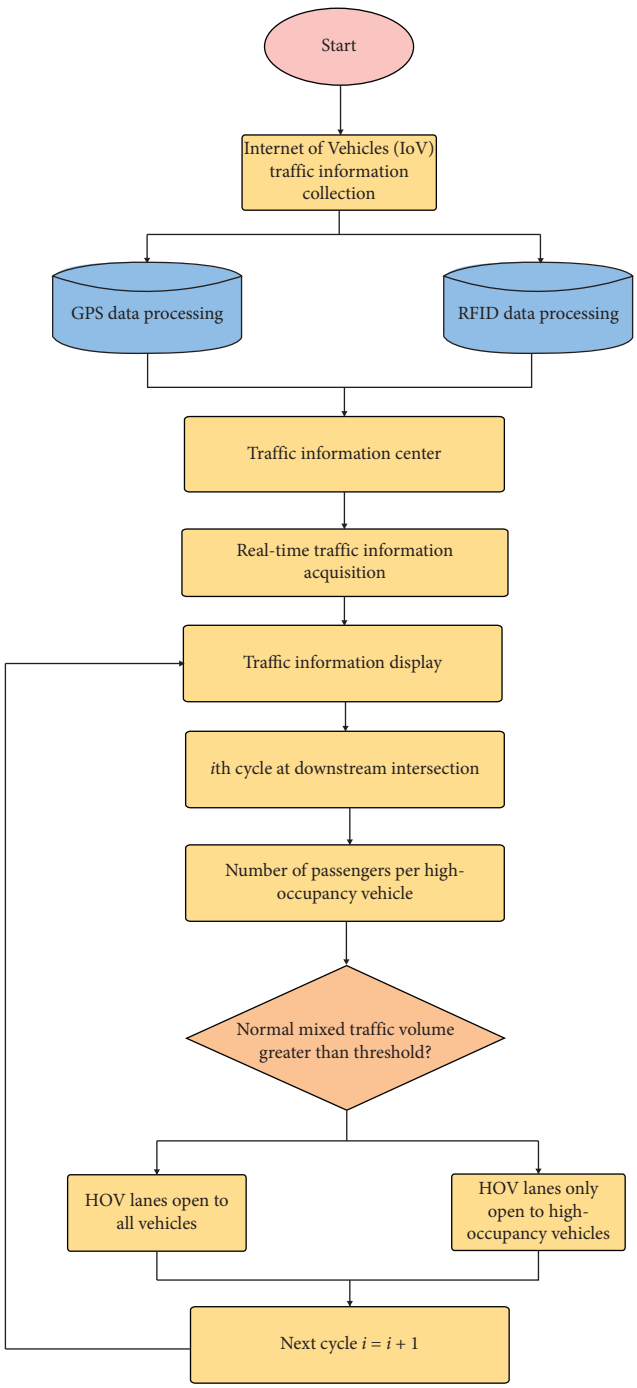


Figure 5: Flow chart of HOV dynamic control process.

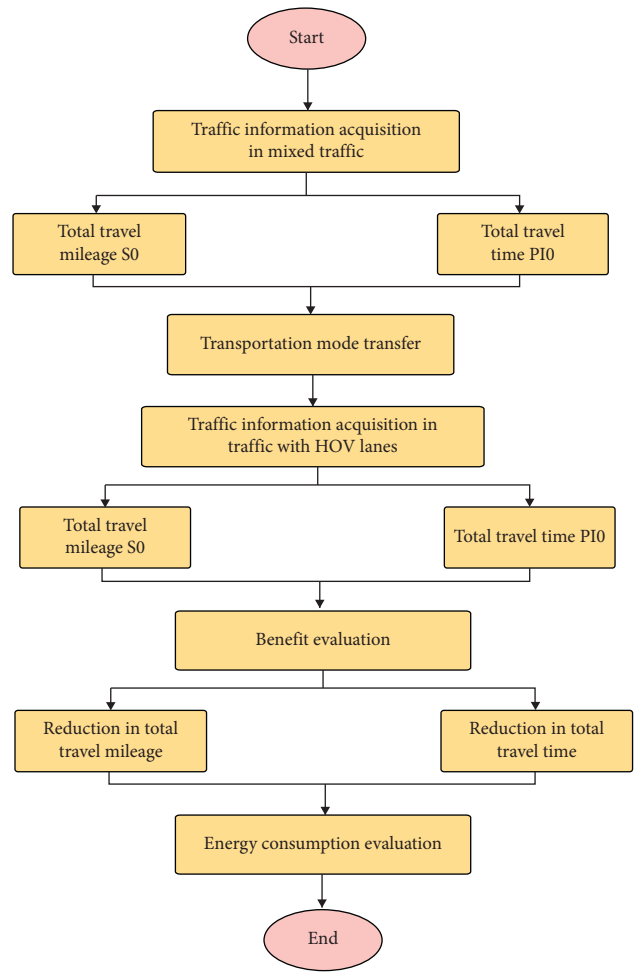


Figure 6: The evaluation process of implementing HOV lane priority dynamic control.

Table 3: The state functions of the transfer from private car to carpool.

State	Percentage increase in travel time (k)	Transfer function $f(k)$
1	<10%	$f(k) = -3.84k + 0.6293$
2	10%–30%	$f(k) = -1.179k + 0.3765$
3	30%–40%	$f(k) = -0.9735k + 0.4224$
4	40%–50%	$f(k) = -0.6045k + 0.0825$
5	>50%	$f(k) = -0.7168k + 0.0029$

where H is the total energy-saving rate of road network; α_m is the energy consumption coefficient of noncarpooling trips, $10.17 \times 10{-}5$ L/p·m; β_n is the energy consumption coefficient of carpooling trips, $3.76 \times 10{-}5$ L/p·m; γ_g is the energy consumption coefficient of public bus trips, $1.24 \times 10{-}5$ L/p·m; J_{a0} is the average travel mileage of noncarpooling trips in the initial operating state (mixed traffic), m; J_{b0} is the average travel mileage of carpooling trips in the initial operating state (mixed traffic), m; J_{c0} is the average travel mileage of public bus trips in the initial operating state (mixed traffic), m; Q_{a0} is the number of vehicles of noncarpooling trips in the initial operating state (mixed traffic), veh; Q_{b0} is the number of vehicles of carpooling trips in the initial operating state (mixed traffic), veh; Q_{c0} is the number of vehicles of public bus trips in the initial operating state (mixed traffic), veh; J_{a1} is the average travel mileage of noncarpooling trips in the posterior operating state (after setting HOV lanes), m; J_{b1} is the average travel mileage of carpooling trips in the posterior operating state (after setting HOV lanes), m; J_{c1} is the average travel mileage of bus trips in the posterior operating state (after

Table 4: Energy consumption comparison of different types of vehicles.

Vehicle categories	Sedan		Public bus	
	Taxi	Private car	Bus with multiple compartments	Bus with single compartment
Fuel consumption (liter/100 km)	10.2	11	25.5	22.5
Energy consumption (liter/(100 people*km))	7.34	6.78	0.41	0.73
Energy consumption (kg standard coal/(100 people*km))	8.52	7.87	0.48	0.85

Note: coefficient for standard coal to fuel conversion = 0.7895 kg/liter × 1.4714 kg standard coal/kg = 1.1617 kg standard coal/liter.

Table 5: Energy consumption comparison of different types of vehicles with different occupancies.

Vehicle categories		Seating capacities	Energy consumption under different occupancies (liter/100 people*km)			
			25%	50%	75%	100%
Sedan	Taxi	4	7.159	3.759	2.68	2.066
	Private car	5	10.17	5.09	3.76	2.95
Public bus	Bus with multiple compartments	80	1.28	0.70	0.51	0.41
	Bus with single compartment	40	2.25	1.24	0.90	0.73

Table 6: Traffic information of the selected roads.

Traffic direction	Road code	Road name	One-way	No. of lanes	Peak hour traffic volume (pcu/h)	Length (m)
Northbound/ Southbound	1	Changjiang North Road	No	6	1548	4120
	2	Xingyuan Middle Road	No	6	1942	4360
	3	Tangnan Road	No	6	1920	3730
	4	Tongyang Road (extended to Jiefang North Road)	No	4	1152	4920
Eastbound/ Westbound	5	Tongjiang Road	No	8	1736	2710
	6	Renmin East Road	No	6	2984	2330
	7	Xueqian East Road	No	6	2275	2340
	8	Yongle East Road	No	6	1697	3690
	9	South Ring Road	No	6	7934	3470

setting HOV lanes), m; Q_{a1} is the number of vehicles of noncarpooling trips in the posterior operating state (after setting HOV lanes), veh; Q_{b1} is the number of vehicles of carpooling trips in the posterior operating state (after setting HOV lanes), veh; and Q_{c1} is the number of vehicles of public bus trips in the posterior operating state (after setting HOV lanes), veh.

3. Case Study-HOV Lane Dynamic Control in Wuxi City, China

3.1. HOV Lane Dynamic Control Scheme in Wuxi City. Wuxi, Jiangsu, is the first city in China to implement HOV lane priority dynamic management and has achieved good results. Both Liangxi District and Xinwu District of Wuxi City are located in the southeast of Jiangsu Province and are located in the city center of Wuxi. The current resident population of Liangxi District is 950,000, and the current resident population of Xinwu District is 364,400. On May 16, 2014, China's first dedicated carpooling lane "HOV lane" was introduced in Wuxi City.

In field survey, the following roads shown in Table 6 were selected to investigate the threshold of setting HOV lanes.

According to the survey, carpooling trips account for 5% of the total private car trips in Wuxi City. Firstly, equation (3) was used to calculate the traffic speed of each road and determine the average delay time considering the signal control of the intersection to obtain the operating efficiency of private car trips in the area, as shown in Table 7.

The operation efficiency of the intersections in the investigated area is summarized in Table 8, and the travel time and travel mileage of different travel modes in the initial operating state are summarized in Table 9.

Travelers in Wuxi City can make reservations online according to their travel time and route in advance. The online system will automatically match according to user needs and supply. This section will scientifically evaluate the efficiency of Wuxi carpooling trips.

Carpooling travel time is mainly composed of waiting time, travel time, and detour time. The travel time is determined by the travel path. The waiting time and detour time are mainly determined by the time and space distribution density of carpooling travelers, as shown in Table 10.

Table 7: Operation efficiency of the road network in the investigated area.

Road code	Road name	Traffic volume per hour (pcu/h)	Length (m)	Free-flow speed (km/h)	Traffic speed (km/h)
1	Changjiang North Road	1548	4120	60	54
2	Xingyuan Middle Road	1942	4360	60	56
3	Tangnan Road	1920	3730	50	45
4	Tongyang Road (extended to Jiefang North Road)	1152	4920	40	32
5	Tongjiang Road	1736	2710	80	75
6	Renmin East Road	2984	2330	40	35
7	Xueqian East Road	2275	2340	40	32
8	Yongle East Road	1697	3690	40	36
9	South Ring Road	7934	3470	80	71
10	East Ring Road	10451	6200	80	73
11	Xingchang South Road	3221	4450	80	67

Table 8: Operation efficiency of the intersections.

Intersection code	Name of the intersecting roads	Traffic volume (pcu/h)	Signal cycle (s)	Per capita delay (s)
1	Tongyang Road, Tongjiang Road, and Xingyuan Middle Road	561	80	25.36
2	Xingyuan Middle Road and Renmin East Road	1484	125	30.91
3	Tongyang Road and Renmin East Road	464	60	22.15
4	Tangnan Road and Renmin East Road	1434	No signal	—
5	Xingyuan Middle Road and Renmin East Road	473	80	30.15
6	Tongyang Road and Xueqian East Road	397	60	18.36
7	Tangnan Road and Xueqian East Road	1569	80	42.36
8	Xingyuan Middle Road and Xueqian East Road	512	80	35.12
9	Changjiang North Road and Xueqian East Road	833	120	40.48
10	Tongyang Road and Yongle East Road	483	55	18.23
11	Tangnan Road and Yongle East Road	1829	110	37.52
12	Xingyuan Middle Road and Yongle East Road	1669	75	24.58
13	Changjiang North Road and Yongle East Road	948	120	27.93
14	Tongyang Road and South Ring Road	617	No signal	—
15	Tangnan Road and South Ring Road	1973	No signal	—
16	Xingyuan Middle Road and South Ring Road	1967	No signal	—
17	Changjiang North Road and South Ring Road	1132	No signal	—

3.2. *Efficiency and Energy Consumption Evaluation of HOV Lane Priority Dynamic Control in Wuxi City*

3.2.1. Preliminary Analysis of Operation Efficiency. After setting up HOV lane, the traffic speed of private cars on ordinary lanes (noncarpooling lanes) is shown in Table 11.

After setting up HOV lane, the traffic speed of private cars on the carpooling lanes is shown in Table 12.

The travel time of noncarpooling private car trips and carpooling private car trips is calculated and summarized in Table 13.

3.2.2. Travel Mode Transfer. Based on of the influence of the increment of travel time on carpooling behavior, after setting HOV lanes, the increase in carpooling travel time is 21.7% compared with that of noncarpooling travel time. The transfer ratio of low-occupancy travel mode to high-occupancy travel mode is 12.0%.

3.2.3. Operation Efficiency Analysis after Travel Mode Transferring. The travel time and travel mileage of different travel modes after travel mode transferring are summarized in Table 14. Moreover, the comparison of number of trips, number of travelers, travel time, and travel mileage in initial (mixed traffic) and posterior operating state (after travel mode transferring) is presented in Figures 7–10, respectively.

3.2.4. Final Evaluation of Operation Efficiency and Energy Consumption. The percentage reduction in total travel mileage can be calculated as follows:

$$M = \frac{80704908 - 76729943}{80704908} = 4.93\%. \quad (14)$$

The percentage reduction in per capita travel time can be calculated as follows:

$$p = \frac{1138 - 1089}{1138} = 4.27\%. \quad (15)$$

Table 9: Travel time and travel mileage of different travel modes in the initial operating state (mixed traffic).

Code	Travel mode	Initial operating state (mixed traffic)			
		Number of vehicles (veh)	Number of travelers (p)	Travel time (s)	Travel mileage (m)
1	Noncarpooling private car trips	23188	23188	1054	3243
2	Carpooling private car trips	384	1360	1625	4970
3	Public bus trips	816	6796	1326	4409
Sum	—	24388	31344	—	—

Table 10: Travel time and detour time for different proportions of carpool trips.

Proportion of carpooling trips (%)	Area (km^2)	No. of vehicles (veh)	Time density (veh/s)	Coverage (m)	Waiting time (s)	Detour time (s)	Total (s)
5	14.87	406	8.86	267.78	298	658	955
10.0	14.87	813	4.43	189.35	210	359	569
15.0	14.87	1219	2.95	154.60	172	259	431
20.0	14.87	1626	2.21	133.89	149	209	358
25.0	14.87	2032	1.77	119.75	133	180	313
30.0	14.87	2439	1.48	109.32	121	160	281
35.0	14.87	2845	1.27	101.21	112	145	258
40.0	14.87	3252	1.11	94.67	105	135	240
45.0	14.87	3658	0.98	89.26	99	126	226
50.0	14.87	4065	0.89	84.68	94	120	214

Table 11: The traffic speed of private cars on ordinary lanes.

Code	Road name	Traffic volume per hour (pcu/h)	Length (m)	Free-flow speed (km/h)
1	Changjiang North Road	4120	60	40
2	Xingyuan Middle Road	4360	60	44
3	Tangnan Road	3730	50	34
4	Tongyang Road (extended to Jiefang North Road)	4920	40	28
5	Tongjiang Road	2710	80	48
6	Renmin East Road	2330	40	27
7	Xueqian East Road	2340	40	22
8	Yongle East Road	3690	40	18
9	South Ring Road	3470	80	42
10	East Ring Road	6200	80	36
11	Xingchang South Road	4450	80	45

Table 12: The speed of private cars on HOV lanes.

Code	Road name	Traffic volume per hour (pcu/h)	Length (m)	Free-flow speed (km/h)
1	Changjiang North Road	4120	60	60
2	Xingyuan Middle Road	4360	60	56
3	Tangnan Road	3730	50	50
4	Tongyang Road (extended to Jiefang North Road)	4920	40	38
5	Tongjiang Road	2710	80	78
6	Renmin East Road	2330	40	35
7	Xueqian East Road	2340	40	32
8	Yongle East Road	3690	40	40
9	South Ring Road	3470	80	71
10	East Ring Road	6200	80	75
11	Xingchang South Road	4450	80	74

Table 13: Travel time and travel mileage of different travel modes in posterior operating state (with HOV lanes).

Code	Travel modes	Posterior operating state (with HOV lanes)			
		Number of vehicles (veh)	Number of travelers (p)	Travel time (s)	Travel mileage (m)
1	Noncarpooling private car trips	23188	23188	1060	3243
2	Carpooling private car trips	384	1360	1291	4970
3	Public bus trips	816	6796	1287	4409
Sum	—	24388	31344	—	—

Table 14: Travel time and travel mileage of different travel modes after travel mode transferring in the posterior operating state (with HOV lanes).

Code	Travel mode	Posterior operating state (with HOV lanes)			
		Number of vehicles (veh)	Number of travelers (p)	Travel time (s)	Travel mileage (m)
1	Noncarpooling private car trips	20405	20405	1006	3243
2	Carpooling private car trips	1568	4143	1225	4438
3	Public bus trips	816	6796	1256	4409
Sum	—	22789	31344	—	—

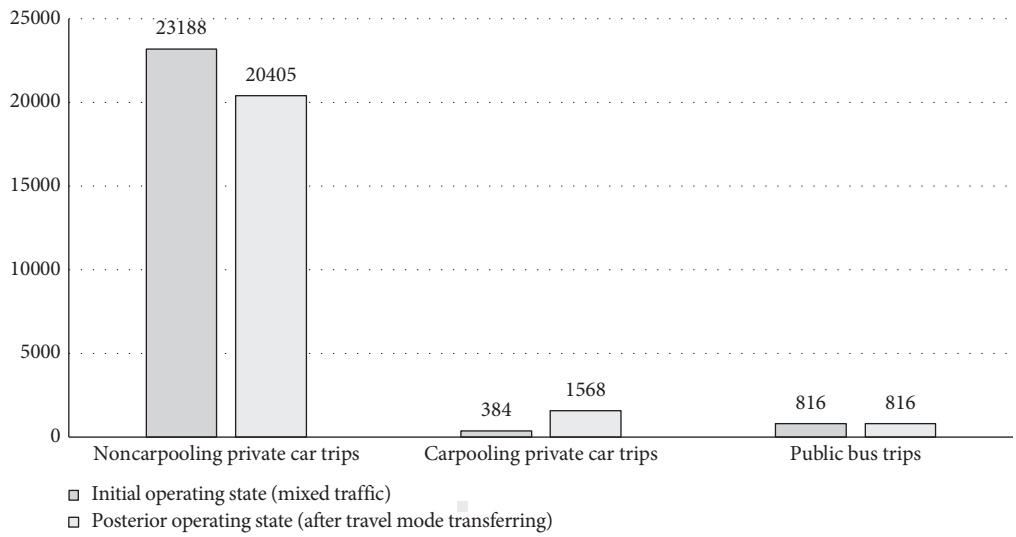


Figure 7: Comparison of number of trips before and after implementing HOV lanes priority dynamic control for different travel modes.

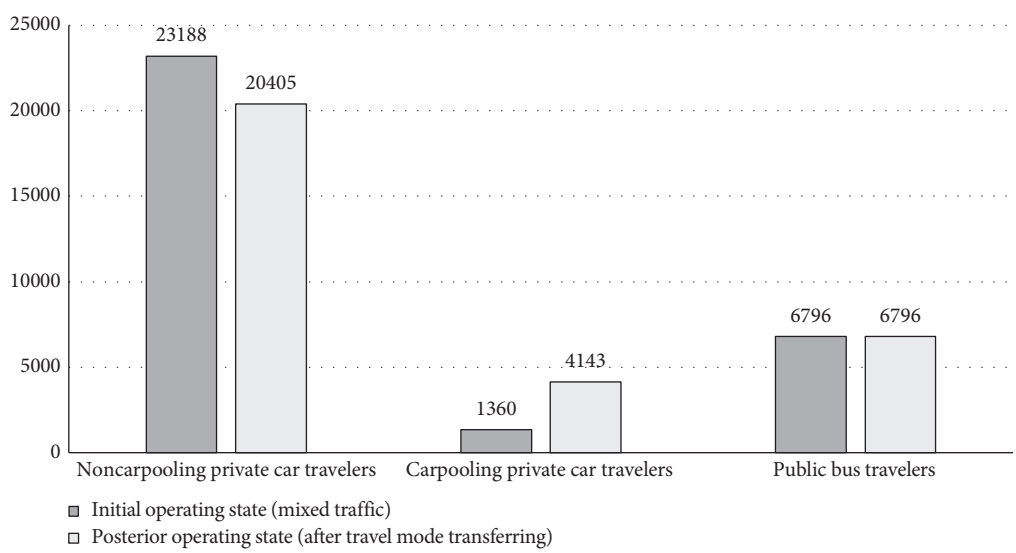


Figure 8: Comparison of number of travelers before and after implementing HOV lanes priority dynamic control for different travel modes.

The total energy-saving rate of the road network can be calculated as follows:

$$H = \frac{177735785.9 - 138709510.2}{177735785.9} = 21.96\%. \quad (16)$$

To sum up, after HOV lane priority dynamic control was implemented and travel mode transferring was completed, the total mileage of vehicles on the road network was saved by 4.93% and the per capita travel time decreased from 1138 s to 1089 s, with a reduction rate of 4.27%; the total

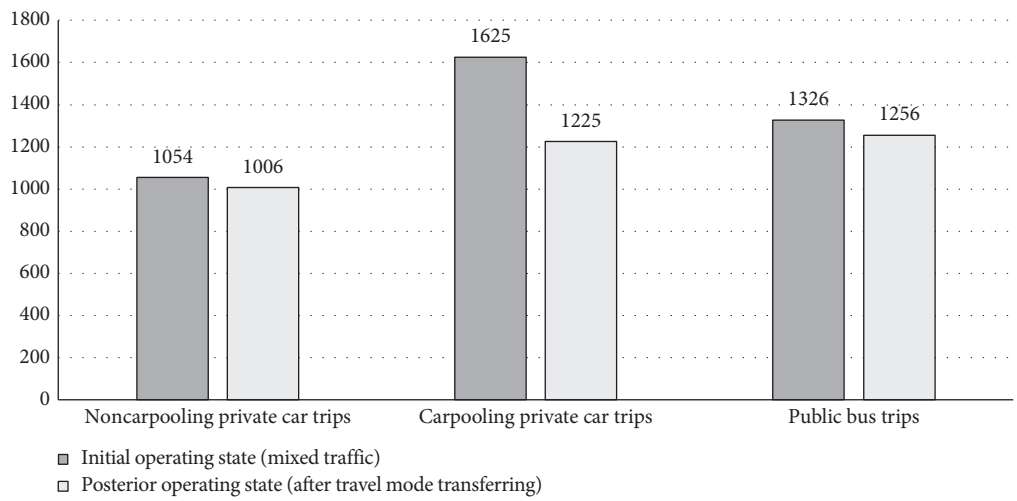


Figure 9: Comparison of travel time before and after implementing HOV lanes priority dynamic control for different travel modes.

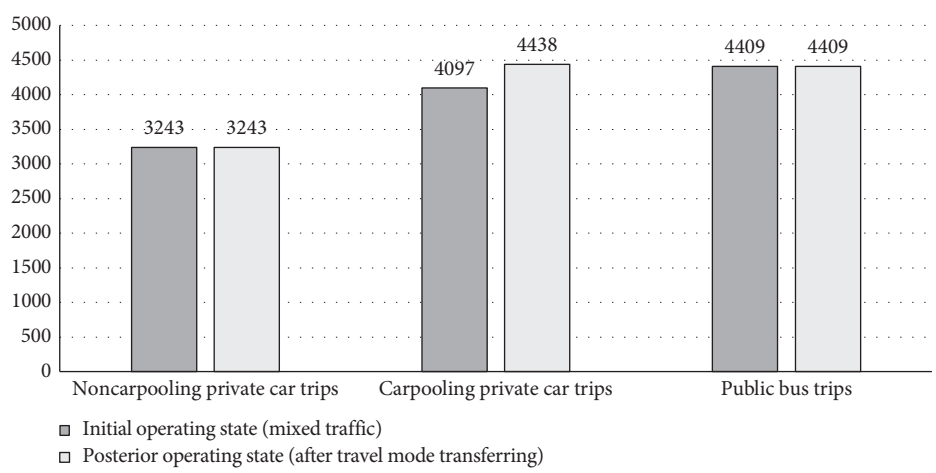


Figure 10: Comparison of travel mileage before and after implementing HOV lanes priority dynamic control for different travel modes.

energy consumption of trips in the network was reduced by 39,26275.62 L, with a saving rate of 21.96%.

4. Conclusions

This paper first summarized the current research on HOV lane implementation and analyzed and identifies the threshold of setting road HOV lane dynamic control under the connected vehicle environment. Moreover, the HOV lane priority dynamic control process was determined, and the operating efficiency and energy consumption evaluation method was proposed. A case study in Wuxi City, China, was carried out. The results showed that, after implementing the HOV lane priority dynamic control, the total mileage of road network vehicles was saved by 4.93%, the average travel time per capita was reduced by 4.27%, and the total energy-saving rate of road network travel was 21.96%. From the results, it can be seen that dynamic control of HOV lanes with priority for carpooling can save the total mileage of vehicles in the road network, the average travel time per capita, and the total energy consumption, thus improving travel efficiency. With the further development of IoV technology, the dynamic management and control of HOV lanes can be combined with real-time dynamic reversible lane technology to achieve real-time dynamic control of HOV reversible lanes to achieve precise control optimization.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

Guiliang Zhou conceived and designed the paper. Guiliang Zhou and Pengsen Hu wrote the paper. Lina Mao and Pengsen Hu conducted the model. Feng Sun and Xu Bao collected traffic data.

Acknowledgments

This research was supported by the Open Fund for Jiangsu Key Laboratory of Traffic and Transportation Security (Huaiyin Institute of Technology) (TTS2020-05 and TTS2020-09), Enterprise-University-Research Institute Collaboration Project of Jiangsu Province (DH20190231), Graduate Innovative Projects of Jiangsu Province (KYLX15_0148), National Natural Science Foundation of China (61573098 and 51308246), University Natural Science Major Basic Project of Jiangsu Province (15KJA580001), Youth Foundation of Huaiyin Institute of Technology (HGC1408), and Natural Science Foundation of Jiangsu Province, China (BK20171426).

References

[1] M. Menendez and C. F. Daganzo, "Effects of HOV lanes on freeway bottlenecks," *Transportation Research Part B: Methodological*, vol. 41, no. 8, pp. 809–822, 2007.

[2] D. Brownstone, A. Ghosh, T. F. Golob, C. Kazimi, and D. Van Amelsfort, "Drivers' willingness-to-pay to reduce travel time: evidence from the San Diego I-15 congestion pricing project," *Transportation Research Part A: Policy and Practice*, vol. 37, no. 4, pp. 373–387, 2003.

[3] D. Diamond, "The impact of government incentives for hybrid-electric vehicles: evidence from US states," *Energy Policy*, vol. 37, no. 3, pp. 972–983, 2009.

[4] K. S. Gallagher and E. Muehlegger, "Giving green to get green? Incentives and consumer adoption of hybrid vehicle technology," *Journal of Environmental Economics and Management*, vol. 61, no. 1, pp. 1–15, 2011.

[5] V. Pravin and H. Han, "Effectiveness of California's high occupancy vehicle (HOV) system," *Urban Transport*, vol. 08, no. 4, pp. 79–93, 2010.

[6] J. Dahlgren, "High occupancy vehicle lanes: not always more effective than general purpose lanes," *Transportation Research Part A: Policy and Practice*, vol. 32, no. 2, pp. 99–114, 1998.

[7] R. G. Gutierrez, J. M. Linhart, and J. S. Pitblado, "From the help desk: local polynomial regression and stata plugins," *The Stata Journal: Promoting Communications on Statistics and Stata*, vol. 3, no. 4, pp. 412–419, 2003.

[8] L. Hultkrantz and X. Liu, "Green cars sterilize congestion charges: a model analysis of the reduced impact of Stockholm road tolls," *Transport Policy*, vol. 21, pp. 110–118, 2012.

[9] J. Kahn, "Gasoline prices and the used automobile market: a rational expectations asset price approach," *Quarterly Journal of Economics*, vol. 101, pp. 41–62, 1986.

[10] M. E. Kahn, "Do greens drive hummers or hybrids? environmental ideology as a determinant of consumer choice," *Journal of Environmental Economics and Management*, vol. 54, no. 2, pp. 129–145, 2007.

[11] W. Chen and B. Chen, "Introducing the HOV priority concept into Chinese traffic planning and management," *City Planning Review*, vol. 27, no. 6, pp. 93–96, 2003.

[12] W. Wang, "Carpool priority planning methods in central area of large cities," *Journal of Chongqing Jiaotong University (Natural Science Edition)*, vol. 29, no. 2, pp. 248–253, 2010.

[13] G. Bi, *Research on Feasibility and Application of HOV Lane Based on Disaggregate model*, Southeast University, Nanjing, China, 2010.

[14] Y. Wu and S. Pan, "On the implementation and prospect of HOV and BRT mixed channel," *Science and Technology Innovation*, vol. 5, no. 5, p. 124, 2012.

[15] Y. Zhou, "Research on information service demand and system framework of carpool priority special lane," *China Transportation Informatization*, vol. 13, no. s1, pp. 51–53, 2015.

[16] J. Zhan, P. A. N. Xiao-dong, and A. Gao, "The Research for the Use and Design of HOV lane," *Traffic and Transportation*, vol. 1, no. 1, pp. 17–20, 2007.

[17] J. Zhang, T. Zhang, and F. Han, "Analysis of the setting of HOV lanes based on traffic efficiency," *Transportation Science and Engineering*, vol. 28, no. 4, pp. 96–100, 2012.

[18] C. Li, J. Chen, and S. Deng, "Characteristics and feasibility analysis of setting HOV lanes," *City Traffic*, vol. 10, no. 6, pp. 58–65, 2012.

[19] Z. Hu, T. Bao, P. U. Zheng, and B. Li, "Feasibility study on HOV lane setting based on traveling total utility," *Comprehensive Transportation*, vol. 39, no. 8, pp. 62–67, 2017.

[20] L. I. Peng-Fei, S. Han, and L. I. N. Hang-Fei, "The application of HOV lane in Shanghai urban traffic management," *Traffic and Transportation*, vol. 2, no. 2, pp. 38–41, 2007.

[21] Y. Wei, Q. Tang, and R. Chen, "Research onthe application effect of HOV lane design——taking Chongqing Xuefu Street as an example," *Transportation Technology and Economy*, vol. 19, no. 6, pp. 12–16, 2017.

[22] T. Zhang, *Study on Different Urban Transport Modes'energy consumption*, Beijing Jiaotong University, Beijing, China, 2010.

[23] L. Yang and H. Wen, *Transportation and Energy*, People's Communications Press, Beijing, China, 2016.

Hindawi
Journal of Advanced Transportation
Volume 2020, Article ID 8851485, 17 pages
https://doi.org/10.1155/2020/8851485

Research Article

A Fatigue Driving Detection Algorithm Based on Facial Motion Information Entropy

Feng You,[1,2] Yunbo Gong,[1] Haiqing Tu,[1] Jianzhong Liang,[1] and Haiwei Wang[3]

[1]*School of Civil Engineering and Transportation, South China University of Technology, Guangzhou 510640, China*
[2]*State Key Lab of Subtropical Building Science, South China University of Technology, Guangzhou, China*
[3]*School of Transportation and Economic Management, Guangdong Communication Polytechnic, Guangzhou 510650, China*

Correspondence should be addressed to Haiwei Wang; whw2046@126.com

Received 8 April 2020; Revised 14 May 2020; Accepted 25 May 2020; Published 15 June 2020

Academic Editor: Dongfang Ma

Copyright © 2020 Feng You 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.

Research studies on machine vision-based driver fatigue detection algorithm have improved traffic safety significantly. Generally, many algorithms asses the driving state according to limited video frames, thus resulting in some inaccuracy. We propose a real-time detection algorithm involved in information entropy. Particularly, this algorithm relies on the analysis of sufficient consecutive video frames. First, we introduce an improved YOLOv3-tiny convolutional neural network to capture the facial regions under complex driving conditions, eliminating the inaccuracy and affections caused by artificial feature extraction. Second, we construct a geometric area called Face Feature Triangle (FFT) based on the application of the Dlib toolkit as well as the landmarks and the coordinates of the facial regions; then we create a Face Feature Vector (FFV), which contains all the information of the area and centroid of each FFT. We use FFV as an indicator to determine whether the driver is in fatigue state. Finally, we design a sliding window to get the facial information entropy. Comparative experiments show that our algorithm performs better than the current ones on both accuracy and real-time performance. In simulated driving applications, the proposed algorithm detects the fatigue state at a speed of over 20 fps with an accuracy of 94.32%.

1. Introduction

Every year, road traffic accidents cause severe damage to human health. According to the statistics from the WHO, fatigue driving is one of the main reasons behind road traffic accidents [1]. National Sleep Foundation points out that about 32% of drivers have at least one fatigue driving experience per month [2]. Fatigue driving is a harmful threat to the driver and other traffic participants. Countries all over the world have made laws to tackle this problem. For example, the Chinese Road Traffic Safety Law stipulates that "Drivers are not allowed to drive continuously for more than 4 hours, and the rest period between every two long-duration driving should be no less than 20 minutes" [3]. In Europe, the law requires that "Drivers should stop and rest for every 4.5 hours of continuous driving, and the rest period should be no less than 20 minutes" [3]. In the United States, the law provision is that "The cumulative maximum daily driving time must not exceed 11 hours, and the continuous daily rest time must not be less than 10 hours" [4]. As mentioned above, fatigue driving is solely associated with driving duration. It is subjective to determine whether the driver is in fatigue state or not without sufficient quantified indexes and reliable data analysis.

According to relevant data, heavy road traffic accidents caused by fatigue driving account for about 50% of all road traffic accidents [5]. Therefore, research on fatigue driving detection is inevitable. The detection algorithms are of the following types.

1.1. Detection Methods Based on Physiology and Behavior. Detection methods based on physiology and behavior are those that judge the driver's status by installing an intrusive sensor and collecting data that characterizes the driver's physiology, psychology, and driving operations. These detection methods include EEG signal detection [6], ECG signal detection [7], pulse beat detection [8], and EMG signal detection [9].

1.2. Detection Methods Based on Machine Vision. With distinctive characteristics of the vehicle motion and the behaviors of the driver obtained, this method assesses the driver's fatigue status. Machine vision-based detection has become the widely used method in fatigue driving detection due to its noninvasion and higher accuracy. This method applies core technologies including face detection, eye positioning, and fatigue assessment. Yan et al. [10] used the mask to locate the eye position by obtaining the driver's facial image and used PERCLOS to evaluate the driver's fatigue state. This method has better performance on individuals with conspicuous features, but the fabrication of the mask has a significant influence on the generalization performance of the model. Niu and Wang [11] divided the face image in the sequence image into nonoverlapping blocks of the same size. Then, they managed to use Gabor wavelet transform to extract multiscale features. In order to select the most recognizable ones, they applied AdaBoost algorithm. This method can effectively recognize different genders and postures under various illumination conditions. Using "bright eye effect," Bergasa and Nuevo [12] located eye position with active near-infrared light source equipment. They used finite-state machine to confirm whether the eye is closed. They also applied fuzzy system to evaluate the fatigue state. However, Bergasa's algorithm depends highly on hardware level; on the other hand, the effectiveness of the "bright eye effect" strictly relies on surrounding light conditions. You et al. [13] applied the CAMShift tracking algorithm to make the targeted areas detectable even they were under occlusion. Then the eye feature points were obtained according to the specific proportion relationship of the facial organs. Finally, they used PERCLOS to determine driver fatigue state.

1.3. Detection Methods Based on Information Fusion. Any fatigue detection method has its advantages and disadvantages. So comprehensive monitoring of driver fatigue status by various methods is promising. "AWAKE" [14] launched by the European Union is a driving behavior comprehensive monitoring system. It used many sensors such as images and pressures to synthesize the driver's eye movement, the direction of eyesight, steering wheel grip, and other driving conditions. Then, it made comprehensive detection and evaluation. Seeing Machines [15] conducts multifeature information fusion by detecting facial features such as driver's head posture, eyelid movement, gaze direction, and pupil diameter. It completed real-time monitoring of driver fatigue status.

Although the technology of fatigue detection has made great progress, it can be better:

(i) Physiology-based driver fatigue detections require a variety of additional monitoring devices or equipment. It would not only reduce comfort during driving but also make the collected data costly and vulnerable, which has set back the popularization of these methods.

(ii) If the light condition changes or the driver's face is partially occluded, for example, wearing glasses or sunglasses, AdaBoost fails to accurately locate the face position and give the alarm to the driver promptly.

(iii) At present, the commonly used algorithms are based on PERCLOS, which judge fatigue by opening and closing state of the driver's eyes. However, when the driver's eyes are too small, the algorithms are easy to misjudge. Moreover, other fatigue indicators are less commonly used due to lower reliability and less robustness.

As above literature studies discussed, results of the driving fatigue detection have defects of high intrusion, low robustness, and low reliability. Therefore, we propose a fatigue driving detection algorithm based on facial motion information entropy. The innovations are as follows:

(i) We design a driver's face detection architecture based on the improved YOLOv3-tiny convolutional neural network and train the network with the open-source data set WIDER FACE [16]. Compared with other deep learning algorithms, such as YOLOv3 [17] and MTCNN [18], the algorithm based on the improved YOLOv3-tiny network is more accurate and simplified. It has fewer calculations and thus is easy to transplant to other mobiles.

(ii) We used the Dlib toolkit to extract facial feature points recognized by improved YOLOv3-tiny convolutional neural network. Then we created the FFT after analyzing the characteristics of the eye and mouth position. Next, we constructed FFV which contains the overall information of the area and centroid of each FFT. We calculate the FFV of each frame and write it to the database. Thereby a state analysis data set is established. In many research studies the basis for assessing the state of the driver is the recognition result of a single frame or a few frames, which reduce the accuracy of fatigue driving detection. Based on the analysis results of a large number of consecutive frames, we design sliding windows of driving fatigue analysis to obtain the statistical characteristics of the facial motion state. Therefore, the process of driver fatigue can be observed.

(iii) To get rid of the interference that originated from the size differences between every FFT, we introduce the face projection datum plane and apply the projection principle to extract the motion feature points of the face. Then, based on the motion feature points, we propose the facial motion information entropy, which quantitatively characterizes the chaotic degree of the motion feature points of the face. Accordingly, the driver's fatigue state can be judged. At present, the commonly used algorithms are based on PERCLOS [19], which judge fatigue by opening and closing state of the driver's eyes. However, when the driver's eyes are too small, the algorithms are easy to misjudge. Therefore, we

reveal the difference in the motion characteristics between fatigue driving and nonfatigue driving by proposing facial motion information entropy.

This paper is divided into the following seven parts. The first chapter is the introduction. In this part, we introduce the background and research significance of our fatigue driving detection system and the research status from home and abroad. We propose a fatigue driving detection algorithm based on facial motion information entropy with technical innovations. In the second chapter, we explain the algorithm in detail. The structure of this algorithm is a combination of improved YOLOv3-tiny network and Dlib toolkit. The former captures ROI, while the latter obtains facial landmarks and creates a fatigue state data set. We make a description of the definition and calculation method of facial motion information entropy, which is the main index to represent the fatigue state. The third chapter is the experimental analysis. Firstly, the experimental environment and data set are introduced. Then we use qualitative description and quantitative evaluation to measure face detection and feature point location. Finally, we evaluate our fatigue driving detection algorithm in two directions: accuracy and real time. The fourth chapter is the conclusion, which mainly summarizes the main work content of this paper and analyzes the shortcomings of the system and the aspects that need to be improved. Then, we propose the future optimization direction and prospect of the algorithm. Other sections are Data Availability, Conflict of Interests, Acknowledgments, and References.

2. Methodology

The overall pipeline of our approach is shown in Figure 1. The algorithm consists of the following 4 modules.

Face Positioning. The original data source is the real-time camera video. Based on deep learning theory, we apply the improved YOLOv3-tiny network to extract suspected face regions from complex backgrounds.

Feature Vector Extraction. FFT is a geometry area in every frame that contains facial features. Based on the coordinates of the suspected face region, we obtain facial landmarks with the application of the Dlib toolkit and construct FFV by calculating the area and centroid of the driver's FFT.

Data Set Building. According to the FFV extracted in a certain period, the driver state analysis data set is established in chronological order.

Fatigue Judgment. We design a sliding window as a sampler; every time it analyzes several sequential FFVs which match with the related sequential frames by projecting the FFV on the facial projection datum. Afterwards, it loops through all FFVs and outputs a facial motion information entropy corresponding to the current facial feature point set. We then compare the facial motion information entropy with its threshold to evaluate the fatigue state of the driver.

2.1. Face Detection Based on the Improved YOLOv3-Tiny Network. Face detection location is the foundation of driver fatigue detection, and the accuracy of the results has a great impact on the algorithm's performance. So, accurate and rapid face detection is the fundamental task of the driving fatigue detection algorithm. In the traditional face detection algorithm, the face features are mostly based on prespecified features such as Haar and HOG [20, 21]. In terms of Haar features, Viola and Jones [22] propose a joint Haar feature for face detection algorithms. However, image features may lose because of inappropriate face postures, dim light conditions, noise interference, or a partially occluded face, which decreases the robustness and reliability of prespecified feature method. Recently, deep learning theory provides new ways for detection and segmentation [23]. It can be divided into 2 categories: one transfers the target detection model to face detection and segmentation process; the other is the cascade methods, such as MTCNN [24, 25] and Cascade CNN [26]. Compared with the traditional methods [27], the face detection based on convolutional neural network extracts features autonomously instead of man-made operation. With the support of data sets, face detection performance has been greatly improved.

The YOLO [28] (You Only Look Once) model is a fast target detection model based on deep learning [29]. It is a separate end-to-end network that turns target detection into a regression problem. Specifically, we can replace the sliding window in the traditional target detection to the regression method and convolutional neural network (CNN) [30]. This method of feature extraction is less affected by the external environment and has the advantage of extracting target features quickly.

Inspired by the idea of YOLO model, we transform the multiobjective regression into the single target regression, hence reducing the calculation amount. Then, we improve YOLOv3-tiny network to locate suspected face regions.

The YOLOv3-tiny network is a simplified version of YOLOv3, so it has better real time than YOLOv3. It simplifies the YOLOv3 feature detection network darknet-53 to 7 conventional convolution layers and 6 Max Pooling layers and 1 Up Sample layer. The improved network structure is shown in Figure 2. In the figure, "Darknetconv2d BN Leaky" (DBL) is the basic component of the network, "Conv" is the convolution layer, and "Leaky ReLU" is the activation function. Batch normalization (Batch Norm) is a regularization method that guarantees the algorithm convergence and avoids overfitting. Concat sandwiches a sample layer in the middle of two DBL. Nonmaximum suppression (NMS) is to eliminate the extra facial box and locate the best driver's face suspected area.

We consider that the images used for analysis for fatigue driving contain only one face. If the network shows high accuracy in multiface detection, one face detection will be more accurate. So, in the YOLOv3-tiny network training phase, we use the WIDER FACE (Face Detection Data Set and Benchmark) (http://wider-challenge.org/2019.html) [16] data set as the driving data. The WIDER FACE data set includes 32,203 images and 393,703 marked faces, which is one of the most common face databases. The data set

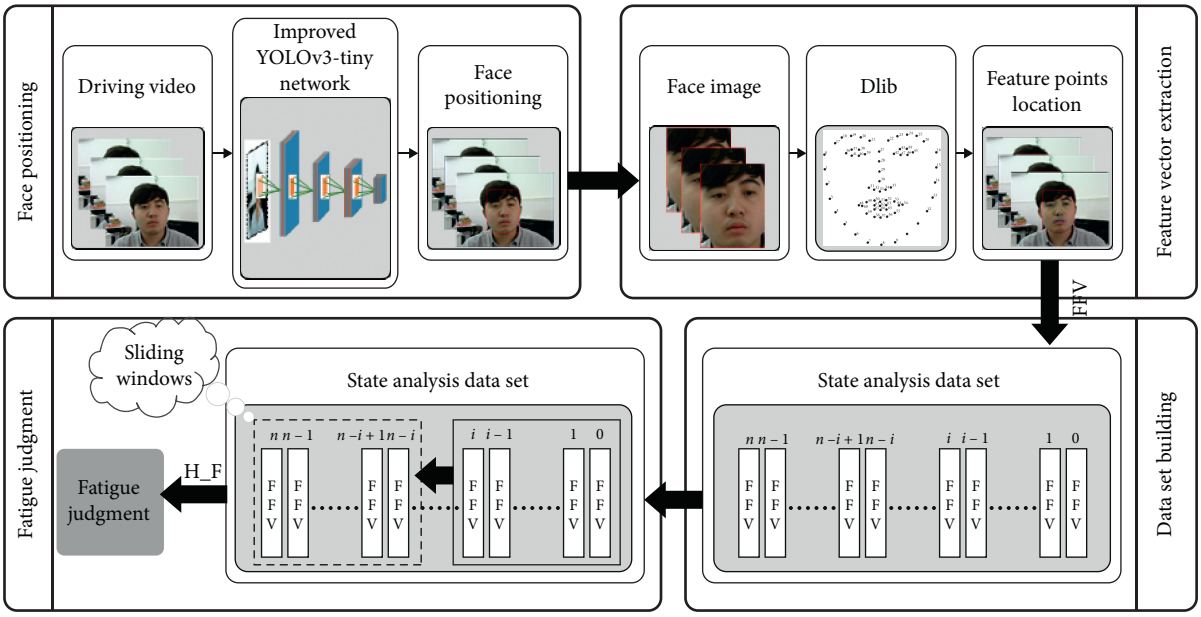


Figure 1: Algorithm structure diagram, where H_F is facial motion information entropy.

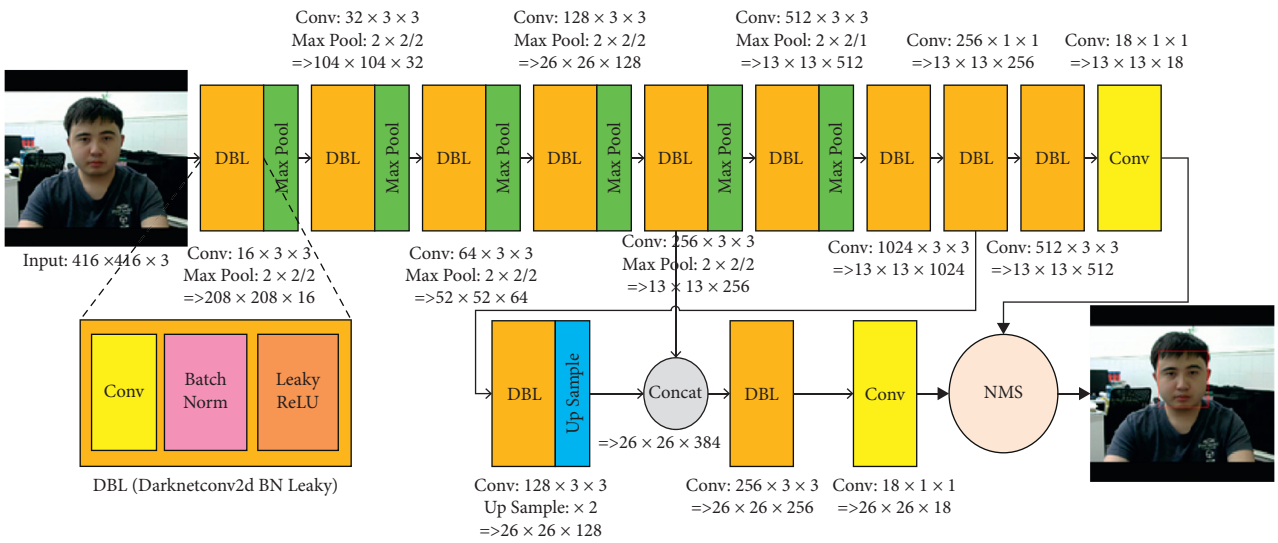


Figure 2: Improved YOLOv3-tiny network structure diagram.

includes different scales, poses, occlusions, expressions, makeup, and lighting, as shown in Figure 3.

The WIDER FACE data set has the following features:

(i) The data set is divided into three types: training set, test set, and verification set, which, respectively, account for 40%, 50%, and 10% of the data set

(ii) There are a large number of faces in each image, which contains an average of 12.2 faces

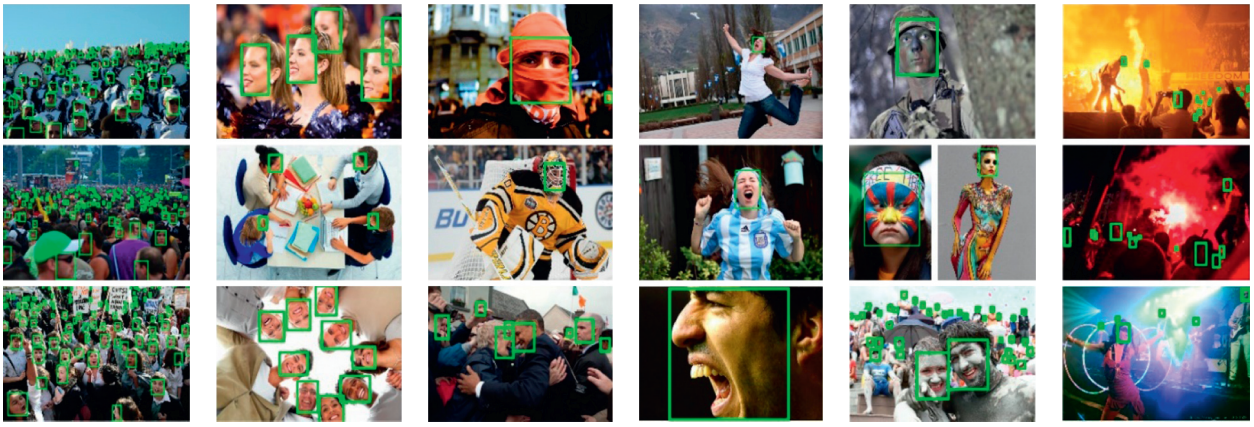
(iii) The data set pictures are high-resolution color images

Firstly, based on the YOLOv3-tiny network, the picture of the WIDER FACE data set is adjusted to 10 different sizes, and every picture is divided into 13×13 grid cells or 26×26 grid cells. Then, we find the location of the driver's face on the nonoverlapping grid cell and classify it. For each grid cell, the network outputs B bounding boxes as well as the corresponding confidence and the conditional probability of the driver's face. Finally, nonmaximal values are used to suppress redundant bounding boxes. The confidence formula is given as

$$\text{score} = P_r\,(\text{Object}) * \text{IOU}_{\text{pred}}^{\text{truth}}, \tag{1}$$

where $P_r\,(\text{Object})$ is the probability of the driver's face. If the face is included, $P_r\,(\text{Object}) = 1$; otherwise $P_r\,(\text{Object}) = 0$. $\text{IOU}_{\text{pred}}^{\text{truth}}$ is the intersection over union (IOU) of the bounding box to the real box.

There are four basic elements in the YOLOv3-tiny network loss function: the central error term of the bounding box, the width and high error term of the



(a) (b) (c) (d) (e) (f)

Figure 3: WIDER FACE data set diagram.

bounding box, the error term of the prediction confidence, and the error term of the prediction category. We managed to use the offline trained YOLOv3-tiny network to extract the accurate face region for further research.

2.2. Driver's Facial Motion Feature Extraction

2.2.1. Face Feature Location Based on the Dlib Toolkit. On the driver's face area located by the improved YOLOv3-tiny network, we used the face key point detection model based on the Dlib-ml [31] library to extract the fine-grained features of a driver's face (as is shown in Figure 4(a)). The Dlib library contains 68 face key points. The testing principle is applying cascading shape regression to check all the key points of the face component.

The face detection process is as follows. Firstly, the feature of the input image is extracted, including the features of the face contour, eyebrows, eyes, nose, and mouth contours. Secondly, the extracted features are mapped to the face feature points through a trained regressor; at this point, an initial shape of the key points of the human face component is generated from its original image. Thirdly, gradient boosting [32] is used to iteratively adjust the initial shape until it matches with the real shape; then the cascaded regressor of each stage is calculated with the least-square method.

The face key point detection method of the Dlib library is based on the ensemble of regression trees (ERT) algorithm [29]. It uses the regression tree set to estimate the face feature points, and the speed of calculation is fast. The detection of 68 key points in each face takes about 1 ms. Similar to [33] and [34], this cascade regressor method is available even though feature points are partially missing in the training sample set. The iterative algorithm process uses the following formula:

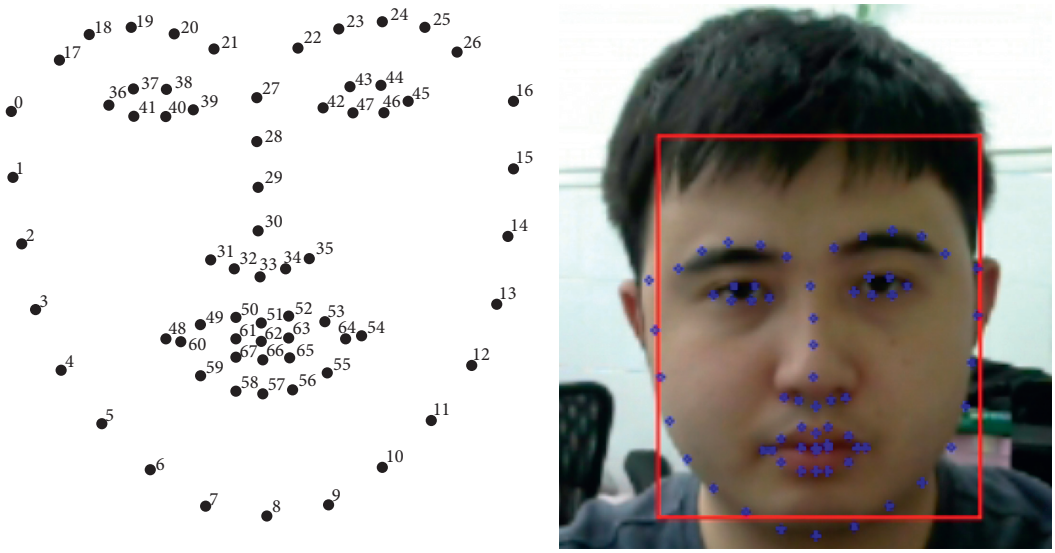
$$\widehat{S}^{(t+1)} = \widehat{S}^{(t)} + R_t\left(h\left(I, \widehat{S}^{(t)}\right)\right), \quad t = 1, \ldots, T, \tag{2}$$

where T is the number of rounds of the regression and $\widehat{S}^{(t)}$ is the current shape estimation; each regression $R_t(,)$ predicts an increment based on the input images I and $\widehat{S}^{(t)}$, that is, $R_t(h(I, \widehat{S}^{(t)}))$. The initial shape used is the average shape of the training data, and the update strategy is the Gradient Boosting Decision Tree (GBDT) algorithm [32]. Every time, for each separate subregion, we train a weak classifier whose predictive value approximates the true value of that subregion. Ultimately, the predicted value of the whole region is the weighted sum of every predicative value.

When the driver's face is detected, the feature points of the face are obtained in real time by the above algorithm, as shown in Figure 4(b).

2.2.2. Motion State Parameter Extraction. As discussed above, drivers get exhausted naturally during driving due to physiological and psychological state changes. At that time, they are in fatigue state. Fatigue driving endangers the driver and other traffic participants, as it declines the driving cognitive and driving skills, therefore resulting in misperception, misjudge, and misoperation. To ensure driving security and traffic safety, the driver must have a clear understanding of the driving condition and surrounding road environments all the time [35]. It requires the driver to continually adjust the head orientation and the fixation point of the eye. Compared to nonfatigue driving, the driver's visual field adjustment behaviors change significantly whether in the early, middle, or late stages of fatigue [36]. The facial motion state, such as movement amplitude and frequency, is abnormal.

Hence, we propose a Face Feature Triangle to characterize the driver's facial motion state. Based on face feature location, we defined a Face Feature Triangle (FFT). As shown in Figure 5, the midpoint of left eye is A, the midpoint of right eye is B, and the midpoint of mouth is C. The three points consist of the FFT. According to the FFT, we define the Face Feature Vector (FFV), as



(a) (b)

Figure 4: Driver's face feature point acquisition based on Dlib. (a) Dlib face feature point positioning. (b) Face feature point positioning effect.

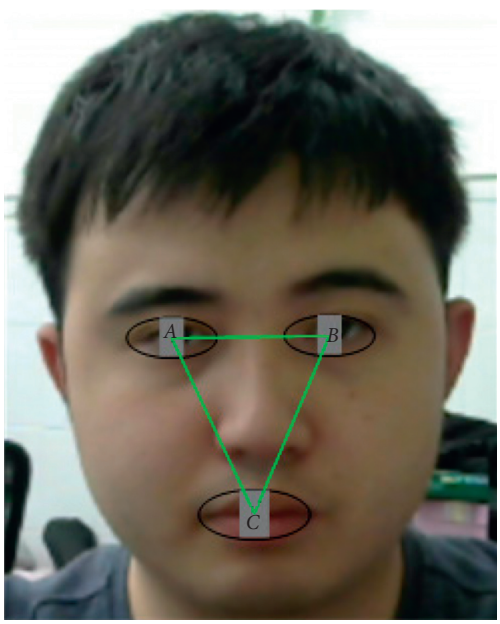


Figure 5: Face Feature Triangle (FFT).

$$\mathrm{FFV} = \left(F_x, F_y, \sqrt{S}\right), \tag{3}$$

where (F_x, F_y) is the midpoint of the FFT and S is the area of the FFT. According to the plane triangle's center of gravity and area formula, F_x, F_y, S are as shown in the following equation:

$$F_x = \frac{A_x + B_x + C_x}{3},$$
$$F_y = \frac{A_y + B_y + C_y}{3},$$
$$S = \frac{\left|A_x * B_y - B_x * A_y + B_x * C_y - C_x * B_y + C_x * A_y - A_x * C_y\right|}{2}. \tag{4}$$

Among them, according to Figure 4(a), Dlib face feature point positioning, and midpoint two-dimensional coordinate formula, the coordinates (A_x, A_y), (B_x, B_y), and (C_x, C_y) are defined as

$$\left(A_x, A_y\right) = \left(\frac{p36_x + p39_x}{2}, \frac{p36_y + p39_y}{2}\right),$$
$$\left(B_x, B_y\right) = \left(\frac{p42_x + p45_x}{2}, \frac{p42_y + p45_y}{2}\right), \tag{5}$$
$$\left(C_x, C_y\right) = \left(\frac{p60_x + p64_x}{2}, \frac{p60_y + p64_y}{2}\right),$$

where p36 is the coordinate of point 36 in Figure 4(a).

As is shown in Figure 6, FFT varies significantly with the driver's face position; therefore the FFV is suitable for characterizing the state of facial motion in the fatigue detection algorithm.

2.3. Driver's Facial Feature Points Collection. Generally, head posture-based fatigue detection algorithms [37] depend on the characteristics of instantaneous head motions such as nodding to determine whether the driver is in fatigue state. It is challenging to judge fatigue based on a single frame or a small number of frames and there may even be misjudgment. Therefore, it is necessary to study the statistical characteristics of the driver's facial movement state during fatigue. As described in Section 2.2, to extract the statistical characteristics of facial motion and find the relationship between statistical characteristics and driving fatigue state, we define FFT. Since the area of the FFT varies with the distance between driver's head and the camera, in order to get regularized data, we apply a face projection datum plane method. As shown in Figure 7, it projects all FFTs to a preset

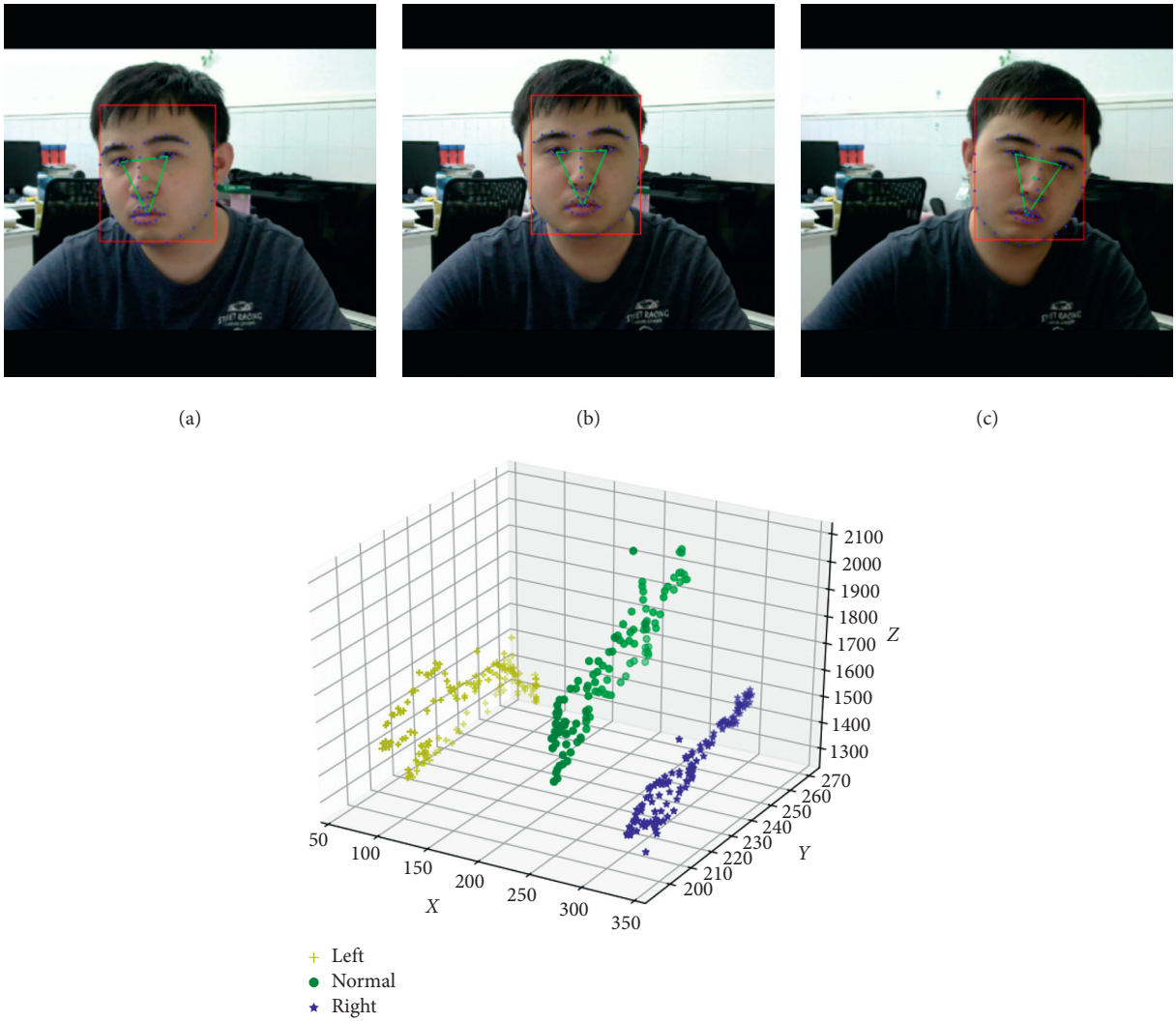


Figure 6: Different facial movement states and FFV differences, where X is F_x, Y is F_y, and Z is $\sqrt{S}$. "Left" stands for the left swing of the face, "Normal" stands for normal face posture, and "Right" stands for the right swing of the face.

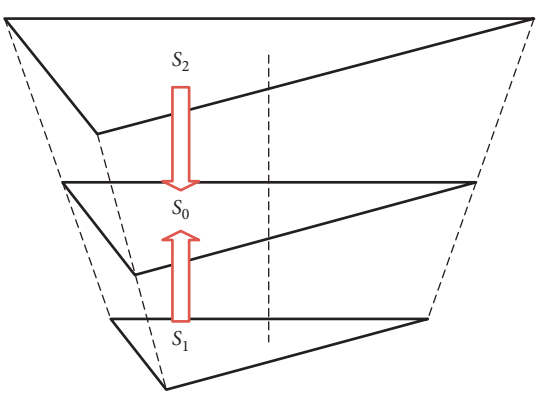


Figure 7: Projection schematic.

projection datum plane and eliminates the interference that originated from the distance difference. The area of the projection datum plane is S_0, and projection formula is shown in the following equation:

$$x = \left(F_x - \frac{\text{col}}{2}\right) * \sqrt{\frac{S}{S_0}} + \frac{\text{col}}{2},$$
$$y = \left(F_y - \frac{\text{row}}{2}\right) * \sqrt{\frac{S}{S_0}} + \frac{\text{row}}{2}, \quad (6)$$

where "row" and "col" are the numbers of rows and columns of the input images. A point (x, y) projected onto the datum projection plane is defined as a feature point of the driver's facial motion. We establish the feature point set of the driver's facial motion by counting the feature points in frames and then construct the statistical model of the driver's facial motion state. The experimental results are shown in Figure 8.

2.4. Driver Fatigue State Assessment Model Based on Facial Motion Information Entropy

2.4.1. Facial Motion Information Entropy. As mentioned above, in nonfatigue state, a driver is active to quickly switch

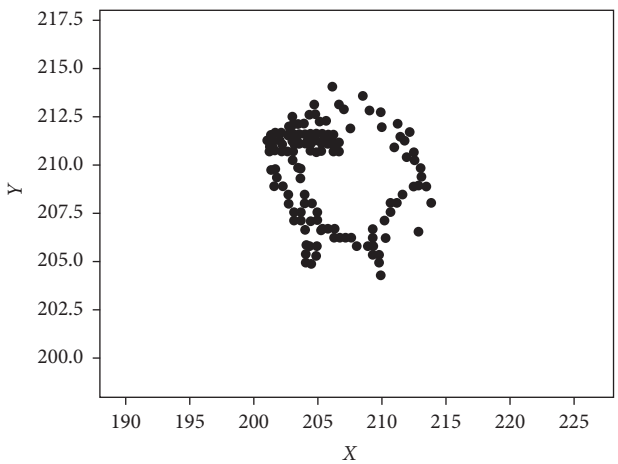


Figure 8: Facial motion feature point set.

the fixation point and head orientation, whereas in the opposite situation the drivers change their head position much more slowly.

To compare the difference between frequency and amplitude of the gaze point and the head orientation in the two driving states, based on the facial motion feature points, we count the set of facial motion feature points under a large number of consecutive frames. Figures 9(a) and 9(b) show the set of facial motion feature points under fatigue and nonfatigue conditions, respectively.

Accordingly, compared with the fatigued driving state, the nonfatigue facial motion feature points are more divergent and chaotic. "A Mathematical Theory of Communication" [38] pointed out that any information is redundant, and the redundancy is related to the probability or uncertainty of each symbol (number, letter, or word) in the message. That is information entropy, a concept from thermodynamics. It refers to the average amount of information after removing the redundant parts. The following equation shows the mathematical expression of information entropy:

$$H(X) = -\sum_{x\in\chi} p(X)\log p(X). \tag{7}$$

Based on the location of facial feature points in Section 2.2.1, we extract the FFV and establish the state analysis data set. Then, the facial motion information entropy is defined according to the concept of information entropy. Thus, the indicator to assess the degree of chaos of the facial feature point set is established. The calculation method is as follows:

(1) Calculate the center point $(\overline{F}_x, \overline{F}_y)$ of the facial motion feature point set, and N is the number of feature points, as is shown in

$$\overline{F}_x = \frac{\Sigma F_x}{N}, \qquad \overline{F}_y = \frac{\Sigma F_y}{N}. \tag{8}$$

(2) Calculate the Euclidean distance denoted as l_i from each feature point to the center point, where $i = 1, 2, \ldots, N$, as shown in

$$l_i = \sqrt{\left(F_x - \overline{F}_x\right)^2 + \left(F_y - \overline{F}_y\right)^2}. \tag{9}$$

(3) Calculate the mean value and standard deviation of distance, as is shown in the following equation

$$\mu_l = \frac{\sum_{i=1}^{N} l_i}{N}, \qquad \sigma_l = \sqrt{\frac{\sum_{i=1}^{N}\left(l_i - \mu_l\right)^2}{N}}. \tag{10}$$

(4) The interval I_i is defined as equation (11), where $i = 1, 2, \ldots, i_{\max}$. $i_{\max}$ is defined as equation (12):

$$I_i = \left[(i-1)*\frac{\mu_l}{\sigma_l}, i*\frac{\mu_l}{\sigma_l}\right], \tag{11}$$

$$i_{\max} = \frac{\max\left(l_1, l_2, \ldots, l_N\right)}{\mu_l/\sigma_l} + 1. \tag{12}$$

(5) According to the distance from each feature point to the center point, the number of distances falling in the interval I_i is counted as n_i.

(6) Calculate facial motion information entropy $H_F(X)$, as is shown in

$$H_F(X) = -\sum_{i=1}^{i_{\max}} p\left(x_i\right)\log p\left(x_i\right), p\left(x_i\right) = \frac{n_i}{N}. \tag{13}$$

2.4.2. Design of Driver's Facial Motion Information Entropy Classifier Based on SVM. As mentioned above, when drivers focus well on driving, they usually switch the fixation point and head orientation in order to get a better view of the driving environments, and the facial motion information entropy is higher. On the contrary, information entropy is much lower under fatigue driving situations. We use the training set in the open-source dataset YawDD (http://www.site.uottawa.ca/~shervin/yawning/) [39]. It contains fatigue driving data sets of all ages and people of all races, including different genders and facial features. It provides videos that record several common driving conditions such as driving with glasses, speaking, and singing while driving, even pretending to be simulating fatigue.

SVM [40] is a machine learning model that adopts the structural risk minimization criterion under the framework of statistical learning theory. It is a linear classifier model with the largest interval defined in the feature space. Given a training data set $S = \{(x_i, y_i), i = 1, 2, \ldots, N\}$ on a feature space, $x_i \in R^d$ is the ith input sample and $y_i \in \{+1, -1\}$ is the label corresponding to x_i. When $y_i = +1$, x_i is called a positive sample, and when $y_i = -1$, x_i is a negative sample.

Generally, a linear discriminant function $f(x) = w^T x_i + b$ in a d-dimensional space can distinguish two types of data, and a classification hyperplane can be described as

$$w^{*T} \cdot x + b^* = 0. \tag{14}$$

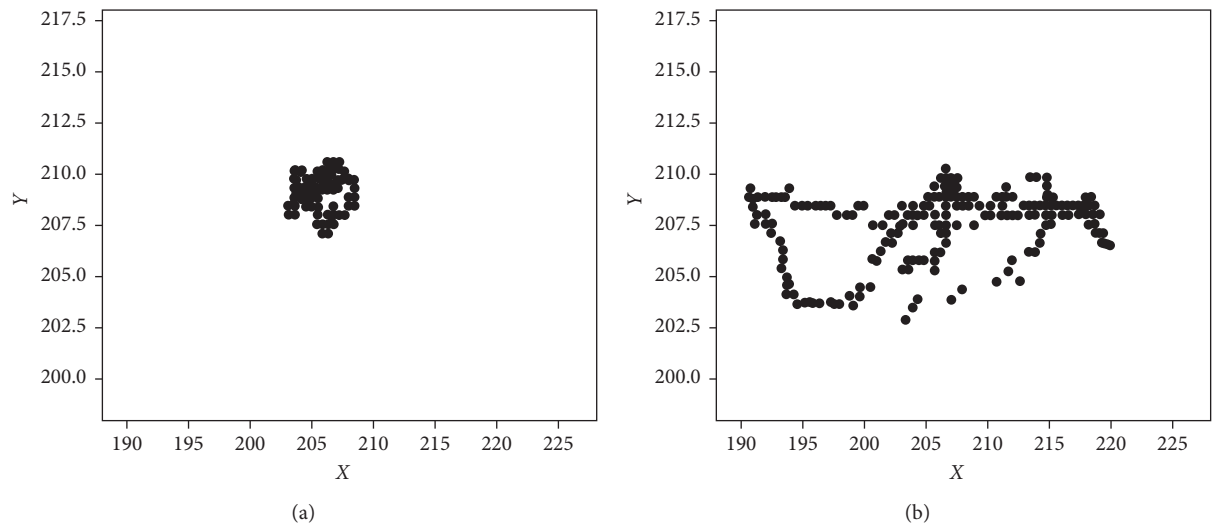


Figure 9: Different drive state facial motion feature point set. Facial motion feature point set in (a) fatigue and. (b)nonfatigue.

The normal vector w^T and the intercept b determine the superclass surface function. According to the basic idea of SVM, the constrained optimization problem of linear separable support vector machine can be obtained:

$$\begin{cases} \min\limits_{w,b} \quad J(w) = \dfrac{1}{2}\|w\|_2^2 \\ \\ \text{s.t.} \quad y_i\left(w^T \cdot x_i + b\right) \geq 1, i = 1, 2, \ldots, N. \end{cases} \tag{15}$$

In the training phase of the driver's face mark box, the improved YOLOv3-tiny is used as the training network, and the training set is applied to detect the driver's face. As described in Section 2.4.1, the driver's facial motion information entropy is calculated based on the positioning information of the Dlib face feature points. Among them, when $y_i = +1$, x_i is a positive sample, indicating that the driver is in nonfatigue driving state, and when $y_i = -1$, x_i is a negative sample, indicating that the driver is in fatigue driving state. Combined with the constraints of equation (15), the hyperplane parameters w^T and b can be calculated to obtain the driver's facial motion information entropy classifier.

Experiments show that the projection datum area S_0 has different values, which will affect the parameters w^T and b of the driver's facial motion information entropy classifier. In the experiment, S_0 is set to 10000.

2.4.3. Fatigue Judgment Based on Facial Motion Information Entropy. As mentioned above, the original image of the driver was acquired with an in-vehicle camera, and the improved YOLOv3-tiny network was used to detect the driver's face. The face area will be extracted as an input subimage, and then the Dlib toolkit is used to obtain the facial feature points of the subimage if the face is detected in a frame image. If not, the system will determine that the driver's head posture is abnormal. If it is determined that the driver's head posture is abnormal for more than 10 consecutive frames, the system will issue an alarm. Based on the face landmarks, the FFV is calculated according to the coordinates of the eye feature points and the mouth feature points. Within a certain number of frames (the number of frames set in this paper is more than 1000 frames), we count the FFV per frame. Considering that fatigue often generates during driving, if directly calculating the facial motion information entropy of all FFVs, the result may be inaccurate. In order to improve accuracy, as is shown in Figure 10, the paper sets a sliding window to calculate the facial motion information entropy in segments on all FFVs. The window size is set to 1000, and the sliding step size is set to 100. Each time the sliding window slides, the 1000 FFVs in the current sliding window are obtained first. Then, we can obtain the set of facial motion feature points in the current window. Finally, the facial motion information entropy $H_F(X)$ in the current window is calculated. Set $\mathrm{Th}_{H_F(X)}$ as the judgment threshold by training the SVM classifier on the YawDD training set. If $H_F(X) < \mathrm{Th}_{H_F(X)}$, the judgment is that the driver is in fatigue state. Otherwise, the sliding window moves to the next position to continue analyzing.

The flow chart of fatigue judgment based on facial motion information entropy is shown in Figure 11.

3. Results and Discussion

In order to verify the validity of the algorithm, we evaluated the performance of the improved YOLOv3-tiny network with the public data sets WIDER FACE and YawDD. On this basis, the design comparison experiment is carried out to verify whether the fatigue driving detection algorithm based on facial motion information entropy is correct.

3.1. Experimental Environment and Data Set. The experimental platform is the Intel Core i5-8400 with x86 architecture, and the CPU clock speed is 2.80 GHz. Graphics card is GTX1060 with Pascal architecture (CUDA: 9.2;

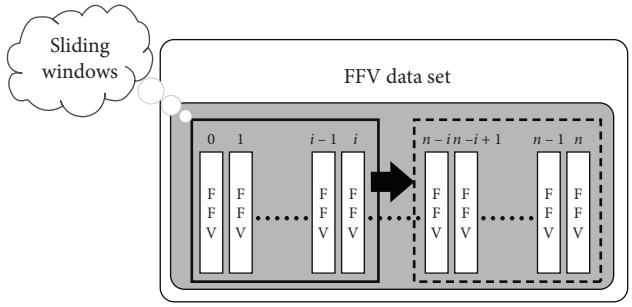


Figure 10: Sliding windows.

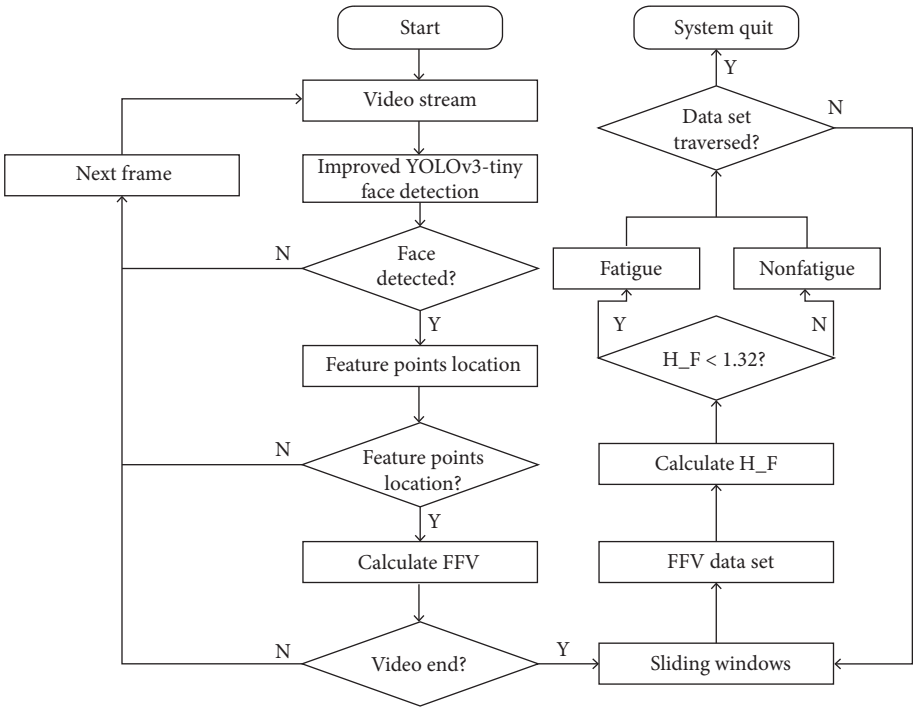


Figure 11: Driver fatigue state assessment model flow chart.

CUDNN: 7.2). The RAM is 8G DDR4, and the opencv3.4.6 image library is used. The deep learning computing framework is PaddlePaddle1.5. The environment of the program is python 3.6. Hardware configuration is shown in Table 1.

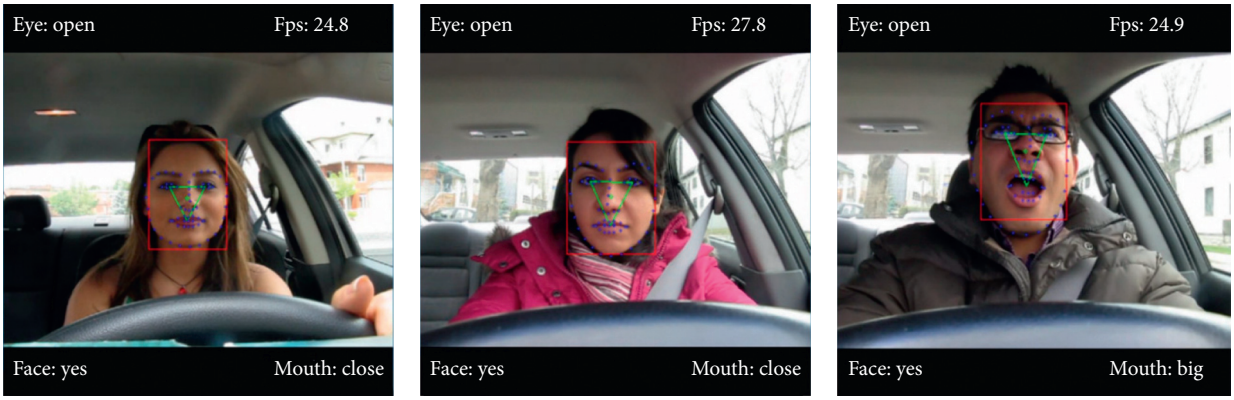
The data set used in the experiment included the public data sets WIDER FACE and YawDD, where the public data set WIDER FACE includes 32203 pictures and 393703 marked faces, which is used to train Yolov3-tiny's face network. However, the WIDER FACE data set only contains marker face images and does not provide any information about the driver's fatigue status. Therefore, the WIDER FACE data set cannot be used to analyze driver fatigue status. YawDD is a data set of fatigue driving detection including male and female volunteers in the naked eye, wearing glasses, normal state, speaking/singing, and simulated fatigue. So we choose YawDD data set as test set of fatigue driving detection. The detection result of the YawDD data set is shown in Figure 12.

Table 1: Hardware configuration table.

Type	Specific parameters
Processor	Intel(R)Core(TM)i5-8400 CPU@2.80 GHz 2.81 GHz
GPU	NVIDIA GeForce GTX1060 6 GB
Computer version	Windows 10
RAM	8.00 GB
Python version	3.6
Opencv version	3.4.6
Paddle version	1.5
CUDA version	9.2
CUDNN version	7.2

3.2. Face Detection and Feature Point Location

3.2.1. Qualitative Description. In order to verify the effectiveness of face detection based on the improved YOLOv3-tiny network and the accuracy based on the Dlib facial feature point location, the experiments were performed in the laboratory and in the vehicles.



(a) (b) (c)

Figure 12: The detect result of YawDD data set.

In the laboratory, the light is uniform and does not drastically change. The face recognition algorithm based on improved YOLOv3-tiny network can accurately detect faces from test videos. The face area can be correctly marked, as is shown in Figures 13(a) and 13(b) (1-1) and (1-2). Besides, the algorithm can detect the driver's face area and mark feature points, even in the cases of wearing glasses (as shown in Figure 13 (2-1)), head tilting (as shown in Figure 13 (1-3)), and expression changing (as shown in Figure 13 (2-2)).

In the vehicle experiment, the change of illumination may cause high interference to the driver's face detection and feature point location. So, it is crucial to verify the effectiveness of the algorithm in the real vehicle scenario. In the real driving scene, the algorithm can complete face detection and feature point location in case of uneven illumination, as is shown in Figure 13 (4-1). It can be seen that the algorithm has excellent recognition performance and robust performance in both the laboratory and real vehicle, and this will provide the basis for the driver's fatigue feature extraction and fatigue state assessment.

3.2.2. Quantitative Evaluation. The improved YOLOv3-tiny network provides face landmarks for fatigue driving detection. Its performance represents the effectiveness of the fatigue driving detection algorithm. Therefore, we quantitatively evaluate of the performance of the improved YOLOv3-tiny network on the WIDER FACE data set.

In this paper, we adopt the ROC curve [41] theory for evaluation. Accuracy is the ratio of the number of correctly predicted samples to the total number of samples, and it is an intuitive evaluation index of model performance. However, the accuracy rate is difficult to express the pros and cons of the model in case of uneven distribution of positive and negative sample data. The sensitivity indicates the proportion of all positive samples correctly detected. Specificity indicates the proportion of all negative samples correctly detected. The ROC curve is a comprehensive indicator formed by the combination of sensitivity and specificity and reflects the sensitivity and specificity of continuous variables.

(1) Accuracy (ACR). In the task of the driver's face detection, the ACR is the ratio of the number of correctly detected images to the total number of images:

$$\mathrm{ACR} = \frac{N_{\mathrm{detected}}}{N_{\mathrm{total}}}, \tag{16}$$

where N_{detected} is the number of correctly detected images and N_{total} is the total number of images.

In the process of improving the YOLOv3-tiny network training and verification, the intersection ratio parameter (IOU) [42] is introduced to measure the similarity between the face detection area and the marked real area. IOU is a standard for measuring the accuracy of a corresponding object in a specific data set. In Figure 14, face_d is the face area detected by the model, face is the real area marked, and the calculation formula is given in the following equation (17): where Area(face_d∩face) is the area of face_d∩face and Area(face_d∪face) is the area of face_d∪face.

$$\mathrm{IoU} = \frac{\mathrm{Area}\,(\mathrm{face\_}d \cap \mathrm{face})}{\mathrm{Area}\,(\mathrm{face\_}d \cup \mathrm{face})}, \tag{17}$$

The intersection ratio indicates the degree of overlap between the model prediction area and the real area. As can be seen from Figure 14, the higher the value is, the higher the detection accuracy is. In the case where IOU = 1, the prediction box overlaps with the real box. Generally speaking, the object is correctly detected when the IOU is more than 0.5. In the face detection process, we adopt a higher threshold. In this paper, when the IOU is more than 0.75, the face is considered to be correctly detected. Figure 15 shows the accuracy curve of the driver's face detection during the training of the improved YOLOv3-tiny network. It can be seen that, with the increase of training rounds, the accuracy of face detection gradually increases. The improved YOLOv3-tiny network has an accuracy rate of 98.5%.

(2) ROC Curve. Sensitivity and specificity are important evaluation indicators of the pattern recognition model. If

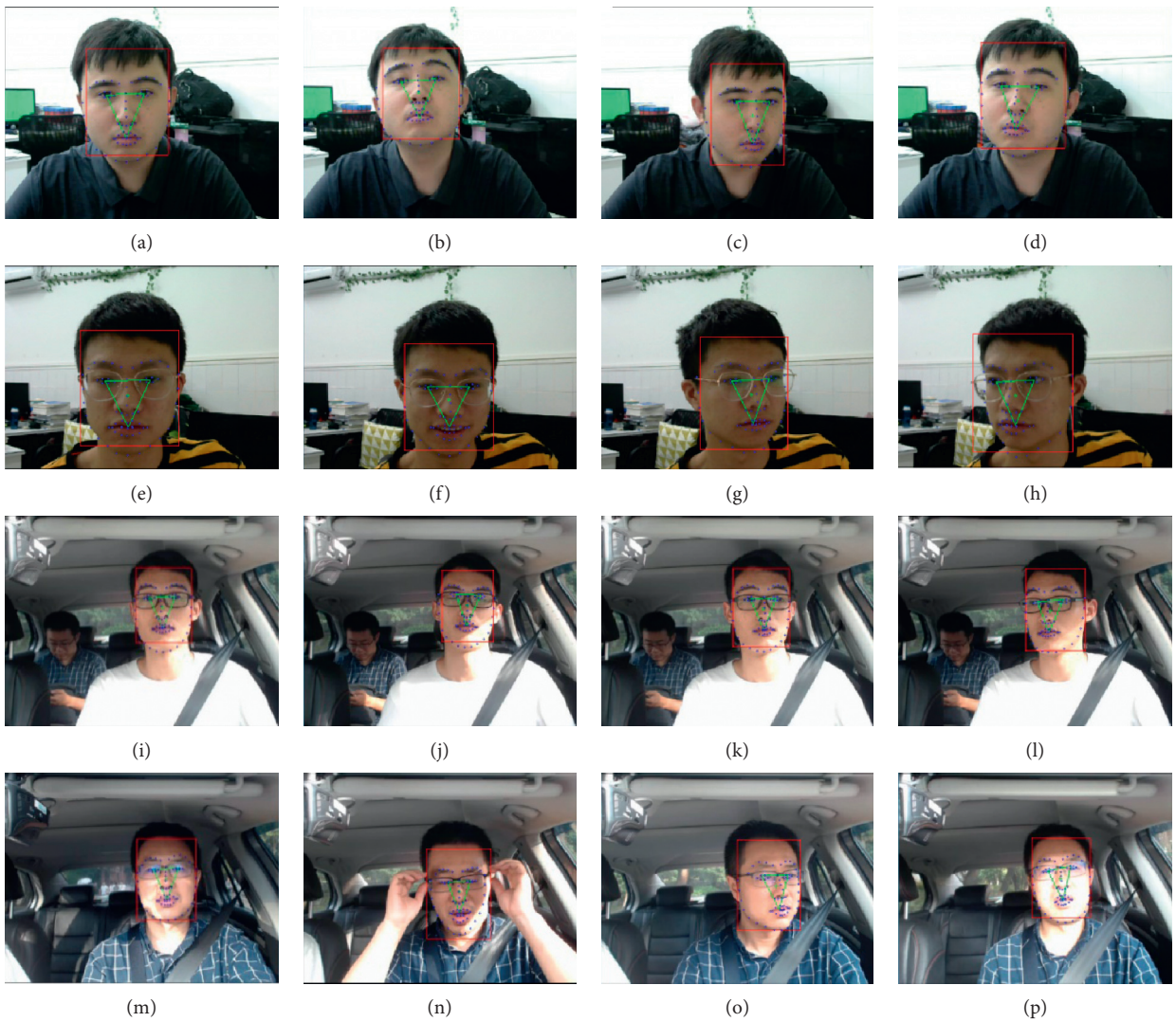


Figure 13: The results of face detection and feature point location. (a) (1-1). (b) (1-2). (c) (1-3). (d) (1-4). (e) (2-1). (f) (2-2). (g) (2-3). (h) (2-4). (i) (3-1). (j) (3-2). (k) (3-3). (l) (3-4). (m) (4-1). (n) (4-2). (o) (4-3). (p) (4-4).

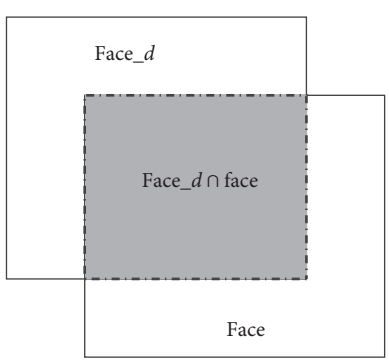


Figure 14: Intersection over union.

you use TP, TN, FP, and FN to indicate the number of true-positive, true-negative, false-positive, and false-negative samples, respectively, in a test, then the definitions of sensitivity S_n and specificity S_p are

$$S_n = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \quad S_p = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}}. \tag{18}$$

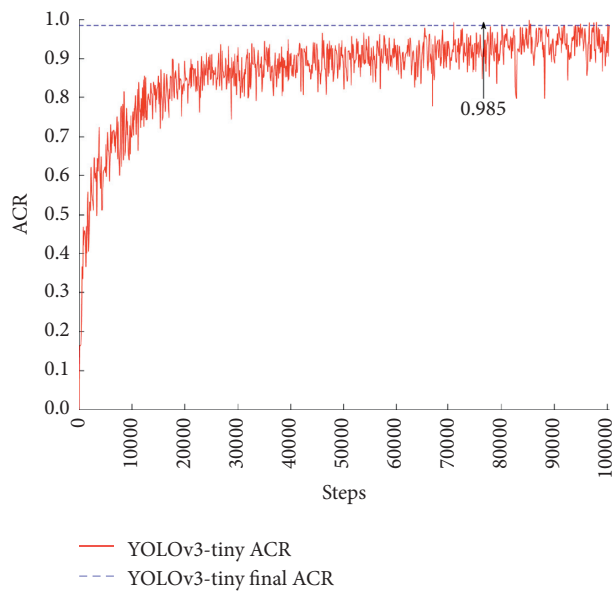


Figure 15: Driver face detection accuracy.

A ROC curve is a graph of the relationship between the true-positive rate (sensitivity) and the false-positive rate (1 – specificity). The ROC curve is one of the comprehensive indicators for characterizing the accuracy of pattern recognition tasks, and the closer the ROC curve is to the upper left corner, the better the model performance is.

Figure 16 shows the ROC curve of the driver's face detection model. As can be seen from the figure, the ROC curve corresponding to the improved YOLOv3-tiny network is close to the upper left corner of the graph, indicating high accuracy in face detection.

In summary, by evaluating the performance of the improved YOLOv3-tiny network on the WIDER FACE data set, it is shown that the improved YOLOv3-tiny network in this paper has high accuracy. Besides, the ROC curve indicates that the algorithm can effectively avoid two types of errors in the driver's face recognition, that is, to ensure that the driver's face can be correctly detected while avoiding the misjudgment on the face.

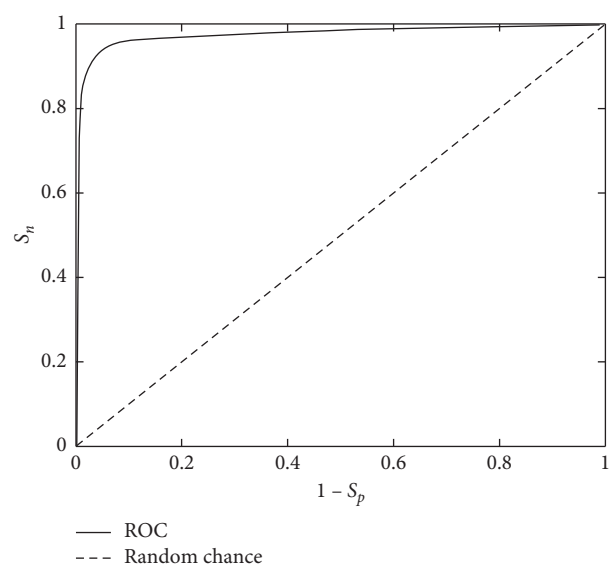


Figure 16: ROC curve.

3.3. Fatigue State Evaluation

3.3.1. Accuracy. We use the YawDD data set to test the performance of fatigue detection. Face detection and facial feature point location are the basis of fatigue driving detection. The FFV of each frame in the on-board video is calculated and stored based on the facial feature points. Calculate the FFVs of all video frames in a certain period, and establish a state analysis data set. The sliding window (discussed in Section 2.4.3) is applied to the state analysis data set to calculate the facial motion information entropy for each sliding. If the entropy does not exceed the threshold, we can conclude that the driver is in fatigue state. Videos are randomly selected from the data set for fatigue driving detection. The process of fatigue driving detection is shown in Figure 11.

In this paper, we randomly select ten videos from the YawDD test set, including nonfatigue driving status and fatigue driving status. The facial information entropy threshold for judging fatigue state is 1.32, and the results are shown in Table 2. It can be seen that the accuracy of the fatigue driving detection in the randomly selected ten videos is 90%, and the correct rate of the system in the entire test set of YawDD is 94.32%.

3.3.2. Speed. Based on hardware configuration as shown in Table 1, a comparison test is performed on the image source to verify the real-time performance of the system. The results are shown in Table 3.

Table 3 illustrates that YawDD Video excels at face detection time. One possible reason is the difference between

Table 2: Sample fatigue test table.

Sample number	Facial motion information entropy	Actual driving status	Predictive driving status
1	[1.23, 0.96, 0.56, 1.20, 1.40, 0.49, 0.65, 0.45, 0.75]	Fatigue	Fatigue
2	[1.10, 1.42, 0.86, 0.52, 0.97, 0.95, 1.50, 0.88]	Fatigue	Fatigue
3	[2.50, 2.42, 2.65, 1.93, 2.01, 2.89, 3.32, 3.21]	Nonfatigue	Nonfatigue
4	[0.57, 0.87, 0.34, 0.67, 0.95, 1.12, 1.21, 1.29, 1.01]	Fatigue	Fatigue
5	[1.98, 1.87, 1.93, 2.03, 3.23, 3.42, 3.34, 2.72]	Nonfatigue	Nonfatigue
6	[0.62, 0.57, 0.88, 1.02, 1.42, 1.45, 0.92]	Fatigue	Fatigue
7	[2.22, 1.52, 2.33, 2, 78, 3.11, 2.07, 2.98, 3.04]	Nonfatigue	Nonfatigue
8	[1.35, 1.02, 1.22, 0.78, 0.56, 0.22, 0.24, 0.31, 0.55]	Fatigue	Fatigue
9	[2.44, 2.57, 2.72, 1.98, 1.42, 1.30, 2.23, 2.89, 2.66]	Nonfatigue	Fatigue
10	[1.50, 0.89, 0.76, 0.71, 0.65, 0.88, 0.31, 0.42, 0.51]	Fatigue	Fatigue

Table 3: The time spent in fatigue status judgment.

Image source	Face detection time (ms)	Facial feature point positioning time (ms)	Calculate FFV time (ms)	Total time (ms)
Camera	34.52	13.91	1	49.43
YawDD Video	32.13	13.91	1	47.04

Table 4: Comparison of fatigue detection algorithms.

Algorithms	Accuracy (%)	Speed ($ms \cdot f^{-1}$)
AdaBoost + CNN	92.10	58.61
CNN + DF_LSTM	91.48	65.64
Algorithm in this paper	94.32	49.43

the data reading methods, and the YawDD Video method gets the data from the video stream directly.

Our algorithm shows that the system has good accuracy and high-speed performance under various conditions and can accurately judge the fatigue state of the driver. Compared with AdaBoost + CNN and CNN + DF_LSTM algorithms [43, 44], our method improves the accuracy of the fatigue driving detection algorithm. It also has better real-time performance, which meets the requirements of the fatigue driving detection system. The comparative result is shown in Table 4.

4. Conclusions and Future Directions

With the rapid increase of global car ownership, road traffic accidents have become one of the leading causes of human death in the world. Fatigue driving is one of the main causes of road traffic accidents. Fatigue driving can seriously affect driving skills and seriously threaten drivers and other traffic participants. At present, fatigue driving detection and early warning have achieved better research results, but they still need some improvements, such as high intrusiveness, poor detection performance in complex environments, and simple evaluation indicator. Therefore, we propose a new detection algorithm for fatigue driving based on facial motion information entropy. The main contributions are as follows.

(i) We design a driver's face detection architecture based on the improved YOLOv3-tiny convolutional neural network and train the network with the open-source data set WIDER FACE. Compared with other deep learning algorithms, such as YOLOv3 [17] and MTCNN [18], the algorithm based on the improved YOLOv3-tiny network improves the face recognition accuracy, simplifies the network structure, and reduces the amount of calculation. Then, it is more convenient to transplant to the mobile. The accuracy rate of face recognition based on the improved YOLOv3-tiny network is up to 98.5%, and single test just takes 34.52 ms.

(ii) The Dlib toolkit is used to extract facial feature points on the face area that is located by the improved YOLOv3-tiny convolutional neural network. Then, the driver's FFT is established by analyzing the positioning characteristics of the eye and mouth. Finally, the driver's FFV is constructed by the area and centroid of FFT. We calculate the FFV of each frame and write it to the database. Thereby a state analysis data set is established. In many research studies, the basis for assessing the state of the driver is the recognition result of a single frame or few frames, which reduce the accuracy of fatigue driving detection. In this paper, based on the analysis results of a large number of consecutive frames, we design sliding windows of driving fatigue analysis to obtain the statistical characteristics of the facial motion state. Therefore, the process of driver fatigue can be observed.

(iii) To eliminate the interference of change of the FFT's area to fatigue driving judgment, we introduce the face projection datum plane and apply the projection principle to extract the motion feature points of the face. Then, based on the motion feature points, we propose the facial motion information entropy, which quantitatively characterizes the chaotic degree of the motion feature points of the face. Then, we train the SVM classifier using the open-source data set YawDD [37]. Experiments show that the

projection datum area S_0 has different values, which will affect the parameters w^T and b of the driver's facial motion information entropy classifier. We design fatigue judgment algorithm based on facial motion information entropy, and the comparison experiments show that our algorithm has an accuracy rate of 94.32% and an algorithm speed of 49.43 ms/f, which further improve the accuracy and speed of the driver's fatigue detection algorithm.

In the future, we will focus on the following research:

(1) Upload the results of the fatigue detection to the cloud platform and combine the big data analysis techniques to analyze the driver's fatigue period [45]

(2) Integrate the fatigue driving detection algorithm into ADAS (Advanced Driving Assistant System) [46, 47]

(3) Expand the applicable environment of the algorithm and explore the driver fatigue detection algorithm based on facial motion information entropy in night environment [48, 49]

Data Availability

The data used to support the findings of this study are available from the first author and the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (Grant no. 51808151), Guangdong Provincial Public Welfare Research and Capacity Building Special Project (Grant no. 2016A020223002), South China University of Technology Central University Fund Project (Grant no. 2017ZD034), Guangdong Provincial Science and Technology Plan Project (Grant no. 2017A040405021), the Fundamental Research Funds for Guangdong Communication Polytechnic (Grant no. 20181014), Guangdong Provincial Natural Science Foundation (Grant no. 2020A151501842), Guangzhou 2020 R&D Plan for Key Areas (Grant no. 202007050004), and by State Key Lab of Subtropical Building Science, South China University of Technology (Grant no. 2020ZB20).

References

[1] A. Amodio, M. Ermidoro, D. Maggi, S. Formentin, and S. M. Savaresi, "Automatic detection of driver impairment based on pupillary light reflex," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 8, pp. 3038–3048, 2019.

[2] X. Li, X. Lian, and F. Liu, "Rear-end road crash characteristics analysis based on Chinese in-depth crash study data," in *Proceedings of the 16th COTA International Conference of Transportation Professionals: Green and Multimodal Transportation and Logistics*, pp. 1536–1545, Shanghai, China, July 2016.

[3] F. Chen and S. Chen, "Injury severities of truck drivers in single- and multi-vehicle accidents on rural highways," *Accident Analysis & Prevention*, vol. 43, no. 5, pp. 1677–1688, 2011.

[4] X. Zhu, Z. Dai, F. Chen, X. Pan, and M. Xu, "Using the visual intervention influence of pavement markings for rutting mitigation-part I: preliminary experiments and field tests," *International Journal of Pavement Engineering*, vol. 20, no. 5, pp. 734–746, 2019.

[5] R. Zhang, F. You, X. N. Chu, L. Guo, Z.-C. He, and R.-B. Wang, "Lane change merging control method for unmanned vehicle under V2V cooperative environment," *China Journal of Highway and Transport*, vol. 31, pp. 180–191, 2018.

[6] Y. Wang, X. Liu, Y. Zhang, Z. Zhu, D. Liu, and J. Sun, "Driving fatigue detection based on EEG signal," in *Proceedings of the 5th International Conference on Instrumentation and Measurement, Computer, Communication, and Control*, pp. 715–718, Qinhuangdao, China, September 2015.

[7] R. Bhardwaj, P. Natrajan, and V. Balasubramanian, "Study to determine the effectiveness of deep learning classifiers for ECG based driver fatigue classification," in *Proceedings of the 13th International Conference on Industrial and Information Systems*, pp. 98–102, Punjab, India, December 2018.

[8] M. K. Sharma and M. M. Bundele, "Design & analysis of *k*-means algorithm for cognitive fatigue detection in vehicular driver using oximetry pulse signal," in *Proceedings of the IEEE International Conference on Computer, Communication and Control (IC4)*, Indore, India, September 2015.

[9] L. Boon-Leng, L. Dae-Seok, and L. Boon-Giin, "Mobile-based wearable-type of driver fatigue detection by GSR and EMG," in *Proceedings of the TENCON 2015-2015 IEEE Region 10 Conference*, Macau, China, November 2015.

[10] J. Yan, H. Kuo, Y. Lin, and T. Liao, "Real-time driver drowsiness detection system based on PERCLOS and grayscale image processing," in *Proceedings of the 2016 International Symposium on Computer, Consumer and Control (IS3C)*, pp. 243–246, Xi'an, China, July 2016.

[11] G. Niu and C. Wang, "Driver fatigue features extraction," *Mathematical Problems in Engineering*, vol. 2014, Article ID 860517, 10 pages, 2014.

[12] L. M. Bergasa and J. Nuevo, "Real-time system for monitoring driver vigilance," in *Proceedings of the IEEE International Symposium on Industrial Electronics, 2005. ISIE 2005*, pp. 1303–1308, Dubrovnik, Croatia, June 2005.

[13] F. You, Y.-h. Li, L. Huang, K. Chen, R.-h. Zhang, and J.-m. Xu, "Monitoring drivers' sleepy status at night based on machine vision," *Multimedia Tools and Applications*, vol. 76, no. 13, pp. 14869–14886, 2017.

[14] R.-H. Zhang, Z.-C. He, H.-W. Wang, F. You, and K.-N. Li, "Study on self-tuning tyre friction control for developing main-servo loop integrated chassis control system," *IEEE Access*, vol. 5, pp. 6649–6660, 2017.

[15] F. Chen, M. Song, and X. Ma, "Investigation on the injury severity of drivers in rear-end collisions between cars using a random parameters bivariate ordered probit model," *International Journal of Environmental Research and Public Health*, vol. 16, no. 14, p. 2632, 2019.

[16] S. Yang, P. Luo, C. C. Loy, and X. Tang, "Wider face: a face detection benchmark," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5525–5533, IEEE Computer Society, Las Vegas, NV, USA, June 2016.

[17] S. Luo, C. Xu, and H. Li, "An application of object detection based on YOLOv3 in traffic," in *Proceedings of the 2019 International Conference on Image, Video and Signal Processing - IVSP 2019*, pp. 68–72, Association for Computing Machinery, Shanghai, China, 2019.

[18] X. Chen, X. Luo, X. Liu, and J. Fang, "Eyes localization algorithm based on prior MTCNN face detection," in *Proceedings of the 2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC)*, pp. 1763–1767, Chongqing, China, May 2019.

[19] D. Sommer and M. Golz, "Evaluation of PERCLOS based current fatigue monitoring technologies," in *Proceedings of the 2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*, pp. 4456–4459, Buenos Aires, Argentina, August 2010.

[20] X. Sun, H. Zhang, W. Meng, R. Zhang, K. Li, and T. Peng, "Primary resonance analysis and vibration suppression for the harmonically excited nonlinear suspension system using a pair of symmetric viscoelastic buffers," *Nonlinear Dynamics*, vol. 94, no. 2, pp. 1243–1265, 2018.

[21] G. Wu, F. Chen, X. Pan, M. Xu, and X. Zhu, "Using the visual intervention influence of pavement markings for rutting mitigation–part I: preliminary experiments and field tests," *International Journal of Pavement Engineering*, vol. 20, no. 6, pp. 734–746, 2019.

[22] P. Viola and M. Jones, "Robust real-time face detection," in *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, vol. 2, p. 747, 2001.

[23] K. Luu, C. Zhu, C. Bhagavatula, T. H. N. Le, and M. Savvides, "A Deep learning approach to joint face detection and segmentation," in *Advances in Face Detection and Facial Image Analysis*, pp. 1–12, Springer International Publishing, Cham, Switzerland, 2016.

[24] J. Xiang and G. Zhu, "Joint face detection and facial expression recognition with MTCNN," in *Proceedings of the 4th International Conference on Information Science and Control Engineering*, pp. 424–427, Institute of Electrical and Electronics Engineers Inc., Hunan, China, July 2017.

[25] W. Shi, J. Li, and Y. Yang, "Face fatigue detection method based on MTCNN and machine vision," *Advances in Intelligent Systems and Computing*, Springer Verlag, vol. 1017, pp. 233–240, Huainan, China, 2020.

[26] S. Zhao, H. Song, W. Cong, Q. Qi, and H. Tian, "End-to-end cascade cnn for simultaneously face detection and alignment," in *Proceedings of the 2017 International Conference on Virtual Reality and Visualization (ICVRV)*, pp. 35–40, Institute of Electrical and Electronics Engineers Inc., 2017, Zhengzhou, China.

[27] M. El-Arabawy, S. Zaki, and F. Harby, "Improved AdaBoost algorithm for face detection," in *Proceedings of the 2010 International Conference on Image Processing, Computer Vision, and Pattern Recognition*, vol. 1, CSREA Press, Las Vegas, NV, USA, pp. 353–358, 2010.

[28] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: unified, real-time object detection," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 779–788, IEEE Computer Society, Las Vegas, NV, USA, 2016.

[29] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[30] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[31] K. D. E. Dlib-ml, "A machine learning toolkit," *Journal of Machine Learning Research*, vol. 10, pp. 1755–1758, 2009.

[32] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," *The Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 2001.

[33] X. Cao, Y. Wei, F. Wen, and J. Sun, "Face alignment by explicit shape regression," *International Journal of Computer Vision*, vol. 107, no. 2, pp. 177–190, 2014.

[34] P. Dollar, P. Welinder, and P. Perona, "Cascaded pose regression," in *Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2010*, pp. 1078–1085, IEEE Computer Society, San Francisco, CA, USA, June 2010.

[35] H. Wang, F. You, X. Chu, X. Li, and X. Sun, "Research on customer marketing acceptance for future automatic driving-a case study in China city," *IEEE Access*, vol. 7, pp. 20938–20949, 2019.

[36] L. Jiang, H. Wang, S. Gao, and S. Jiang, "Research of the automotive driver fatigue driving early warning system," *Communications in Computer and Information Science*, Springer-Verlag, Berlin, Germany, pp. 383–391, 2011.

[37] G. Sun, Y. Jin, Z. Li, F. Zhang, and L. Jia, "A vision-based head status judging algorithm for driving fatigue detection system," *Advances in Transportation Studies*, vol. 2015, pp. 51–64, 2015.

[38] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, 1948.

[39] S. Abtahi, M. Omidyeganeh, S. Shirmohammadi, and B. Hariri, "YawDD," in *Proceedings of the 5th ACM Multimedia Systems Conference, MMSys 2014*, pp. 24–28, Association for Computing Machinery, Singapore, March 2014.

[40] Z. You, Y. Gao, J. Zhang, H. Zhang, M. Zhou, and C. Wu, "A study on driver fatigue recognition based on SVM method," in *Proceedings of the 4th International Conference on Transportation Information and Safety, ICTIS 2017*, pp. 693–697, Institute of Electrical and Electronics Engineers Inc., Banff, Canada, August 2017.

[41] J. Hernández-Orallo, "ROC curves for regression," *Pattern Recognition*, vol. 46, no. 12, pp. 3395–3411, 2013.

[42] L. Tychsen-Smith and L. Petersson, "Improving object localization with fitness NMS and bounded IoU loss," in *Proceedings of the 31st Meeting of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2018*, pp. 6877–6885, IEEE Computer Society, Salt Lake City, UT, USA, June 2018.

[43] G. Lei, X. Liang, Z. Xiao, and Y. Li, "Real-time driver fatigue detection based on morphology infrared features and deep learning," *Infrared & Laser Engineering*, vol. 47, no. 2, Article ID 203009, 2018.

[44] J. M. Guo and M. Herleeyandi, "Driver drowsiness detection using hybrid convolutional neural network and long short-term memory," *Multimedia Tools & Applications*, vol. 78, no. 20, pp. 29059–29087, 2019.

[45] C. Xu, Y. Yang, S. Jin, Z. Qu, and L. Hou, "Potential risk and its influencing factors for separated bicycle paths," *Accident Analysis & Prevention*, vol. 87, pp. 59–67, 2016.

[46] F. Chen, H. Peng, X. Ma, J. Liang, W. Hao, and X. Pan, "Examining the safety of trucks under crosswind at bridge-tunnel section: a driving simulator study," *Tunnelling and Underground Space Technology*, vol. 92, Article ID 103034, 2019.

[47] H. Xiong, X. Zhu, and R. Zhang, "Energy recovery strategy numerical simulation for dual axle drive pure electric vehicle based on motor loss model and big data calculation," *Complexity*, vol. 2018, Article ID 4071743, 14 pages, 2018.

[48] X. Qu, M. Zhou, Y. Yu, C. T. Lin, and X. Wang, "Jointly dampening traffic oscillations and improving energy consumption with electric, connected and automated vehicles: a reinforcement learning based approach," *Applied Energy*, vol. 257, Article ID 114030, 2019.

[49] M. Zhou, Y. Yu, and X. Qu, "Development of an efficient driving strategy for connected and automated vehicles at signalized intersections: a reinforcement learning approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 1, pp. 433–443, 2019.

Hindawi
Journal of Advanced Transportation
Volume 2020, Article ID 8838896, 8 pages
https://doi.org/10.1155/2020/8838896

Research Article

Design of Real-Time Dynamic Reversible Lane in Intelligent Cooperative Vehicle Infrastructure System

Lina Mao,[1] Wenquan Li,[1] Pengsen Hu,[2] Guiliang Zhou,[3] Huiting Zhang,[3] and Jin Dai[3]

[1]*School of Transportation, Southeast University, Nanjing 210096, China*
[2]*Department of Civil and Environmental Engineering, The Pensylvania State University, University Park, State College, PA 16802, USA*
[3]*Jiangsu Key Laboratory of Traffic and Transportation Security, Huaiyin Institute of Technology, Huaian 223003, China*

Correspondence should be addressed to Wenquan Li; seu_liwenquan@163.com and Guiliang Zhou; zglpaper@qq.com

Received 20 March 2020; Revised 5 May 2020; Accepted 11 May 2020; Published 28 May 2020

Academic Editor: Dongfang Ma

Copyright © 2020 Lina Mao 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.

The rapidly growing traffic demand and the slowly increasing traffic supply have produced an mounting contradiction, which is mainly manifested in cities as road congestion and unbalanced bidirectional traffic flow. Most of the reversible lanes are implemented on fixed sections and fixed times and are mainly guided by ground markings, road signs, railings, and traffic police officer. It requires a lot of human and material costs. And, the control effect is lagging and inaccurate. Aiming at these problems, a real-time dynamic reversible lane scheme in the Intelligent Cooperative Vehicle Infrastructure System (CVIS) was proposed. Traffic information was collected in real time through the CVIS, and a reversible lane scheme was established based on the real-time service level V/C and BRP functions. A lane change control model was applied to determine the number of lanes and the timing of lane changes. Then, the reversible lanes were managed in real time through intelligent road stud lights and light curtain walls. Buffer sections and no-entry sections were set to ensure reversible lanes operating safely and efficiently. VISSIM simulation was used for case analysis, and the results showed that compared with the traditional time-controlled reversible lane scheme, the real-time dynamic reversible lane scheme could reduce the average vehicle delay by 27.4% and decrease the vehicle VOC, CO and NO_X emissions by 13.5%.

1. Introduction

By the end of 2018, China's car ownership had reached 240 million units, an increase of 22.85 million units over 2017. The national highway mileage reached 4,486,500 km, an increase of 73,100 km and 1.53% over the previous year. The rapidly increasing transportation demand and the slowly increasing transportation supply have produced increasing contradictions between supply and demand [1, 2]. Some urban roads have serious congestion and unbalanced bidirectional traffic flow. In 2017, the number of connected vehicles in China reached 17.8 million. With the integration and innovation of big data, cloud computing, and other technologies, the penetration rate of connected vehicles will continue to accelerate in the future [3–6].

Intelligent road stud light is currently mainly used for underground parking lot vehicle indication, detection, and lane keeping reminder and has not been used as a signal indicating device on the road. In the proposed reversible lane scheme, intelligent solar road stud lights were used, which were controlled in real time under connected vehicle environment, to achieve the same indication function as road traffic lights.

A huge amount of study has been carried out on reversible lanes. Brian Wolshon and Laurence Lambert studied the application status, control and management measures, and evaluation methods of reversible lanes, and found that although there was no unified planning and well-established standards, the introducing of reversible lanes still achieved the expected goals and were accepted by the public [7]; Matthew and Peter et al. established a linear programming

model to calculate the optimal configuration of variable lanes and used the information collected by traffic sensors to determine the direction of reversible lanes on the road. The direction was dynamically changed, and the results showed that the road capacity can be effectively improved by introducing the dynamic reversible lane scheme [8]; Golub et al. put forward suggestions from the perspective of economic benefits and traffic benefits from the design of reversible lanes, marking, and management [9]; Yu and Tian studied the setting conditions of the reversible lanes and established a two-level planning model based on the road network. The upper layer of the model integrates and optimizes the lane allocation, and the lower layer is a network equilibrium model for predicting the driver's response to lane allocation [10]; Cui and Liu investigated the possibility and necessity of introducing reversible lane schemes based on field survey and put forward solutions to manufacturing-related hidden traffic hazards of the reversible lane schemes [11]; Sun established an optimization model for the allocation of reversible lanes, dividing 24 hours of a day into several continuous time sections, which were used to allocate the number of tidal lanes in different sections according to the traffic flow characteristics of different time sections [12–14];

Dai et al. analyzed the current status of reversible lane schemes in foreign countries, proposed three steps for setting reversible lane schemes, and verified the effect of introducing reversible lane schemes through simulation [15, 16].

Reversible lanes have been studied by scholars all over the world, and they have been widely used in urban roads, but most of them are fixed time and fixed lanes, which require manual operation depending on the traffic situation. At present, no intelligent lane-combined technology with road studs has been achieved for the dynamic control of reversible lanes [17–20]. Where road information in CVIS was collected, the control model was used to allocate reversible lanes and intelligent road studs were used to guide traffic [21, 22]. Compared with the existing reversible lane control methods, the biggest advantage is that reversible lane allocation can be dynamically changed according to the traffic volume of the road, no manual operation is needed, and energy conservation and environmental protection are achieved, thereby achieving optimal use of road resources.

2. Design Principles and Schemes of Real-Time Dynamic Reversible Lanes

2.1. Reversible Lane Scheme Implementation Conditions. According to the reversible lane implementing conditions recommended by the American Society of Transportation Engineers, the road condition, road capacity, and traffic volume are generally considered. In this paper, the road conditions and traffic conditions for opening the reversible lane schemes were investigated.

2.1.1. Road Conditions

① Lane setting conditions: for the introduce of variable lanes, the number of original road lanes must be at least three in both directions. In cities with large traffic flow, the number of lanes on the road should be greater than 6 or no less than 5 to ensure the room for reversible lanes.

② Traffic facility conditions: city roads with reversible lanes generally should not equip immovable facilities such as central separation zone or trolley tracks.

2.1.2. Traffic Conditions

① Traffic directions: stable traffic flow is the primary condition to ensure the implementation of reversible lane scheme, and the traffic flow with obvious traffic imbalance phenomenon is the premise to set the reversible lane. It is required that the directional distribution coefficient below 2/3, so as to ensure the benefits of implementing of the variable lane.

② Traffic capacity: after the reversible lane is introduced, the capacity of the road should still meet the original traffic demand.

2.2. Dynamic Lane Changing Criteria

2.2.1. Determination of Traffic Conditions. In CVIS, wireless sensors are used to obtain real-time traffic volume information V and the traffic capacity C which is used to calculate the service level V/C in both directions of the road. The traffic conditions are divided into five levels according to its service level for 5 states as shown in Table 1.

2.2.2. Determination of the Number of Lanes Based on the Minimum Travel Time. Among the influencing factors of travel behavior, time is the most valued factor by travelers. For the study of travel time on the road, the BPR model is most frequently used, and the road resistance function is as follows:

$$t = t_0\left\{1 + \alpha (V/C)^{\beta}\right\}, \tag{1}$$

where t is the travel time, t_0 is the travel time for free flow, V is the traffic volume, C is the traffic capacity, and α and β are parameters equal to 0.15 and 4, respectively.

Based on the minimum impedance, the path resistance function model is as follows:

$$\begin{aligned} \min f(x,i) &= t_0\left\{\left[1 + \alpha\left(\frac{V_1}{mC}\right)^{\beta}\right] + \left[1 + \alpha\left(\frac{V_2}{nC}\right)^{\beta}\right]\right\}, \\ t_o &= \frac{K_o L}{(V_s \cdot \gamma \cdot \eta)}, \\ i &= |m - n|, \end{aligned} \tag{2}$$

where V_1 and m refer to the traffic volume and number of lanes in one direction, V_2 and n refer to the traffic volume and number of lanes in the other direction, C represents the traffic capacity, I is the number of reversible lanes, K_o is the delay coefficient ranging from 1 to 1.2, L is the length of reversible lane, V_s is the design speed of the road, r is a parameter, if the nonmotorized lane and motor lane are

Table 1: Road conditions and corresponding service levels.

Service levels	State 1	State 2	State 3	State 4	State 5
V_1/C_1	0~0.6	0~0.8 (0.6~0.8)	0~0.8	≥0.8	≥0.8
V_2/C_2	0~0.6	0.6~0.8 (0~0.8)	≥0.8	0~0.8	≥0.8

V_1/C_1 represents the service level of the traffic in one direction; V_2/C_2 represents the service level of the traffic in the other direction. State 1 is the best state, and state 5 the worse state.

separated, r equals to, otherwise r equals to 0.8, and η equals to 1 if the width of lanes is 3.5 meters.

2.2.3. Determination of the Two-Way Traffic Conditions after the Lane Change Based on the Adjusted Service Level.

$$V_1 = \sum_{i=1}^{m} V_i = m \cdot v_1,$$
$$V_2 = \sum_{i=1}^{n} V_i = n \cdot v_2, \tag{3}$$
$$V_1/C = \frac{V_1}{m \cdot c} = \frac{m \cdot v_1}{m \cdot c} = \frac{v_1}{c},$$
$$V_2/C = \frac{V_2}{n \cdot c} = \frac{n \cdot v_2}{n \cdot c} = \frac{v_2}{c}.$$

Let $V_M = \max(V_1, V_2)$ and $V_N = \min(V_1, V_2)$, then the congestion service level can be calculated as follows:

$$V_M/C = \frac{V_M}{(M+i) \cdot c} = \frac{M \cdot v_M}{(M+i) \cdot c}. \tag{4}$$

The adjusted service level of the uncongested segment is not lower than the service level of the congested segment, then we have the critical state:

$$\frac{N \cdot v_N}{(N-i) \cdot c} \geq \frac{M \cdot v_M}{(M+i) \cdot c},$$
$$V_N \geq \frac{N-i}{M+i} V_M. \tag{5}$$

2.2.4. Establishment of Dynamic Control Model of Lane Change. The lane change dynamic control model obtains road traffic information in CVIS and evaluates the road traffic flow conditions based on the service level of the road. Then, the corresponding control module is used to solve the optimal switching scheme and determine whether the scheme meets the preset conditions. If it is satisfied, the optimal solution is executed; if it is not satisfied, the existing solution is maintained. The flow chart of dynamic control model is shown in Figure 1.

2.3. Road Signaling. The intelligent road stud is used as a road signal indicating device to dynamically change the lane allocation and guide the vehicle throughout the road. The intelligent road stud light uses lithium battery and solar battery to power. It can work continuously for 36 days under the supply of lithium battery power. It uses 6 2000 mcd LED lights to meet the signal indication requirements under different conditions. Through LoRa's new generation of IoT communication technology, point-to-point and point-to-center ultra-long-distance communication is realized at the same time. The road stud light adopts cast aluminum housing, with IP68 waterproof feature, and the working temperature range is −40° to 85°; it can withstand 20 tons of static pressure, which can meet the requirements of different scenarios and different roads.

The Intelligent road stud has three light colors, red, yellow, and green, and can be displayed in flashing. Red flashing means reminding to leave the lane and prohibiting traffic; green flashing means the lane is about to open and allowing traffic; the yellow light indicates the buffer section, which means to drive away as soon as possible and warn drivers that there is a no-entry section ahead. There are six patterns for road stud lights, as shown in Figure 2. The road management agency can change the color of the lights through the remote communication function to achieve the indication of different road traffic conditions. The remote instructions of the road stud lights are encrypted by RSA to achieve the security of traffic infrastructure control.

In combination with the existing laser projection technology, a virtual wall curtain idea formed by light projection was proposed. A light curtain wall was provided with a small projection device in a stud light to project light onto a reversible lane to form a virtual wall curtain; the brightness and color of the wall curtain are consistent with the stud lights. The light curtain wall can cooperate with road stud lights to provide instructions and reminders to ensure that vehicles can safely change lanes and drive. At present, the light curtain wall technology does not meet the requirements for practical application to roads. According to technical research and experimental analysis, light curtain wall technology can be applied to roads as a traffic signal indicator in the future.

2.4. Traffic Safety Analysis

2.4.1. Conflict Zones. A conflict zone of the reversible lane is divided into a starting conflict zone and an ending conflict zone. The objective of setting conflict zones is to make the vehicles in both directions change lanes smoothly and avoid conflicts. The starting conflict zone consists of a no-entry section. The ending conflict zone consists of two buffer sections and a no-entry section. The vehicle can complete the lane change or stay for a while waiting for a lane change in the buffer section. During the operation of the variable lane, the traffic is allowed to enter the no-entry section, forming a spatial separation zone for traffic flow to avoid oncoming conflicts.

When setting a conflict zone on a city road, the location should be as far away as possible from intersections, bus stops, and other places where traffic flow will be disturbed. If the reversible lane is at the beginning or end of the intersection, it is not necessary to set a conflict zone. The intersection is an effective area to avoid oncoming conflict.

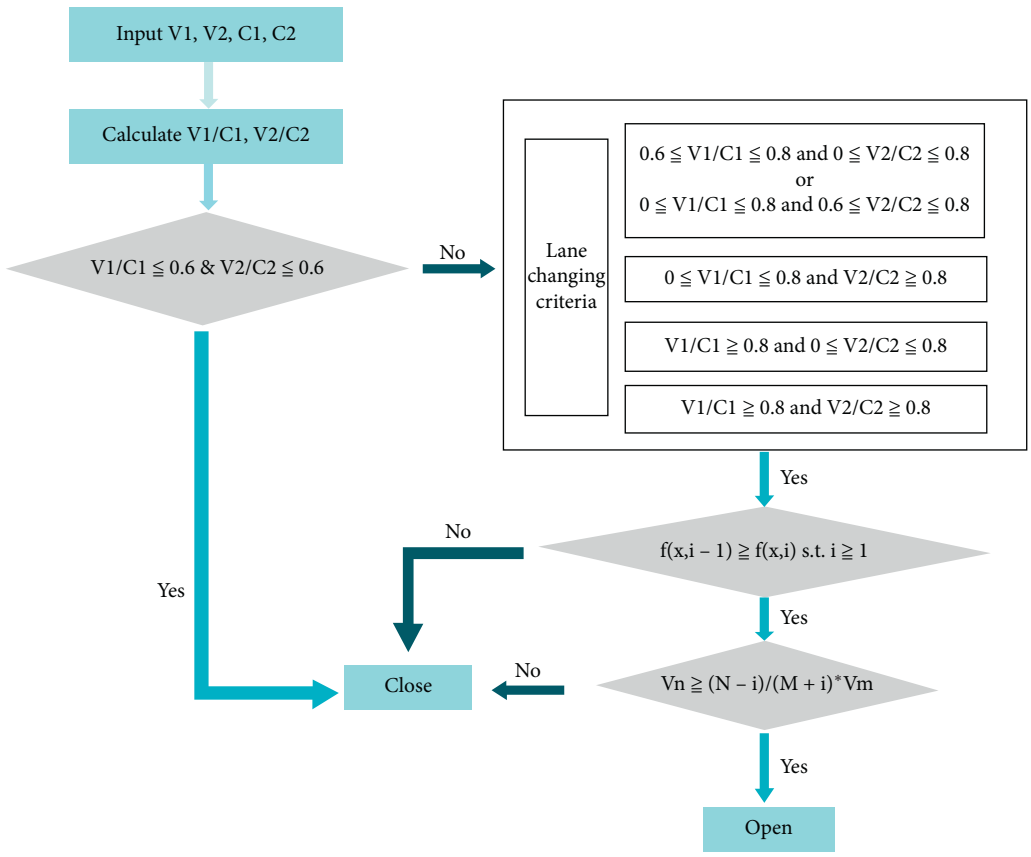


Figure 1: Dynamic control model flow chart, where V_m and m are traffic volume and number of lanes in the congested direction, respectively, V_n and n are traffic volume and number of lanes in the opposite direction, i refers to number of reversible lanes, and when $f(x, i-1) \geq f(x, i)$, the number of reversible lanes equals to i.

Road stud light patterns	Pattern 1	Pattern 2	Pattern 3	Pattern 4	Pattern 5	Pattern 6
Color						

Figure 2: Different patterns of intelligent road stud light.

Therefore, when the distance between the start or end of the reversible lane of the road section and the center of the intersection is within 300 m and the intersection satisfies the conditions of introducing the reversible lane, the reversible lane should be directly extended to the intersection. When the intersection does not meet the requirements for setting a reversible lane, the reversible lane should be shortened to ensure that the distance from the conflict zone to the intersection is more than 300 m. Because the reversible lane extending to the intersection does not need to set the conflict zone, we will only discuss the corresponding setting parameters for setting the conflict zone far away from the intersections. The starting conflict zone of the reversible lane is relatively simple to set. There is no direct oncoming conflict for two-way traffics in starting conflict zone, so we only need to avoid conflicts happen in the lane change process. The distance of the starting conflict zone depends on the vehicle's safe following distance and road stud laying interval which is generally 5–10 m.

In order to alleviate traffic congestion of eastbound traffic, vehicles in the eastbound traffic can borrow a part of the middle lane of the westbound traffic. Figure 3 shows the starting conflict area of the reversible lane. Vehicles in the eastbound traffic can choose to merge left to enter the reversible lane in the starting conflict area, and vehicles in the westbound traffic can merge left to get back to its original lane after the starting conflict zone. In Figure 4, the ending conflict area is the end area of the lane occupancy for vehicles in the eastbound traffic and the starting area of lane occupancy by the eastbound traffic for vehicles in the westbound traffic. Therefore, vehicles in the eastbound traffic must drive back to the original lane, and vehicles in the westbound traffic must drive into other lanes in the eastbound direction to avoid oncoming conflicts.

2.4.2. Design of No-Entry Section. The no-entry section is the isolated area of the vehicle during the operation of the reversible lane. It is set at the beginning and end of the reversible lane where the two-way traffic conflicts. In order to ensure the safety of reversible lane operation, the longer the distance of the no-entry section is, the better. However, in the perspective of utilization of road resources, the shorter no-entry section is preferred. Therefore, it is necessary to make the distance of the no-entry section as small as possible while ensuring the safety operation of the traffic. The length

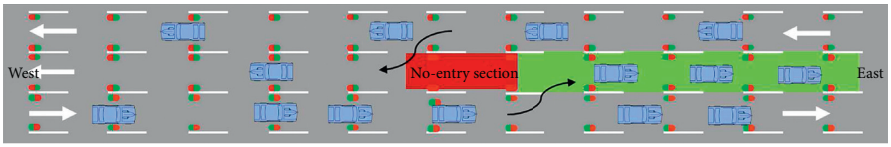


Figure 3: Configuration of starting conflict zone.

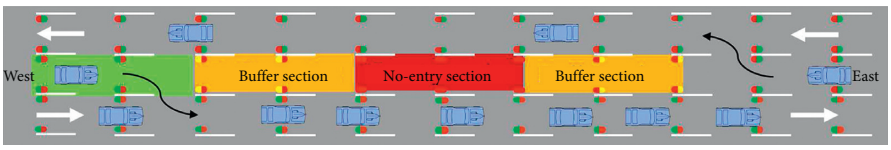


Figure 4: Configuration of ending conflict zone.

of the starting no-entry zone is the length of the starting conflict zone, and it is the safety following distance between vehicles. Generally, it is 5–10 m. The parameters of the ending on-entry zone are discussed below.

If a driver in one direction enters the end of the no-entry section, then they must be able to make a complete stop before the end of the no-entry section; if two drivers from opposite directions have entered the no-entry section, they must have sufficient space to stop cars to avoid a collision. Therefore, the length of the no-entry section must meet the sight distance of the vehicle and ensure the safe braking distance. The shortest length is determined by the reaction distance (l_{r1}, l_{r2}), breaking distance (l_{b1}, l_{b2}), and safety following distance (l_0) of two opposite traveling drivers as shown in Figure 5.

Assume the vehicle braking performance and response time of drivers are the same, then

$$\begin{aligned} l_1 &= l_2 = \frac{V_0}{3.6}t, \\ l_{b1} &= l_{b2} = \frac{V_0^2}{2g\varphi \times 3.6^2}, \\ S &= l_{r1} + l_{r2} + l_{b1} + l_{b2} + l_0, \end{aligned} \tag{6}$$

where t is the reaction time ranging from 1 to 2.5 seconds, V_0 is the vehicle speed before breaking, g is the gravity, and φ is the coefficient of adhesion of wheels.

2.4.3. Design of Buffer Section. If one is driving in the buffer section of a reversible lane, one should wait for the opportunity to change lanes and leave the reversible lane when one sees the yellow light on the road stud. If one reaches the buffer section and has not found the opportunity to leave the reversible lane, one should immediately brake and stop, waiting for the time to leave the reversible lane. If the one is forced to change lanes here, one should change lanes before running into a no-entry section. Therefore, the distance of the buffer section of a reversible lane must meet the maximum braking distance of the driver at the beginning the buffer section and the maximum length of the lane change forced by the driver at this point.

A vehicle will be guided to compulsory lane change at the beginning of a buffer section. According to the critical distance model of compulsory lane change, the deceleration of vehicle braking during is a and the initial driving speed is v, taking into account the degree of caution of the driver, the traffic density of the target lane, and the number of interval lanes between the current lane and the target lane. The formula for calculating the distance traveled by a vehicle during a lane change is

$$D_c = \frac{v^2}{2a} + \theta\left(\alpha_1 \frac{\rho}{\rho_{jam}}\right) + \alpha_2 n, \tag{7}$$

where θ is the cautious coefficient of drivers ranging from 0 to 1, the larger the number, the more cautious the driver is, α_1 and α_2 are parameters depends on traffic condition and number of interval lanes between the current lane and the target lane, ρ is the traffic density, ρ_{jam} is the traffic congestion density, and n is the number of reversible lanes.

Vehicle braking deceleration is related to the driver's judgment of the braking distance and is also related to the driver's driving habits, road conditions, and vehicle speed on the road. From the perspective of the braking capacity of a car, the maximum deceleration of the car during emergency braking is generally $7.5 \sim 8\text{m/s}^2$. For a normal braking, the average deceleration of the car should be $3 \sim 4\text{m/s}^2$. However, in real situation, except for emergency situations, the braking deceleration should normally be $1.5 \sim 2.5\text{m/s}^2$. Thus, a value of $2\ \text{m/s}^2$ of deceleration is used in this paper.

If the driver takes breaks at the beginning of the buffer section, the braking distance of the vehicle is calculated based on the vehicle's speed, braking performance, and road friction coefficient. The driver's reaction time from making a braking decision to force the brake is called the driver's reaction time t_0. The time interval from when the brake is forced to when the brake is in effect is called the driver's operating time, which is represented by t_1. Assume total the reaction time $t = t_0 + t_1$, and define t_2 is the time interval from when brake is in effect to when the car completely stops. The total braking distance of the vehicle could be calculated as follows:

$$D_b = \frac{V_0}{3.6}t + \frac{1}{2}t_2\frac{V_0}{3.6} + \frac{V_0^2}{2g\varphi \times 3.6^2}. \tag{8}$$

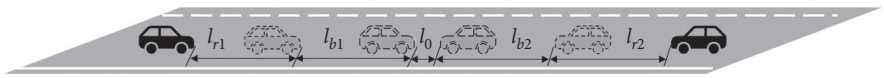
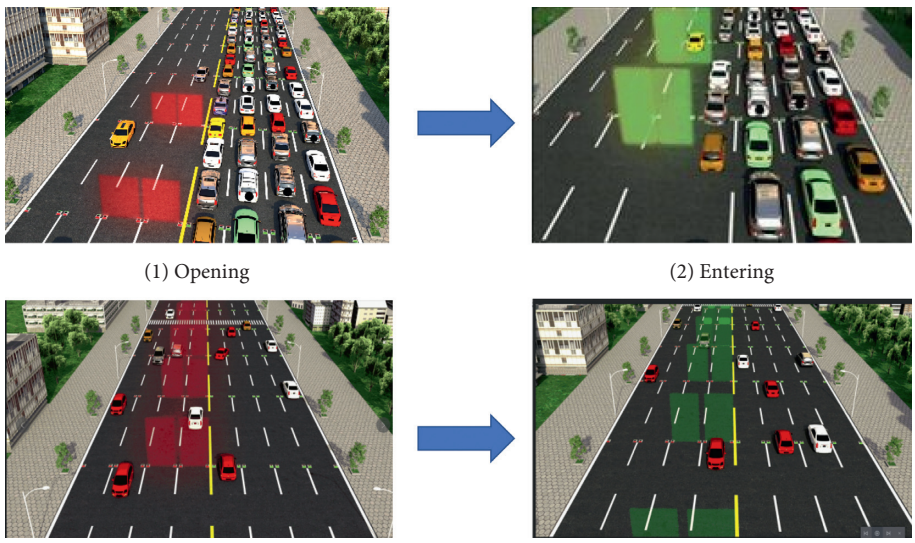


Figure 5: Configuration of a no-entry section.



(1) Opening (2) Entering

(3) Closing 1 (4) Closing 2

Figure 6: Illustrations of the real-time dynamic reversible lane operation process.

Let $t = 2.5$ s, and t_2 is negligible so the second term drops out. The reaction distance S_1 and breaking distance S_2 can be calculated as follows:

$$S_1 = \frac{V_0}{3.6} t = 0.694 V_0,$$

$$S_2 = \frac{V_0^2}{2g\varphi \times 3.6^2} = 0.00394 \frac{V_0^2}{\varphi}, \tag{9}$$

$$D_b = S_1 + S_2 = 0.694 V_0 + 0.00394 \frac{V_0^2}{\varphi}.$$

The length of a buffer section is

$$L_{\text{buffer}} = \text{Max}\left(D_c, D_b\right) + l_0. \tag{10}$$

3. Simulation and Data Analysis

3.1. Simulation. In CVIS, real-time traffic information is obtained and lanes in the direction of light traffic flow are selectively cleared based on real-time traffic volume. Taking an eight-lane road as an example, the two sslanes on the inner side of the light traffic flow direction are cleared, and the real-time dynamic reversible lane operation process is demonstrated through animation simulation.

(1) *Opening.* When the reversible lane opening conditions are reached, the real-time dynamic reversible lane will be open. In the direction of light traffic, the red studs on two lanes are flashing first and the light curtain wall flashing red, indicating that the vehicles in these lanes should leave as soon as possible. No traffic is allowed on these lanes, supplemented by a voice prompt in the car: "Please leave these lanes," as shown in Figure 6(1).

(2) *Entering.* After the target lanes are cleared, vehicles in heavy traffic flow can merge onto the cleared lanes. First, the stud light and the light curtain wall flash green, indicating that the target lanes are about to open, and then the stud light and the light curtain wall stay constant green, indicating that the target lanes have been opened and allow entrance, as shown in Figure 6(2).

(3) *Closing.* In the direction of the original heavy traffic, the stud lights on two lanes are first flashing red and light curtain walls flashing red, indicating that the vehicles in the lanes should leave the lane as soon as possible. Then, the stud lights and light curtain walls are in constant red, indicating that vehicles in the heavy traffic are not allowed on these two lanes and supplemented by a voice prompt in the car: "Reversible lane is closed, please drive away from these lanes," as shown in Figure 6(3). In the direction of the light traffic flow, stud lights are first flashing green and the light curtain wall flashing green as well, which indicates that the lanes are about to be reopen to light traffic. Then, the stud light and the light curtain wall are in constant green, indicating that the target lanes have been reopened to vehicles in the light traffic, as shown in Figure 6(4).

Table 2: Traffic flow data.

T	18 : 05	18 : 10	18 : 15	18 : 20	18 : 25	18 : 30	18 : 35	18 : 40	18 : 45	18 : 50	18 : 55	19 : 00
V_M	2560	2670	2840	3000	3260	3500	3730	3900	4000	4200	4350	4460
V_N	2100	2150	1980	1870	1740	1620	1580	1390	1280	1200	1170	1050

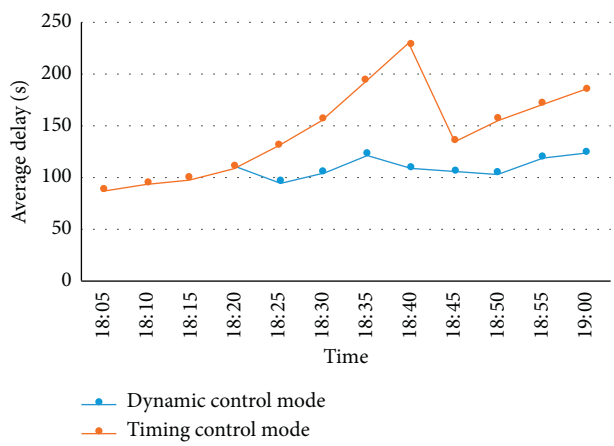


Figure 7: Comparison of average delay of dynamic control mode and timing control mode.

3.2. Data Analysis. A survey was conducted to collect traffic flow data from 18 : 00 to 19 : 00 of an 8-lane highway in Huai'an city, as shown in Table 2. The microscopic simulation software VISSIM and its secondary development are used to simulate and evaluate the vehicle running effect of the road in timing control mode and dynamic control mode. The timing control mode is controlled by the traffic police officer according to the traffic conditions of the road. It was turned on at 18 : 40 in the simulation. The dynamic control mode is controlled using the lane change control model of the proposed scheme.

Through the analysis of simulation data, with the continuous increase of road traffic, the average delay of vehicles in the dynamic control mode is relatively slow (Figure 7), the overall average delay of vehicles is reduced by 27.4%, and the vehicle VOC, CO, and NOX emissions are reduced by 13.5%. Therefore, the dynamic control mode has better effects in improving traffic efficiency and environmental protection.

4. Conclusions

This paper proposed a real-time dynamic reversible lane scheme in CVIS. Compared with the traditional time-controlled lane change mode, the proposed scheme can reduce the average delay of vehicles by 27.4% and reduce vehicle VOC, CO, and NO_X emissions. The volume reduction of 13.5% effectively improves the imbalance of traffic flow and alleviates road congestion, improves road traffic efficiency, reduces energy consumption, and provides a new solution for the promotion of new energy-saving and environmentally friendly reversible lane designs. With the continuous development and application of urban traffic guidance system and vehicle-road collaboration technology, combining it with this solution can more effectively optimize road resource allocation and improve vehicle traffic efficiency. The proposed dynamic signal control mode with intelligent road stud lights can guide urban roads in different road conditions. The combination with bus priority can effectively improve the utilization of existing bus lanes and other facilities. Adhering to the concept of energy saving and environmental protection, optimizing road resources, and innovating signal control modes, a new solution for the optimization of smart city transportation systems was proposed.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

Lina Mao conceived and designed the paper; Wenquan Li conducted the model and simulation; Lina Mao and Pengsen Hu wrote the paper; Guiliang Zhou designed the real-time dynamic reversible laneschemes; Huiting Zhang analyzed the simulation results, Jin Dai collected traffic data.

Acknowledgments

This research was supported by the open fund for Graduate Innovative Projects of Jiangsu Province (KYLX15_0148), the National Natural Science Foundation of China (61573098 and 51308246), Jiangsu key laboratory of traffic and transportation security (Huaiyin Institute of Technology) (TTS2020-06), Enterprise-University-Research Institute Collaboration Project of Jiangsu Province (DH20190231), University Natural Science Major Basic Project of Jiangsu Provence (15KJA580001), The Natural Science Foundation of Jiangsu Province, China (BK20171426), Youth Foundation of Huaiyin Institute of Technology (HGC1408), and Chinese Government Scholarship for Overseas Studies (CSC NO. 201506090109)

References

[1] J. F. Guo and Y. Liu, "Understanding of traffic congestion in China's big cities," *Urban Traffic*, vol. 9, no. 2, pp. 8–14, 2011.

[2] H. F. Tan, "Current situation and development countermeasures of intelligent transportation system in china," *Management and Technology of Small and Medium-Sized Enterprises*, vol. 1, pp. 48-49, 2019.

[3] S. Mou, "Research on urban traffic network based on Internet of things technology," *Electronic World*, vol. 13, pp. 206-207, 2018.

[4] G. H. Zhou, "Study on the setting of reversible lane in urban road engineering," *Transportation Science &Technology*, vol. 3, pp. 116–119, 2012.

[5] X. Sun, H. Zhang, W. Meng, R. Zhang, K. Li, and T. Peng, "Primary resonance analysis and vibration suppression for the harmonically excited nonlinear suspension system using a pair of symmetric viscoelastic buffers," *Nonlinear Dynamics*, vol. 94, no. 2, pp. 1243–1265, 2018.

[6] H. Xiong, X. Zhu, and R. Zhang, "Energy recovery strategy numerical simulation for dual axle drive pure electric vehicle based on motor loss model and big data calculation," *Complexity*, vol. 2018, Article ID 4071743, 14 pages, 2018.

[7] B. Wolson and L. Lambert, "Reversible lane systems:synthesis of practice," *Journal of Transportation Engineeringg*, vol. 132, no. 12, pp. 933–944, 2006.

[8] M. Hausknecht, T.-C. Au, and P. Stone, "Dynamic lane reversal in traffic management," in *Proceedings of the 2011 14th International IEEE Conference on Intelligent Transportation Systems*, IEEE, Washington, DC, USA, pp. 1929–1934, October 2011.

[9] A. Golub, "Perceived costs and benefits of reversible lanes in phoenix, Arizona," *Institute of Transportation Engineers*, vol. 82, no. 2, pp. 38–42, 2012.

[10] Q. Yu and R. Tian, *Research on Reversal Lane Application Method of Urban Road Network Based on the Bi-level Programming*, pp. 983–992, Springer, Berlin, Germany, 2014.

[11] Y. Cui and D. Liu, "Traffic organization of reversible lane in chaoyang district, Beijing," *Road Traffic and Safety*, vol. 6, no. 9, pp. 21–24, 2006.

[12] Q. Sun, "Research on dynamic reversible lane optimization algorithm," *Highway*, vol. 8, pp. 300–303, 2009.

[13] R.-H. Zhang, Z.-C. He, H.-W. Wang, F. You, and K.-N. Li, "Study on self-tuning tyre friction control for developing main-servo loop integrated chassis control system," *IEEE Access*, vol. 5, pp. 6649–6660, 2017.

[14] M. Liu, L. Xu, L. Shen, and S. Jin∗, "Modeling capacity at signalized intersections with a left-turn storage bay considering signal timing plan," *ASCE Journal of Transportation Engineering Part A: Systems*, vol. 145, no. 2, Article ID 04018084, 2019.

[15] L. L. Dai, J. G. Gu, J. Yu, and H. T. Qiu, "Study on simulation of tidal lane traffic flow characteristics and setting scheme," *Traffic Information and Safety*, vol. 1, no. 30, pp. 15–19, 2012.

[16] D. Leilei, "Traffic flow characteristics on reversible lane and its operational plan based on simulation," *Computer and Communications*, vol. 30, no. 1, pp. 15–19, 2012.

[17] S. K. You, "Optimization design of reversible lane guidance signs and organizational plan," in *Proceedings of the 2016 Annual Conference on Urban Transportation Planning in China*, Shanghai, China, July 2016.

[18] L. Yang, *Study on the Scheme of Setting Reversible Lanes on Torch Avenue*, Dissertation of Chongqing Jiaotong University, Chongqing, China, 2015.

[19] C. Xu, H. Guo, L. Xu, and S. Jin∗, "Speeding behavior and speed limits for heterogeneous bicycle flow," *Traffic Injury Prevention*, vol. 20, no. 7, pp. 759–763, 2019.

[20] M. Zhou, X. Qu, and S. Jin∗, "On the impact of cooperative autonomous vehicles in improving freeway merging: a modified intelligent driver model-based approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 6, pp. 1422–1428, 2017.

[21] Y. Y. Ma and L. Y. Zeng, *Research on Dynamic Control Method of Reversible Lane Driving Direction, Control Theory and Application*, vol. 11, no. 1, pp. 1457–1462, 2016.

[22] F. T. Jiao, *Reversible Lane Dynamic Control Method Based on Multi-Source Data*, Dissertation of Shandong University of Science and Technology, Zibo, China, 2018.

Hindawi
Journal of Advanced Transportation
Volume 2020, Article ID 6147974, 11 pages
https://doi.org/10.1155/2020/6147974

Research Article

A Distribution Model for Shared Parking in Residential Zones that Considers the Utilization Rate and the Walking Distance

Wenhui Zhang,[1] Fan Gao,[2] Shurui Sun,[1] Qiuying Yu,[1] Jinjun Tang,[2] and Bohang Liu[3]

[1]*School of Traffic and Transportation, Northeast Forestry University, Harbin 150040, Heilongjiang, China*
[2]*Smart Transport Key Laboratory of Hunan Province, School of Traffic and Transportation Engineering, Central South University, Changsha 410012, Hunan, China*
[3]*Key Laboratory of Traffic Safety and Control of Hebei Province, Shijiazhuang Tiedao University, Shijiazhuang 050043, Hebei, China*

Correspondence should be addressed to Jinjun Tang; jinjuntang@csu.edu.cn and Bohang Liu; liubohang@126.com

Received 6 January 2020; Accepted 11 February 2020; Published 23 March 2020

Guest Editor: Dongfang Ma

Copyright © 2020 Wenhui Zhang 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.

Efficient parking tends to be challenging in most large cities in China. Drivers often spend substantial amounts of time looking for parking lots while driving at low speeds, thereby resulting in interference with road traffic. This paper focuses on efficiently allocating parking spaces to the demanders. A double-objective model is proposed that considers both the utilizing rate and the walking distance. First, managers want to utilize parking resources fully. Therefore, they tend to prioritize the efficient distribution of parking spaces in response to parking demands. However, demanders typically choose parking spaces according to convenience. The second objective is the acceptable walking distance from the parking space to the destination. The particle swarm optimization (PSO) algorithm is used to solve this model. We collected parking demand and supply data in a central business district (CBD) of Harbin in China and evaluated the feasibility of the model. The results demonstrate that the proposed model increases the occupying rates of parking lots in residential zones while decreasing the walking distance. The shared use of parking spaces maximizes the utility and alleviates the shortage of parking spaces in downtown.

1. Introduction

Parking efficiently is a significant concern in traffic planning in many modern cities due to the fast increase in the number of private cars [1]. Parking planning has attracted attention from traffic engineers and managers because the unreasonable design of parking lots may reduce the capacity of urban roads and increase the risk of accidents. For example, the shortage of auxiliary parking lots and public parking spaces forces drivers to leave their cars at the roadside, which not only causes traffic congestion but also traffic conflicts. Drivers tend to decrease the speed of their cars to find a parking lot, which will also affect the following vehicles' normal operation. However, it is difficult to thoroughly solve these problems by establishing new parking facilities [2], especially in central business districts (CBDs), where there is a shortage of parking spaces due to limited land use.

Similarly, parking problems are severe in many core areas in the urban city where the buildings are dense. The availability of parking spaces is even more inadequate. The time that travelers spend finding a suitable parking space and walking to his or her workplace may constitute a large part of the total travel time [3]. Cruising process can also produce a substantial amount of congestion and exhaust pollution [4, 5]. However, many residential parking spaces near a CBD are available after the owners go out in daytime. These vacant parking spaces can be meanwhile provided for demanders who go shopping or do other things in a CBD. This mode of availability is cycled every day, which provides a parking gap for demanders who come to CBD in the daytime. Hence, this gap parking mode, which is called shared parking, not only helps the parking space owners get extra profits but also substantially alleviates the urgent parking needs downtown.

Systematic parking management mostly focuses on effective countermeasures for resolving the imbalance between supplies and demands. The proposed parking strategy involves not only increasing the parking supply but also efficiently utilizing current parking lots [6]. Effective parking

management requires a more comprehensive consideration of the balance between supply and demand [7–9]. This is challenging for most parking managers. They typically take various methods which include shared parking, publicly displaying available parking spaces, and adjusting parking charges [10, 11]. The safety and reliability of parking facilities during use are considered by managers [12, 13]. The management objectives for parking lots basically include realizing the maximum occupying rate, the minimum cruising duration, and the most rational benefits.

Parking choice, which seems a subjective behaviour for a driver, directly affects parking management strategies [14]. Driver choice behaviour has been extensively studied. Drivers' previous experiences are believed to play a substantial role in the selection of a parking lot [15]. In addition, drivers will take the travel distance, parking cost, and travel purpose into account [16, 17]. Accordingly, shared parking is gradually becoming accepted by suppliers and demanders of parking spaces. The shared parking fee is typically lower than the fee in the nonshared parking mode, and the demanders can easily select the nearest parking lots online in advance. Zargayouna et al. proposed an online shared information system for parking demanders. The drivers could choose the parking spaces in advance, and the searching time was decreased [18]. Kaspi et al. presented a parking reservation method online by use of a Markov model. The applications showed that the total excess time was reduced by more than 14% [19, 20]. Shao et al. suggested that drivers reserved the favorite parking lots on platform, which provided profit both for owners and users [21].

Searching a vacant parking space tends to take a long time and leads to exhaust emission and congestion [22]. Fortunately, intelligent systems, by collecting and releasing the data from demanders and suppliers, can facilitate the automatic distribution of parking spaces [23]. Therefore, managers can more easily plan parking lots and drivers can more easily locate available parking spaces. Many macro- and micromodels of parking management and space distribution have been proposed [24–26]. In addition, the imbalance between parking supply and demand by constructing complex intelligent parking systems is a hot topic. For example, the "Smart Parking" and "Parking guidance" systems can intelligently assign a selected parking space to a driver [27, 28]. Afterwards, parking spaces can be assigned among buildings or lots based on a network algorithm [29]. Jin Cao and Monica Menendez evaluated the practicability and economy of intelligent parking systems and found that the application of an intelligent system reduced the searching duration by 17% [30]. With the rapid development of smart devices and wireless sensor networks, a variety of web-based apps and online parking search sites have emerged in recent years. For example, SpotHero, Pango, and Parkme are web and mobile applications that enable drivers to view parking information and reserve and pay for parking from any smart device in New York City.

The following sections of this paper describe the shared parking characteristics, parking spaces allocating model and solution, case study and conclusions. Section 2 analyzes the shared parking characteristics and then presents parameters and variables to be used later. Section 3 proposes a mathematical model for allocating parking spaces and the particle swarm optimization to solve the model. Based on the data collected from CBD in Harbin, China, a numerical simulation is conducted in Section 4. The last section presents the conclusions of this paper and provides perspectives.

2. Shared Parking Characteristics and Term Definitions

2.1. Problem Description. Shared parking basically involves the collection, release, and processing of information from supply and demand sides. These operations can be handled by an online shared parking system. Normally, reserving a parking space online is ahead of one's actual occupation. Drivers need to search the suitable parking spaces in advance. The demander side seems to play an active role because it can choose parking lots and cancel reservations at will. Parking locations depend on drivers' preferences because they tend to take the properties of parking lots, walking distances, and parking fees into account in most cases. However, the shared parking system can provide recommendations based on optimization models as well. These models generally consider the walking distance of drivers after parking, time windows of supply and demand, and utilization of parking spaces.

2.2. Conditions for Shared Parking. Shared parking is an effective way of optimizing the utilization of parking spaces, especially in large cities. To realize parking sharing, three conditions should be satisfied: (1) multiple destinations are located around the parking lots; (2) the walking distance after parking is within an acceptable range; (3) parking spaces are open to the public or vehicles are conditionally allowed to park. These conditions may meet people's needs regarding access to multiple destinations after parking and may ensure that drivers choose parking spaces in advance. Shared parking is typically an inherent part of downtown, as the same parking space serves multiple destinations within acceptable walking distance. For example, the parking lots of a residential zone may practically be shared by the people who work in an office building or shop in a supermarket nearby. Many parking spaces in a residential zone will be free during the daytime when there are urgent parking demands at the office building or the supermarket.

Shared parking may realize benefits for every participant. Owners of parking spaces can specify free periods on the platform where the drivers can easily search. A residential property or a parking management company can use some algorithms to manage parking spaces more efficiently. Shared parking can be simply described as follows. There are several destinations near the parking lots. Assume that each destination generates parking demands on a specified day and there is more than one residential zone near each destination. The owners of parking spaces formulate a leasable timetable of parking spaces according to their travel habits, commuting time, and occupying time. Then, the timetable will be available on the parking management

platform. A shared parking space may have one or more sharing periods. Every start time and end time for sharing should be specified on the management platform in advance. The platform normally allocates the parking spaces according to the requirements of demanders. If the demanders have no special requirements, the platform will automatically allocate parking spaces in accordance with the leasable timetable.

2.3. Term Definitions. The essential terms used in this paper are described as follows:

D: Commercial zones studied in this paper.

A: The set of destinations near the commercial zones, namely, $A = \{a/a = 1, 2, \ldots, k\}$, where k is the number of all destinations. The number of parking demands which are generated from each destination on a specified day is i. And there is no less than one residential zone near each destination.

H: The set of shared parking spaces in the residential zones, namely, where $\mathrm{h_i}$ is the number of parking spaces.

M: The set of parking demands, namely, $M = \{m/m = 1, 2, \ldots m_i\}$, where m_i is the number of all overflowing parking demands.

x_{mh}: A binary variable. $x_{\mathrm{mh}} = 1$ represents that parking demands can be successfully distributed to parking spaces; otherwise, $x_{\mathrm{mh}} = 0$.

In summary, x_{mh} is determined via

$$x_{\mathrm{mh}} = \begin{cases} 1, & \text{distributed successfully,} \\ 0, & \text{else.} \end{cases} \tag{1}$$

y_h: the parking space utilization means the ratio of the shared duration to the whole available duration

t^m_{arr}: the time when the demander arrives to the scheduled parking space

t^m_{lea}: the time when the demander leaves the parking space

t^m_{dur}: the actual occupying duration of a parking space

$$t^m_{\mathrm{dur}} = t^m_{\mathrm{lea}} - t^m_{\mathrm{arr}}. \tag{2}$$

a^s_h: the start time when the parking space is available

b^e_h: the end time when the parking space is available

s_h: the actual available duration of a parking space, which covers from a^s_h to b^e_h for per time gap

$$s_h = b^e_h - a^s_h. \tag{3}$$

d_{mh}: a variable that indicates whether the demanding time follow into the available time gap

f: the walking distance from the parking space to the destination

L_{mh}: the walking distance that drivers can be acceptable

$L_{\max}$: the maximum walking distance after parking

To analyze the influence of walking distance after parking on driver's choice, we conducted a survey in Le Song CBD in Harbin city, China. This survey was inspired by Waerden et al. in 2017 [31]. We proposed eight levels of waking distaces after parking, that were less than 200 m, 200 to 250 m, 250 to 300 m, 300 to 350 m, 350 to 400 m, 400 to 450 m, 450 to 500 m, and more than 500 m. The description of the question is "How far is your acceptable distance after parking?". The survey lasted for 7 days including weekdays and weekends. A total of 685 valid questionnaires were collected and 34 participants could accept more than 350 m. It meant that more than 95% of participants were unwilling to walk more than 350 m after parking. According to this survey, this study proposed that the acceptable maximum walking distance is equal to 350 m.

3. Methodology

3.1. Model Assumptions and Constraints. To realize parking sharing between the commercial zones and the residential zones, we propose a biobjective model. Assumptions and constraints are imposed to ensure the operability and the feasibility of the model.

3.1.1. Model Assumptions. The following assumptions are imposed in the model:

(1) The parking lots in the residential zone are accessible to the demanders and the parking spaces are easy to reach, namely, that there are no locks or other obstacles.

(2) The owners of the parking spaces comply with the platform regulations. They may choose the period for sharing their parking spaces, namely, they may set the sharing time window and the unit time price.

(3) Demanders who want to utilize the shared parking spaces should understand and abide relevant regulations. For example, if they occupy the parking spaces after the shared time ends, they will accept the penalty charging method.

(4) The demanders have no preference for specific parking lots in residential zones.

3.1.2. Constraints on Various Parking Demand Periods. Because travel purposes differ, parking times differ among drivers for the same parking space. Therefore, each parking demand is restricted by the dwell time of other demands:

$$n_{m_1 m_2} = \begin{cases} 1, & \left(t^{m_1}_{\mathrm{arr}}, t^{m_1}_{\mathrm{lea}}\right) \cap \left(t^{m_2}_{\mathrm{arr}}, t^{m_2}_{\mathrm{lea}}\right) \neq \varnothing, \\ 0, & \left(t^{m_1}_{\mathrm{arr}}, t^{m_1}_{\mathrm{lea}}\right) \cap \left(t^{m_2}_{\mathrm{arr}}, t^{m_2}_{\mathrm{lea}}\right) = \varnothing, \end{cases}, m_1, m_2 \in M. \tag{4}$$

3.1.3. Constraints on the Shared Timetable and Parking Demand Periods. Similar to the parking demand, the

sharing period of a parking space is related to the owner's travel habits and commuting time. The demanding duration should be content with the available time gap of a parking space. So the demand decision matrix $D = [d_{\mathrm{mh}}]$, where $m = 1, 2, \ldots, M$ and $h = 1, 2, \ldots, H$:

$$d_{\mathrm{mh}} = \begin{cases} 0, & (t_{\mathrm{arr}}^m, t_{\mathrm{lea}}^m) \in (a_h^s, b_h^e), \\ 1, & \text{else}. \end{cases} \tag{5}$$

3.2. Mathematical Model of Parking Space Allocation. The shorter the walking distance after parking is, the more likely a demander will choose the corresponding parking space. The walking distance tends to be the dominant factor for demanders. However, for parking platforms, the shared space utilization rate is the main consideration because a higher utilization rate is associated with higher profits. Therefore, the most important objective of allocating parking spaces is to maximize the utilizing rate, as expressed in equation (6). The premise is that the walking distance does not exceed the maximum acceptable distance, as expressed in equation (7). Integrating the objective function and the specified constraints, the model can be expressed as follows.

$$\max\ y = \frac{\sum_{h_i=1}^{H} \sum_{m_i=1}^{M} t_{\mathrm{dur}}^m x_{\mathrm{mh}}}{S(H)}, \tag{6}$$

$$\begin{aligned} & f = \min_{L_{\mathrm{mh}} \le L_{\max}} x_{\mathrm{mh}} L_{\mathrm{mh}}, \\ \text{subject to,}\quad & x_{m_i} + n_{m_i m_j} \cdot x_{m_j} \le 1, m_i, \quad m_j \in m; m_i \ne m_j, \\ & d_{\mathrm{mh}} \cdot x_{\mathrm{mh}} = 0, \quad m \in M; h \in H, \\ & \sum_h x_{\mathrm{mh}} \le 1, \end{aligned} \tag{7}$$

where $S(H) = \sum_{h_i} s_h$, $s_h = \sum_{h_i} (a_h^s - b_h^e)$.

3.3. Particle Swarm Optimization Algorithm. Currently, Genetic Algorithm [32, 33], Markov Chains [34], and Neural Network Algorithm [35, 36] are widely used to solve multiobjective optimization problems. Although these algorithms can obtain a set of feasible solutions which satisfies the objective function, problems such as slow convergence and complex coding began to become apparent when the number of problem reach a larger size. PSO can overcome such problems of traditional global searching algorithms. PSO is inspired by the foraging behaviour of birds in nature and then gradually developed based on global searching algorithms. Compared with other heuristic algorithms, PSO algorithm is suitable for solving large-scale multiobjective problems. It converges fast to optimal solution and encodes simply because it only uses a few parameters for tuning [37–39]. Based on the mentioned characteristics, the PSO algorithm is applied to solve the double-objective model that considers the utilizing rate and the walking distance.

PSO algorithm generally includes some particles which can be expressed by two vectors, namely, a position and a velocity. The first vector means a potential action to address a problem and the second means the changing direction and magnitude variable. For example, in a dimension d space involving I particles, the position set of the i_{th} particle can be described as $x_i = (x_{i1}, x_{i2}, \ldots, x_{id})$, $i = 1, 2, \ldots, I$. Meanwhile, its speed can be described by $v_i = (v_{i1}, v_{i2}, \ldots, v_{id})$. The particle's position can be fixed by two factors, namely, the inertia factor ω and the learning factors c_1 and c_2. The individual optimal position p_{id} and the global optimal position p_{gd} are able to adjust their positions. The update of a particle position set $\{x_k\}$ and the velocity set $\{v_k\}$ is a random process with substantial random features. The following equations express the updating performance of the position and velocity of the i_{th} particle. They mean an iterating process from step k to $k+1$.

$$v_{\mathrm{id}}^{k+1} = \omega v_{\mathrm{id}}^k + c_1 r_1 \left(p_{\mathrm{id}}^k - x_{\mathrm{id}}^k\right) + c_2 r_2 \left(p_{\mathrm{gd}}^k - x_{\mathrm{id}}^k\right), \tag{8}$$

$$x_{\mathrm{id}}^{k+1} = x_{\mathrm{id}}^k + v_{\mathrm{id}}^{k+1}. \tag{9}$$

In the formula (8), r_1 and r_2 represent the independent random values which distribute uniformly from 0 to 1. c_1 and c_2 are learning factors, which are typically integers that are equal to 2.

The term ω, applied in global and local search, means the inertia factor. If it is large, the PSO algorithm is biased towards global search. If it is small, the algorithm is biased towards local search. The value of ω mostly adopts the linearly decreasing weighting strategy in the classical PSO iterative process, as expressed in equation (10), which is randomly distributed in the range of [0.1, 0.9].

$$\omega_k = \omega_{\mathrm{end}} + \left(\omega_{\mathrm{ini}} - \omega_{\mathrm{end}}\right) \frac{K - k}{K}. \tag{10}$$

In the formula (9), ω_{end} and ω_{ini} are defined as constants and their values are 0.4 and 0.9. The terms k and K are defined as the present and maximum iterative times, respectively. Figure 1 presents a flowchart of PSO algorithm. The process of obtaining optimal solutions is shown as follows:

(1) Creation and initialization of the parameters: the parameters involve the particles number, learning factors, weighting factors, maximum iterative times, current positions, and velocities

(2) Calculation and determination of the particles values: these values involve the fitness value, individual, and group optimized value

(3) Calculation and update of the inertia factor via equation (10)

(4) Iteration and replacement of the particles positions and velocities by use of formulas (8) and (9)

(5) Update and determination of the optimal values for individual and group particles

(6) Calculating stop or returning: the research will suspend and the calculating values will be exported when the times of maximum iteration is achieved, or go back to (3)

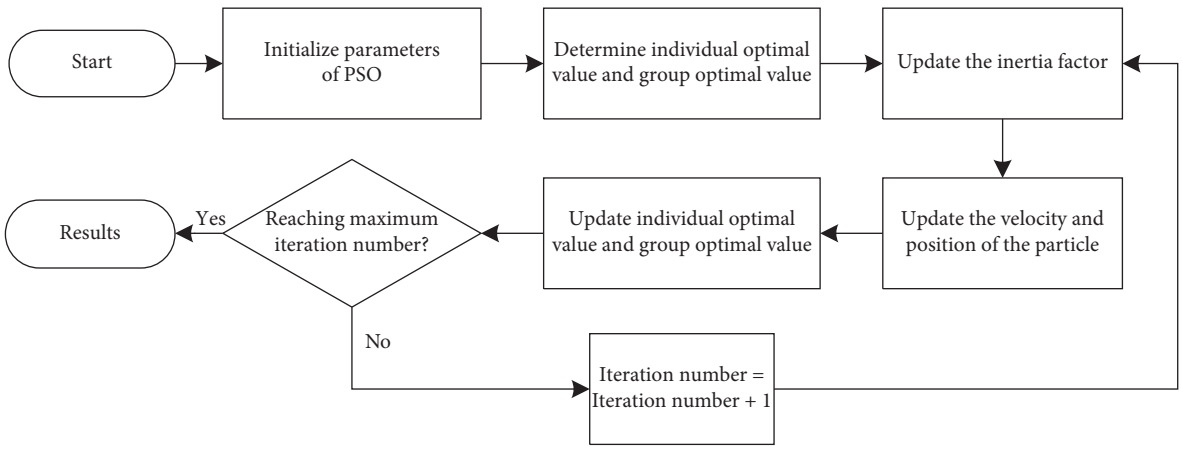


Figure 1: Flowchart of the PSO algorithm.

4. Case Study

4.1. Data Setup

4.1.1. Study Area. In order to observe the application of the developed model, the essential parking data was collected in Le Song CBD in Harbin City, China. It is one of the most prosperous commercial centers in the city. There are two commercial zones: Le Song Plaza and Song Lei Commercial Building. Two residential zones, namely, Peace Community and Triumph Square, are also located near the CBD. The commercial activities of Le Song Plaza and Song Lei Commercial Building are intensive during the daytime, which results in high parking demands. At the same time, the parking spaces in Peace Community and Triumph Square that are near the commercial zones are not fully utilized. The geographical locations of the four study zones in this paper are illustrated in Figure 2.

4.1.2. Data Preparation. To solve a dynamic online problem such as parking space distribution, a large amount of spatiotemporal data, including the walking distance after parking, the parking requests in commercial zones, and the idle parking spaces in residential zones, is necessary for identifying the optimal solution.

Table 1 lists the walking distances after parking between commercial and residential zones. These distances are actual distances between the entrances or exits of the buildings and the parking spaces. The distance between Le Song Plaza and Triumph Square is 213 m, the distance between Song Lei Commercial Building and Peace Community is 268 m, and the distance between Song Lei Commercial Building and Triumph Square is 296 m. However, the pairwise distance between Le Song Plaza and Peace Community is 472 m. The walking distance data set between these two commercial zones will not be used in this paper because the distance exceeds 350 m due to their geographical locations and intersection.

Le Song Plaza and Song Lei Commercial Building have no real-time parking requests because there is no shared parking guidance or distribution system. Therefore, the parking request data are represented by data regarding the vehicles in the commercial building parking spaces. The survey period was determined by the residents' commuting time in this area, from 9:00 AM to 5:00 PM. Table 2 shows the number of demands, the destination of each demand, and t_{dur}.

A total of 347 parking requests in these two commercial buildings during the 8-hour period is recorded in Table 2. Tables 3 and 4 show the shared parking lots and their rental time windows in Peace Community and Triumph Square, respectively. Note that the parking lots include both underground and ground parking lots. The Peace Community can provide a total of 150 parking lots for requests submitted by demanders while the Triumph Square provides approximately 100 parking lots. Suppose that no special events or holidays affect demanders' parking preference.

4.1.3. Parameter Settings. The multiobjective model of shared parking distribution can be solved by PSO algorithm. In the parameter optimization, the population size is taken as 50, the maximum iterative times is taken as 100, learning factors is taken set as 2, respectively ($c_1 = c_2 = 2$), the inertia factor is taken as 0.9 ($w_{\text{ini}} = 0.9$), and the ending inertial factor is set as 0.4 ($w_{\text{end}} = 0.4$). The optimization process is implemented according to Figure 2. A total of 50 independent runs of all algorithms is performed. After running the algorithms, we can obtain the demands from two commercial buildings to Peace community and Triumph Square.

4.2. The Characteristic of Parking. The utilizing rate of parking spaces reflects the performance of a parking lot. The more vacant parking spaces the parking lots have, the more the spaces can be provided for sharing. Figure 3 illustrates the utilization rate of Triumph Square and Peace community during daytime (between 9:00 and 17:00). We can observe that the parking lots have an obvious temporal characteristic in residential zones. Particularly, in Peace Community, two low peak periods (9:00–11:00 and 15:00–17:00 pm) are shown during daytime.

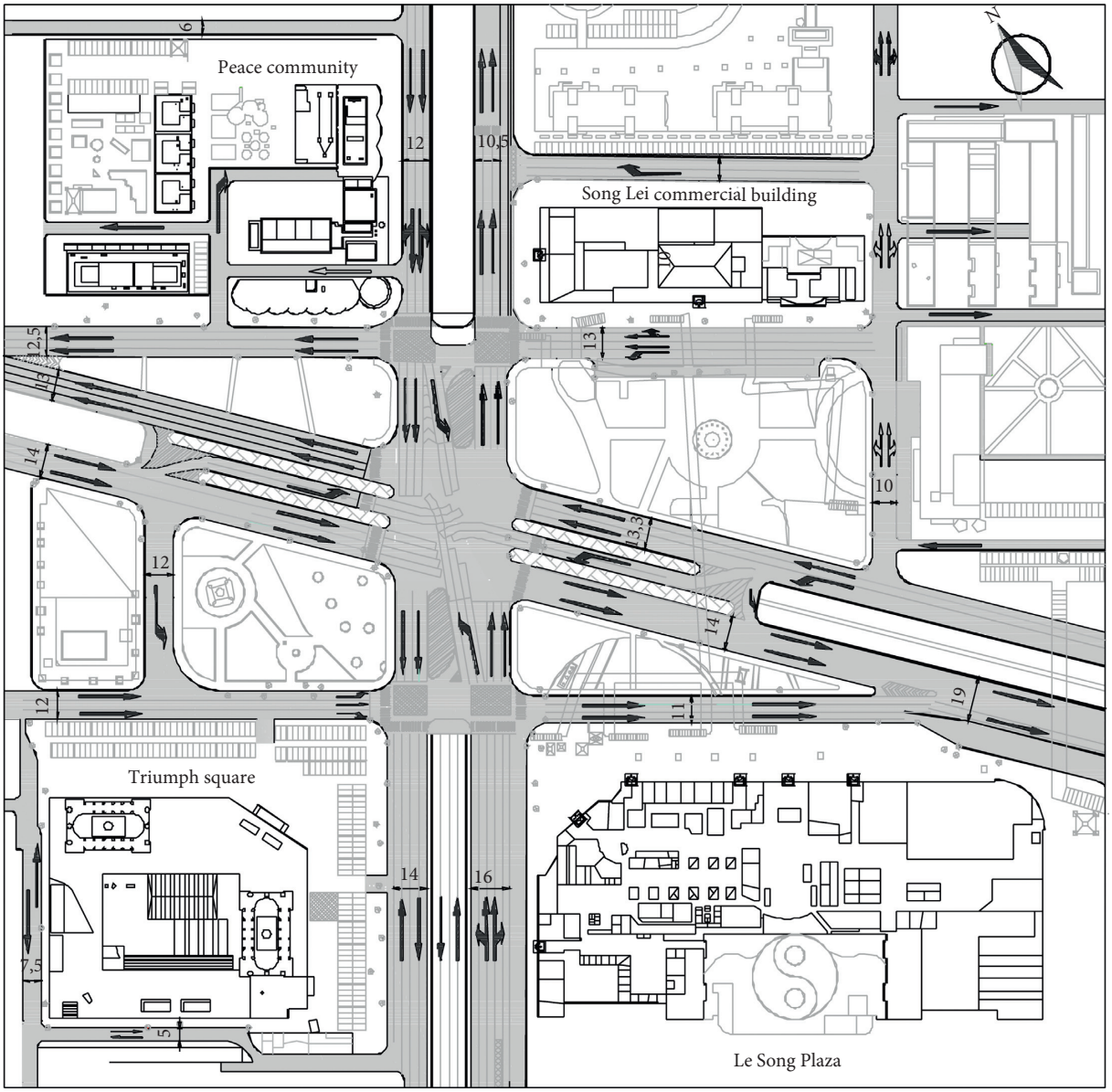


Figure 2: Study area.

Table 1: Walking distance after parking.

Commercial zones	Residential zones	
	Peace community (m)	Triumph square (m)
Le Song Plaza	472	213
Song Lei commercial building	268	296

The number of vehicles arriving at the commercial buildings reflects the potential parking demands. As shown in Figures 4 and 5, the number of arriving vehicles to Le Song Plaza and Song Lei commercial building varies as time goes by. The maximum demanding peak appears at 11:00 and 15:00. However, the nearby residential zones may provide the parking lots just then, as shown in Figure 3. As a result, the sharing of parking lots between residential and commercial zones tends to be a feasible approach to alleviate parking demands in commercial zones.

4.3. Analysis of the Results. The model considers two objectives, maximum utilization of spaces and minimum walking distance after parking. The two objectives represent the optimal utilization of public parking lots and the optimal

Table 2: Parking demands.

No.	Time	t_{dur}	Destination (1, 2)
1	9:00	9:00-10:30	1
2		9:00–11:00	2
3		9:00–11:00	1
4		9:00–15:00	2
⋮	⋮	⋮	⋮
346	16:00	16:00–16:3 0	2
347		16:00-17:00	1

1 represents Le Song Plaza and 2 represents Song Lei Commercial Building.

Table 3: Vacant parking spaces in Peace Community.

No.	Rental time window
1	9:00–12:00
2	9:00–11:00
3	9:00–14:00
...	...
149	16:00–17:00
150	16:00–17:00

Table 4: Vacant parking spaces in Triumph Square.

No.	Rl Te window
1	9:00–12:00
2	9:00–11:00
3	9:00–11:30
...	...
108	15:00–16:00
109	15:00–17:00

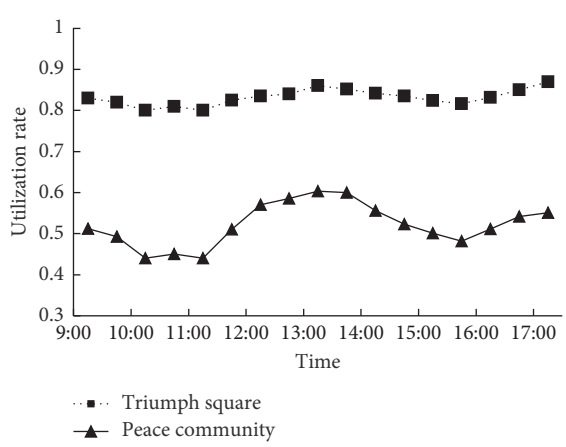


Figure 3: The changes of utilization rate between residential zones.

walking distance for the drivers. In addition, to efficiently guide drivers' parking, the optimal solution prioritizes the walking distance. Figures 5–7 show the assigned demands from two commercial buildings to the Peace Community and Triumph Square in the parking sharing plan.

Based on the characteristics of parking spaces supply and demand, the excessive demands of Le Song Plaza are distributed to Triumph Square and the demands of Song Lei Commercial Building are allocated to Peace Community and Triumph Square. It is possible to implement a shared parking distribution plan in CBD. In addition, the demands that are allocated by the three sharing plans during the study period are approximately equal; however, the peak hours of demand allocation to parking spaces differ. The peak hours of the sharing plans from Le Song Plaza to Triumph Square and from Song Lei Commercial Building to Triumph Square are 13:00–14:00, while the sharing plan from Song Lei Commercial Building to Peace District exhibits the lowest demand during this time. This is because the staff of Triumph Square go home at noon and leave many parking spaces open, whereas the opposite occurs in Peace Community.

The parking demands of Le Song Plaza cannot be allocated to Peace Community, which may be the cause of this phenomenon. The main time intervals when the demands of Song Lei Commercial Building are allocated to Peace Community are 10:30–12:00 and 14:00–16:00 since the residents in the community drive more during this time, thereby leaving many lots vacant.

Figure 7 shows the results of the sharing plan between Song Lei Commercial Building and Triumph Square. Cars drive from Song Lei Commercial Building to Triumph Square Parking all the time after 11:00, except 14:00-15:00. A possible explanation is that the staff of Song Lei Commercial Building return to their workplace from 13:00-14:00 and they prefer to walk across the street compared to driving through the street.

For travelers, they tend to pay more attention to travel time. Walking distance after parking commonly becomes an important factors considering by drivers. Figures 8–10 show the total walking distance among three distribution plans. For the distribution plan of Le Song Plaza and Triumph Square, the distance increases dramatically from about 1.5 km to 6.6 km at 13:30, and then steadily decreases to around 0.1 km at 15:00. From 15:00 to 17:00, the distance reaches a maximum value at 16:30. And for the distribution plan of Song Lei commercial building and Triumph Square, the changes of distance are similar as in Figure 8. This phenomenon indicates that the number of parking lots provided by Triumph Square is much more during 13:00 to 14:00 than other time. Therefore, the operator can transfer a large amount of parking requests in these two commercial zones to Triumph Space in order to ease the congestion of commercial zones. Compared to other distributions, the walking distance between Song Lei commercial building and Peace Community shows lowest during 13:00 and 14:00.

As discussed previously, distances that exceed 350 m were eliminated from the study data. The average walking distances are summarized in Figure 11. It denotes the ratio of the total walking distance to the requests. The average walking distance was approximately 250 m, and most of these distances were shorter than 300 m, which illustrates that the empirical application is successful. In other words, in the case of ensuring the walking distance, the shared parking distribution plan can effectively improve the utilization rate of parking lots in residential zones and alleviate congestion in commercial zones. In addition, most of the

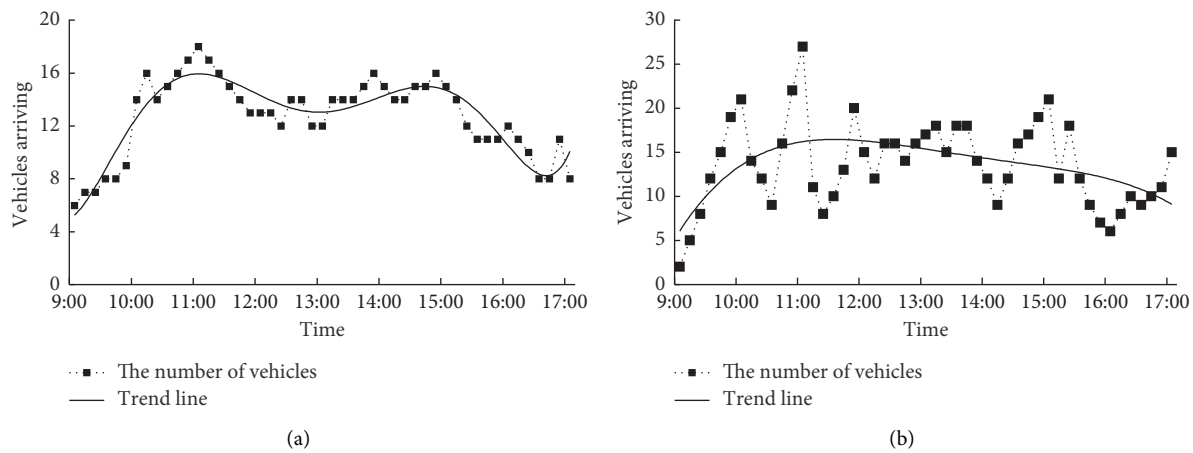


Figure 4: The changes of vehicle arriving between commercial zones. (a) Le Song Plaza. (b) Song Lei commercial building.

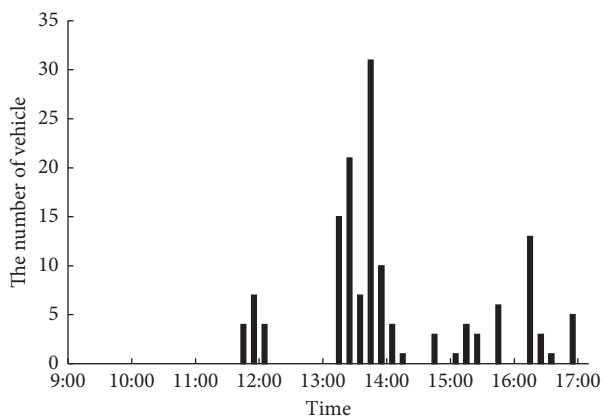


Figure 5: Distribution plan between Le Song Plaza Commercial Building and Triumph Square.

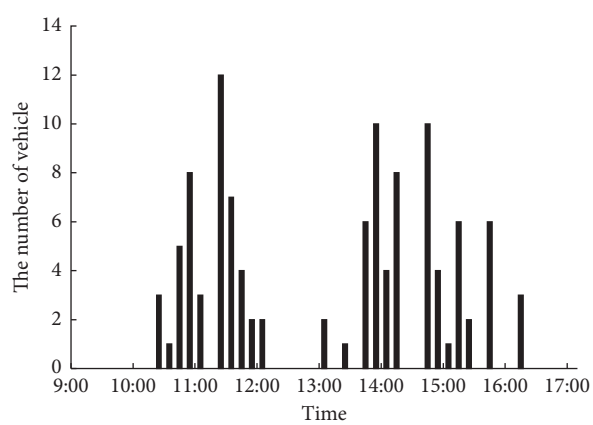


Figure 6: Distribution plan between Song Lei Peace Community.

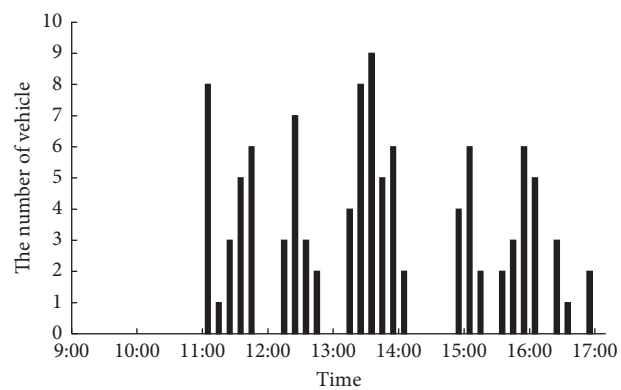


Figure 7: Distribution plan between Song Lei Commercial Building and Triumph Square.

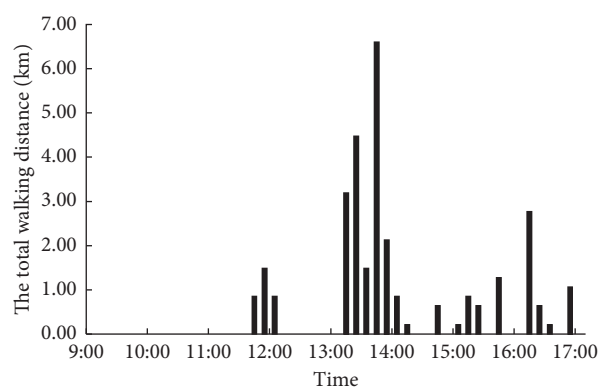


Figure 8: Total walking distance after parking from Le Song Plaza to Triumph Square.

walks will be less than 3.5 min in duration at the typical speed of an adult (1.2 m/s). Note that time spent waiting for the traffic lights at intersection is not included. As a result, the walking time may be only 1/5 or less of the driver's time for finding spaces.

Figure 12 shows the utilizing rate of parking lots in Triumph Square when using the distributing algorithm. The utilization rate of parking lots tends to be 10% higher when we apply the distributing algorithm, especially during 14:00 to 16:00. Figure 13 shows the utilizing rate of parking lots in Peace Community when using the distributing algorithm.

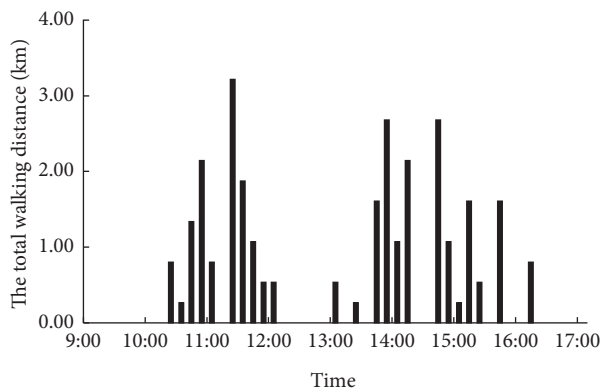


Figure 9: Total walking distance after parking from Song Lei Commercial Building to Peace Community.

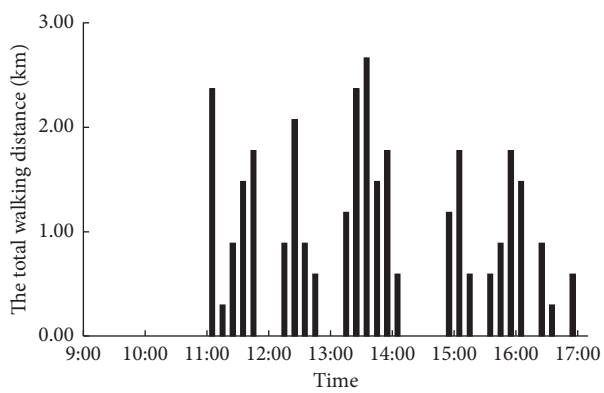


Figure 10: Total walking distance after parking from Song Lei Commercial Building to Triumph Square.

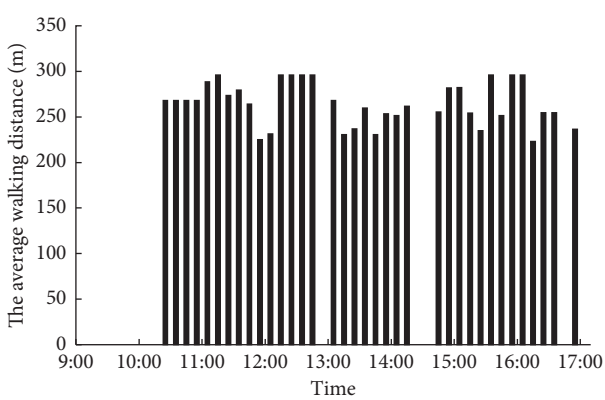


Figure 11: Average walking distance after parking.

Similarly, the utilization rate of parking lots increases about 10%.

It can be seen from the Figures 12 and 13, the utilization rates of parking lots in both residential zones and commercial zones have temporal characteristics. Using the sharing plan and distributing algorithm, the vacant parking lots are taken full use. Meanwhile, the parking demands in commercial zones are alleviated accordingly.

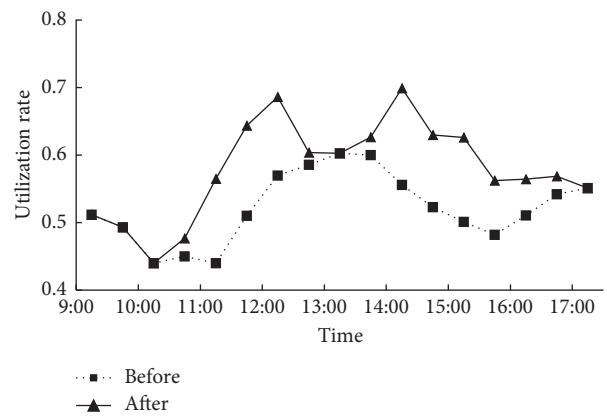


Figure 12: Utilizing rate of parking lots in Triumph Square.

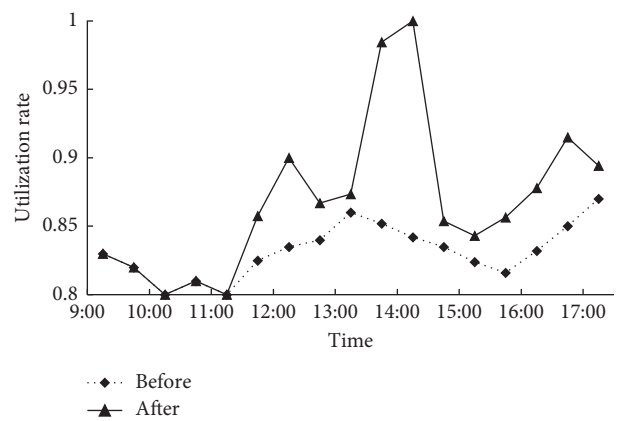


Figure 13: Utilizing rate of parking lots in Peace Community.

5. Conclusions

This paper analyzes shared parking allocation problems between parking demands in commercial buildings and parking supplies in residential zones. The concept of shared parking is proposed, which is according to the preconditions of shared parking implementation. Then, the feasibility of shared parking between parking requests from commercial buildings and private paid or public free parking lots in residential zones is initially evaluated by analyzing the characteristics of shared parking, which include win-win, convenience, economy, and real-time performance. Next, a bitarget parking spaces allocating model involving the minimum walking distance and the maximum utilization is proposed. The model comprehensively considers the drivers' walking distance and the utilization of parking spaces. It not only receives reception requests for buildings in commercial zones, but also assigns them to corresponding vacant parking lots in accordance with the model hypothesis and parking space-time constraints. PSO algorithm is applied to solve the parking allocation model. Finally, a numerical simulation is conducted to determine whether the allocation model can feasibly realize shared parking in residential zones by collected data. The results demonstrate that the paid or free use of the vacant parking lots in the residential zones can

effectively attract parking requests from the surrounding commercial buildings. The proposed model can increase the utilization of parking spaces in residential zones and decrease the walking distance after parking.

The shared parking scheme in residential areas has demonstrated potential value because it can reduce traffic congestion, exhaust gas pollution, and time wasting due to parking difficulties at nearly zero cost. Although not thoroughly studied in this paper, the commercial value (economic profits) of the shared schemes, against the background of fast-paced lifestyles and Internet technology, cannot be ignored. In the future, the shared parking strategy will be combined with the customer-to-customer (C2C) model to realize a win-win outcome for drivers, parking lot owners, and the platform. Profits can be maximized on the basis of the original multi-objective allocation model, namely, the driver spends the minimal amount on parking fees, the parking space owner earns the maximal amount of income, and the shared platform obtains the maximal profit. The shared parking scheme did not consider drivers' choice behaviours. Thus, the validity of this multiobjective model can be improved by incorporating the drivers' subjective selection.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare no conflict of interest.

Authors' Contributions

Wenhui Zhang designed the structure of paper and studied the literature. Junjun Tang wrote and revised the paper. Fan Gao established the model and conducted numerical simulation. Shurui Sun and Qiuying Yu collected parking data and analyzed parking demand and supplement. Bohang Liu proposed the methodology and solving algorithm.

Acknowledgments

The research was funded by the Fundamental Research Funds for the Central Universities (2572018BG01), Open Subject of Traffic Safety and Control Lab in Hebei Province (JTKY 2019003), National Natural Science Foundation of China (71701215), Foundation of Central South University (502045002), Postdoctoral Science Foundation of China (2018M630914 and 2019T120716), and National Key Research and Development Projects (2017YFC0803901).

References

[1] Q. Su and D. Z. W. Wang, "Morning commute problem with supply management considering parking and ride-sourcing," *Transportation Research Part C: Emerging Technologies*, vol. 105, pp. 626–647, 2019.

[2] S. M. Dave, G. J. Joshi, K. Ravinder, and N. Gore, "Data monitoring for the assessment of on-street parking demand in CBD areas of developing countries," *Transportation Research Part A: Policy and Practice*, vol. 126, pp. 152–171, 2019.

[3] W. Liu, H. Yang, and Y. Yin, "Expirable parking reservations for managing morning commute with parking space constraints," *Transportation Research Part C: Emerging Technologies*, vol. 44, pp. 185–201, 2014.

[4] F. Caicedo, "Real-time parking information management to reduce search time, vehicle displacement and emissions," *Transportation Research Part D: Transport and Environment*, vol. 15, no. 4, pp. 228–234, 2010.

[5] S. Dale, M. Frost, S. Ison, M. Quddus, and M. P. Warren, "Evaluating the impact of a workplace parking levy on local traffic congestion: the case of Nottingham UK," *Transport Policy*, vol. 59, pp. 153–164, 2017.

[6] C. G. Hoehne, M. V. Chester, A. M. Fraser, and D. A. King, "Valley of the sun-drenched parking space: the growth, extent, and implications of parking infrastructure in Phoenix," *Cities*, vol. 89, pp. 186–198, 2019.

[7] J. Wang, X. Zhang, and H. M. Zhang, "Parking permits management and optimal parking supply considering traffic emission cost," *Transportation Research Part D: Transport and Environment*, vol. 60, pp. 92–103, 2018.

[8] J. Wang, X. Zhang, H. Wang, and M. Zhang, "Optimal parking supply in bi-modal transportation network considering transit scale economies," *Transportation Research Part E: Logistics and Transportation Review*, vol. 130, pp. 207–229, 2019.

[9] M. Nourinejad and M. J. Roorda, "Impact of hourly parking pricing on travel demand," *Transportation Research Part A: Policy and Practice*, vol. 98, pp. 28–45, 2017.

[10] S. X. Xu, M. Cheng, X. T. R. Kong, H. Yang, and G. Q. Huang, "Private parking slot sharing," *Transportation Research Part B: Methodological*, vol. 93, pp. 596–617, 2016.

[11] X. Yan, J. Levine, and R. Marans, "The effectiveness of parking policies to reduce parking demand pressure and car use," *Transport Policy*, vol. 73, pp. 41–50, 2019.

[12] J. Chen, Z. Li, H. Jiang, S. Zhu, and W. Wang, "Simulating the impacts of on-street vehicle parking on traffic operations on urban streets using cellular automation," *Physica A: Statistical Mechanics and Its Applications*, vol. 468, pp. 880–891, 2017.

[13] N. Levy, M. Render, and I. Benenson, "Spatially explicit modeling of parking search as a tool for urban parking facilities and policy assessment," *Transport Policy*, vol. 39, pp. 9–20, 2015.

[14] Y. Xue, H. Fan, and H. Guan, "Commuter departure time choice considering parking space shortage and commuter's bounded rationality," *Journal of Advanced Transportation*, vol. 2019, Article ID 4864287, 7 pages, 2019.

[15] J. J. Soto, L. Márquez, and L. F. Macea, "Accounting for attitudes on parking choice: an integrated choice and latent variable approach," *Transportation Research Part A: Policy and Practice*, vol. 111, pp. 65–77, 2018.

[16] K. Shaaban and A. Pande, "Classification tree analysis of factors affecting parking choices in Qatar," *Case Studies on Transport Policy*, vol. 4, no. 2, pp. 88–95, 2016.

[17] J. K. Brueckner and S. F. Franco, "Employer-paid parking, mode choice, and suburbanization," *Journal of Urban Economics*, vol. 104, pp. 35–46, 2018.

[18] M. Zargayouna, F. Balbo, and K. Ndiaye, "Generic model for resource allocation in transportation: application to urban parking management," *Transportation Research Part C: Emerging Technologies*, vol. 71, pp. 538–554, 2016.

[19] M. Kaspi, T. Raviv, and M. Tzur, "Parking reservation policies in one-way vehicle sharing systems," *Transportation Research Part B: Methodological*, vol. 62, pp. 35–50, 2014.

[20] M. Kaspi, T. Raviv, M. Tzur, and H. Galili, "Regulating vehicle sharing systems through parking reservation policies: analysis and performance bounds," *European Journal of Operational Research*, vol. 251, no. 3, pp. 969–987, 2016.

[21] C. Shao, H. Yang, Y. Zhang, and J. Ke, "A simple reservation and allocation model of shared parking lots," *Transportation Research Part C: Emerging Technologies*, vol. 71, pp. 303–312, 2016.

[22] F. Alemi, C. Rodier, and C. Drake, "Cruising and on-street parking pricing: a difference-in-difference analysis of measured parking search time and distance in San Francisco," *Transportation Research Part A: Policy and Practice*, vol. 111, pp. 187–198, 2018.

[23] M. Alam, D. Moroni, G. Pieri et al., "Real-time smart parking systems integration in distributed ITS for smart cities," *Journal of Advanced Transportation*, vol. 2018, Article ID 1485652, 13 pages, 2018.

[24] A. El-Zonkoly and L. dos Santos Coelho, "Optimal allocation, sizing of PHEV parking lots in distribution system," *International Journal of Electrical Power & Energy Systems*, vol. 67, pp. 472–477, 2015.

[25] H. Chai, R. Ma, and H. M. Zhang, "Search for parking: a dynamic parking and route guidance system for efficient parking and traffic management," *Journal of Intelligent Transportation Systems*, vol. 23, no. 6, pp. 541–556, 2019.

[26] A. Amer and J. Y. J. Chow, "A downtown on-street parking model with urban truck delivery behavior," *Transportation Research Part A: Policy and Practice*, vol. 102, pp. 51–67, 2017.

[27] J.-H. Shin and H.-B. Jun, "A study on smart parking guidance algorithm," *Transportation Research Part C: Emerging Technologies*, vol. 44, pp. 299–317, 2014.

[28] S. Butler, R. Graham, and C. H. Yan, "Parking distributions on trees," *European Journal of Combinatorics*, vol. 65, pp. 168–185, 2017.

[29] J.-H. Shin, H.-B. Jun, and J.-G. Kim, "Dynamic control of intelligent parking guidance using neural network predictive control," *Computers & Industrial Engineering*, vol. 120, pp. 15–30, 2018.

[30] J. Cao and M. Menendez, "Quantification of potential cruising time savings through intelligent parking services," *Transportation Research Part A: Policy and Practice*, vol. 116, pp. 151–165, 2018.

[31] P. waerden, H. Timmermans, and M. 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, 2017.

[32] C. Ma, R. He, and W. Zhang, "Path optimization of taxi carpooling," *PloS One*, vol. 13, no. 8, Article ID e0203221, 2018.

[33] C. Ma, W. Hao, F. Pan, and W. Xiang, "Road screening and distribution route multi-objective robust optimization for hazardous materials based on neural network and genetic algorithm," *PLoS One*, vol. 13, no. 6, Article ID e0198931, 2018.

[34] J. Tang, J. Hu, W. Hao, X. Chen, and Y. Qi, "Markov Chains based route travel time estimation considering link spatio-temporal correlation," *Physica A: Statistical Mechanics and its Applications*, vol. 2020, Article ID 123759, 13 pages, 2020.

[35] C. Ma, W. Hao, A. Wang, and H. Zhao, "Developing a coordinated signal control system for urban ring road under the vehicle-infrastructure connected environment," *IEEE Access*, vol. 6, pp. 52471–52478, 2018.

[36] J. Tang, X. Zhang, W. Yin, Y. Zou, and Y. Wang, "Missing data imputation for traffic flow based on combination of fuzzy neural network and rough set theory," *Journal of Intelligent Transportation Systems*, vol. 2020, pp. 1–16, 2020.

[37] A. Poole and A. Kotsialos, "Second order macroscopic traffic flow model validation using automatic differentiation with resilient backpropagation and particle swarm optimisation algorithms," *Transportation Research Part C: Emerging Technologies*, vol. 71, pp. 356–381, 2016.

[38] L. Li, L. Qin, X. Qu, J. Zhang, Y. Wang, and B. Ran, "Day-ahead traffic flow forecasting based on a deep belief network optimized by the multi-objective particle swarm algorithm," *Knowledge-Based Systems*, vol. 172, pp. 1–14, 2019.

[39] Y. Zhang, Q. Zhang, A. Farnoosh, S. Chen, and Y. Li, "GIS-based multi-objective particle swarm optimization of charging stations for electric vehicles," *Energy*, vol. 169, pp. 844–853, 2019.