

Journal of Advanced Transportation

Advances in Modelling and Data-Driven Optimisation of Urban Transport and Logistics

Lead Guest Editor: Erfan Hassannayebi

Guest Editors: Ehsan Nikbakhsh, Saeid Saidi, and Majid Eskandarpour





Advances in Modelling and Data-Driven Optimisation of Urban Transport and Logistics

Journal of Advanced Transportation

**Advances in Modelling and Data-Driven
Optimisation of Urban Transport and
Logistics**

Lead Guest Editor: Erfan Hassannayebi





Guest Editors: Ehsan Nikbakhsh, Saeid Saidi, and
Majid Eskandarpour



Copyright © 2022 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 Data-Driven Functional Classification of Urban Roadways Based on Geometric Design, Traffic Characteristics, and Land Use Features

Mostafa Mehdian, Hamid Mirzahosseini , and Ali Abdi Kordani 
Research Article (9 pages), Article ID 9970464, Volume 2022 (2022)




A Hybrid Machine Learning and Optimization Model to Minimize the Total Cost of BRT Brake Components

Saeed Najafi-Zangeneh , Naser Shams Gharneh , Ali Arjomandi-Nezhad , and Erfan Hassannayebi 
Research Article (11 pages), Article ID 5590780, Volume 2021 (2021)

Evaluation and Analysis Model of the Length of Added Displaced Left-Turn Lane Based on Entropy Evaluation Method

Binghong Pan , Jinfeng Ying , Shasha Luo , Yang Shao , Shangru Liu , Xiang Li, and Zhenjiang Xie 
Research Article (22 pages), Article ID 2688788, Volume 2021 (2021)




Short-Term Traffic Flow Prediction: A Method of Combined Deep Learnings

Chuanxiang Ren , Chunxu Chai, Changchang Yin , Haowei Ji, Xuezhen Cheng , Ge Gao, and Heng Zhang
Research Article (15 pages), Article ID 9928073, Volume 2021 (2021)





Linear Programming Model and Online Algorithm for Customer-Centric Train Calendar Generation

Tommaso Bosi , and Andrea D'Ariano 
Research Article (18 pages), Article ID 4664010, Volume 2021 (2021)


A Balanced Strategy for the FFBS Operator Integrating Dispatch Area, Route, and Depot Based on Multimodel Technologies

Qingfeng Zhou , Jun Zhou , and Chun Janice Wong 
Research Article (16 pages), Article ID 6637251, Volume 2021 (2021)

Calibrating Path Choices and Train Capacities for Urban Rail Transit Simulation Models Using Smart Card and Train Movement Data

Baichuan Mo , Zhenliang Ma , Haris N. Koutsopoulos , and Jinhua Zhao 
Research Article (15 pages), Article ID 5597130, Volume 2021 (2021)

Profit Maximization Model with Fare Structures and Subsidy Constraints for Urban Rail Transit

Qing Wang , Paul Schonfeld , and Lianbo Deng 
Research Article (14 pages), Article ID 6659384, Volume 2021 (2021)

Research Article

A Data-Driven Functional Classification of Urban Roadways Based on Geometric Design, Traffic Characteristics, and Land Use Features

Mostafa Mehdian, Hamid Mirzahosseini , and Ali Abdi Kordani 

Department of Civil-Transportation Planning, Faculty of Technical and Engineering,
Imam Khomeini International University (IKIU), Qazvin 34149, Iran

Correspondence should be addressed to Hamid Mirzahosseini; mirzahosseini@eng.ikiu.ac.ir

Received 24 April 2021; Accepted 3 January 2022; Published 11 March 2022

Academic Editor: Alessandro Severino

Copyright © 2022 Mostafa Mehdian 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 functional classification system (FCS) of roads means categorizing roads based on their service characteristics. The two primary considerations in classifying highway and street networks are accessibility and mobility, where by increasing the role of one, the other's role is reduced. In this paper, besides the conventional variables such as geometric design characteristics, parking lots, land use features, and accessibility; the Sydney Coordinated Adaptive Traffic System (SCATS) data following the real-time traffic flow and average speed of vehicles collected by Location-Based Services (LBS) are considered as new variables for estimating the FCS. Linear regression is used to model the importance of the variables. The chi-square test compared the observational and predicted speeds in the five categories of roads in Tehran, the capital of Iran. Results show that on-street parking has the highest impact and the land use variable has the lowest impact on speed that changes the FCS. Moreover, the presented classification was one to two categories compared with the conventional FCS presented in manuals in the case of Tehran's transportation network as a developing city.

1. Introduction

Various functional classification systems (FCS) are defined based on different criteria including road geometry design, volume and type of traffic, and origin-destination (OD) of trips. These are beneficial in developing the standard road network. Roads can have a functional classification that expresses their functional importance in the whole network. Classifying volume and type of roads can be presented as a result of traffic assignment. Moreover, sometimes an environmental classification could be considered heavy vehicles and transportation's harmful impacts on the environment [1].

According to the two types of services that roads carry out, two main criteria are used to classify them, which are accessibility and mobility level. In fact, these are two criteria by which different types of roads are created based on changes in each of them [2]. The two criteria of accessibility

and mobility are inversely related, so increasing accessibility means reducing mobility and vice versa.

The conflict between providing mobility for traffic movements and spreading origins and destinations in a city requires countless roads with different functions to respond to the generated demand [3]. The leading purpose of the highway is to provide mobility that is defined at different levels. For example, mobility on the highway can mean riding comfort and privation of speed changes. However, due to the importance of sustainable transportation, attention to accessibility has become more important in these days and it can be used as an indicator to define the reliability of services in the transportation system [4]. Of course, it should be noted that access is not the only factor in achieving sustainable transport, and the public transport network must also support it [5]. In this regard, balancing mobility based on accessibility concerns is essential for managing the urban traffic network.

Many factors describe accessibility. Generally, it consists of traffic characteristics and geometric design features such as volume, angle of access, the distance of access points from each other, access control and access management, turning radius of the access point, entering speed from the access point to the roadway, location of bus and taxi stations, total ramp density, number of lanes, lane widths, type of median, on-street parking, the distance between intersections, deceleration and acceleration lanes, and types of land use [3, 6].

In addition to those conventional attributes, we consider more recent data source including Location-Based Services (LBS) and Sydney Coordinated Adaptive Traffic System (SCATS) to investigate the accurate classification of roadways by fusion of their data to conventional on-field attributes. LBS are applications on mobile portable devices (e.g., smartphones) that provide information depending on the location of the device and the user through mobile networks [7, 8]. Rapid advances in LBS with the continuous evolution of mobile devices and telecommunication technologies were presented just in a few years. Thus, LBS became more popular in outdoor and indoor environments (shopping malls, museums, airports, and big transport hubs). Moreover, LBS was applied in services like emergency services, tourism services, navigation guidance, intelligent transportation systems (ITS), entertainment (gaming), assistive services, healthcare/fitness, and social networking [9–11]. This service could be used as a tool for investigating and comparing traffic patterns [12], evaluating the Origin-Destination (OD) trips [13], and even verifying the accuracy of conventional traffic assignment methods [14]. In addition to this data, traffic volumes are gathered by installed detectors based on SCATS. SCATS is a traffic management system designed to optimize traffic flow [15] and metering the internal/external traffic during the rush hour to minimize the queue lengths at intersections [16].

2. Material and Methods

In this study, the variables extracted from the accessibility were selected as independent variables, and the effect of these variables on the spot speed was evaluated using linear regression. Furthermore, the data of real-time traffic characteristics have been used in addition to the conventional data like roadway's geometric design features. The roadway data for geometric design features are the angle of access, total ramp density, number of lanes, on-street parking, deceleration and acceleration lanes, and types of land usage. In addition, the traffic characteristics data for the roadway are spot speed and volume. In this study, each independent variable's effect on the dependent variable is measured using linear regression, and the effects of all variables on each other are measured by using the correlation matrix. Then the observed and predicted speeds were compared by the chi-square test.

2.1. Process. Generally, this study's collected data were geometric design features and traffic characteristics by considering roadside land use. In the first step, the functional systems for urban areas used in the American Association of

State Highway Officials (AASHTO) were selected. The four functional highway systems for urban areas used in conventional functional classification are principal arterial, minor arterial, collector streets, and local streets [3]. Functional systems for urban areas are schematically shown in Figure 1.

According to the classification of Figure 1, for each of the functional systems, roadways of Tehran were selected as a case study, which is shown in Figure 2 that is gathered from Tehran Traffic Maps (<https://map.tehran.ir>). For the principal arterial, Ayatollah Hashemi Rafsanjani (Niyayesh) was selected (Figure 2(a)). For the minor arterial, Resalat Expressway was selected (Figure 2(b)). In Niyayesh and Resalat expressways, both directions were selected, west to east and east to west. For the collector, two case studies were selected. One of them was Mofatteh Street (Figure 2(c)), and the other one was Motahhari Street (Figure 2(d)). The case study selected for the local street was Mehrdad Street (Figure 2(e)). There are 360 access points in case studies that affected accessibility.

In the next step, each of these roadways was divided into ten segments with equal length. The number of segments in the case study is 61. And then, the required data were collected from the available segments. Finally, modeling was done by selecting the appropriate regression model, and its results were extracted.

Angles of access data were obtained with AUTO CAD, CIVIL 3D, and ENGAGE DIGITIZER software. The collected number of angles is equal to the number of access points. The geometric design features of the roadways, like the number of lanes and on-street parking, are gathered by observation of field studies and checked by Tehran Traffic Maps (<https://map.tehran.ir>). The deceleration and acceleration lanes data were collected by referencing AASHTO standards and the field study. The ArcGIS software has been used to collect the land use data based on Tehran's spatial land use data. The collected data, deceleration and acceleration lanes, and land use data are qualitative, and other research data have been quantitatively and numerically introduced in the model.

3. Theory/Calculation

Highways' classification methods, Location-Based Service applications, and real-time traffic characteristics are used to find the proposed model in this section.

3.1. Highways' Classification. Highways classify into different operating systems based on functional classes or geometric types. Functional classification, the grouping of highways by the service they provide, was developed for transportation planning purposes. The FCS provides the starting point for assigning highways to different access categories. FCS is applied to categorize streets and highways according to their role [3, 6].

The urban roads have six primary roles:

- (i) Providing mobility for motor vehicles (mobility role).

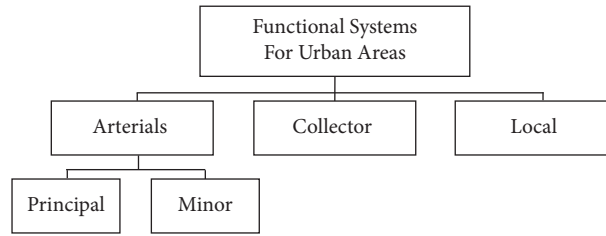


FIGURE 1: Functional system for urban areas schematic diagram.



FIGURE 2: Location of case studies-Tehran Traffic Maps (<https://map.tehran.ir>).

- (ii) Providing access to motor vehicles and facilities (accessibility role).
- (iii) Creating a platform for social communication such as working, traveling, playing, and meeting (social roles).
- (iv) Formation of urban architecture (urban architectural role).
- (v) Impact on the environment weather surrounding the road (climate effect role).
- (vi) Impact on city economics (economic role).

Roads usually take more than one role, and some of these roles conflict with another one. The role of mobility can be measured by the speed and amount of traffic volume. In general, in the six parts, the three roles of mobility, accessibility, and social role are the main criteria for calculating urban roads.

Various classification schemes have been applied for distinct purposes in different rural and urban regions. This research will examine whether traffic characteristics such as volume and speed corresponding to it can be combined with other geometric properties and use statistical methods such as regression to produce a favorable result in urban areas' functional systems. The questions are the following: Is there a meaningful relationship between each of the independent and dependent variables? Is there a simultaneous effect of independent variables with the decreasing effect of the dependent variable's speed value?

Moreover, the assumptions used in this research are as follows:

- (i) The minimum width of the lane is 3.67 m.
- (ii) All vehicles in the traffic flow are passenger cars.
- (iii) The inclement weather conditions are ignored.
- (iv) The level of service in roadways is not *E* or *F*.

The American Institute of Architects (AIA), with ten classes, includes all streets within a city or town based on differing degrees of suitability for traffic movement, pedestrian activity, and building types [17]. The proposed system is shown in Table 1.

In road classification research, Qin et al. [19, 20] presented a straightforward and yet accurate methodology named speed-independent road classification strategy (SIRCS). This method is based on the sole measurement of unsprung mass acceleration. The framework was proposed with two phases named offline and online. In the offline phase, in two stages, the transfer function from acceleration to mobility is first formulated. The frequency range based on the random forest is then classified according to the ISO 8608 road standard definition. In the online phase, first, the mass acceleration and velocity of the vehicle are combined to calculate the appropriate road profile in the area. The second step is to classify the two-stage road mobility based on the power spectral density criterion (PSD) [20]. In this paper, the harmony superposition function generates the road profile in the time domain based on [19, 20]

TABLE 1: AIA street classification system [18].

Classification	Scale	Speed	Location	Specific feature
Highway	Long-distance	Medium	Open country	Free of intersections, driveways, and adjacent buildings
Boulevard	Long-distance	Medium	Urbanized area	Buildings line, expansive parking, and sidewalk inside and planting trees in center
Avenue	Short-distance	Medium	Urban area	Ends with a significant building or monument
Drive	Edge of the urban area and beside of natural zone	Medium	Along a waterfront, park, or headland	One side of the drive, boulevard, with sidewalk and buildings, while the other has the qualities of a parkway, with naturalistic planting and rural detailing
Street	Small-scale	Low	Access to higher density areas like business zones or rowhouses	Raised curbs, wide sidewalks, closed drainage, parallel parking, trees in individual planting areas, and buildings aligned on short setbacks
Road	Small-scale	Low	Frontage of low-density buildings such as houses	Rural landscape with open areas, plantings and narrow sidewalks
Alley	Narrow access route	—	Servicing the rear of buildings on a street	Usually paved to their edges, with center drainage via an inverted crown
Lane	Narrow access route	—	Access to houses' backyard	Useful for accommodating utility runs, enhancing the privacy of rear yards, and providing play areas for children
Passage	Narrow, pedestrian-only connector	—	Cutting between buildings	Access from the middle of long blocks and connect frontage and backyard of blocks
Path	Narrow pedestrian and bicycle connector	—	A park or the open country	Emerge from the sidewalk network, necessary along highways but not required to supplement boulevards, streets, and roads

$$X_r(t) = \sum_{u=1}^U \sqrt{2 \cdot G_q(f_{\text{mid}-u}) \cdot \frac{f_2 - f_1}{U}} \sin(2\pi f_{\text{mid}-u} t + \Phi_{u-l}), \quad (1)$$

where U is the total number of the time-frequency components; $f_{\text{mid}-u}$ is u^{th} middle frequency in Hz, $G_q(f_{\text{mid}-u})$ is the PSD of $f_{\text{mid}-u}$ in m^3 ; Φ_{u-l} is the independent and identically distributed (IID) random phase over $(0, 2\pi)$. f_1 and f_2 are 0.33 Hz and 28.3 Hz, respectively [19].

Adafer and Bensaibi [21] developed a methodology based on determining a numerical indicator called the Vulnerability Index (VI). The vulnerability index also can be used for seismic vulnerability assessment for roads. The main parameters are identified, especially on past Algerian earthquakes and worldwide seismic feedback experiences. To quantify the identified parameters and define an analytical expression of the "VI", Analytical Hierarchy Process (AHP) is used. According to the obtained Vulnerability Index value, the classification of road sections' seismic vulnerability is proposed. This study's analysis and evaluation are number of lanes, pavement type, height, compaction quality, slope, ground type, landslides potential, pavement conditions, and slope protection measures. According to the vulnerability index, landslides potential, pavement conditions, and the number of lanes were the essential components.

In another study, Friedrich [22] describes the general methodology of the German Guideline for Integrated Network Planning (GGINP). He presented the approach and its methodology, including the form of transportation networks and the characteristics of the network elements and showed some examples of applying it. Examples of such characteristics are alignment speed, number of lanes, and the control type at intersections. In this research, the relationship between travel time, length, connectivity function level,

and length in build-up areas or sensitive areas with impedance was investigated by the regression model. The relationship is [22]

$$w_l = (\beta_0 + \beta_1 \cdot CFL_l + \beta_2 \cdot b_l) \cdot t_l + \beta_3 \cdot s_l, \quad (2)$$

where w_l is the impedance of link l , t_l is the travel time of link l , s_l is the length of link l , CFL_l is the connectivity function level of link l ($0 \leq CFL \leq 5$), b_l is the share of link length in build-up areas or sensitive areas ($0 \leq b_l \leq 1$), β_0 is the parameter that describes the influence of travel time, i.e., the accessibility, β_1 is the parameter that describes the influence of the road hierarchy, i.e., the bundling of traffic flows, β_2 is the parameter that describes the influence of sensitive areas, i.e., the compatibility of environment, and β_3 is the parameter that describes the influence of length, i.e., the directness [22].

3.2. Location-Based Services (LBS). Compared with other traditional geographic information systems (GIS) and web mapping applications, LBS is more adaptable to the contents and presentation according to its users' context [23]. Thus LBS is more dynamic and more probable to develop other GIS applications and open many research questions beyond the scientific field of geographic information science (GIScience) [24]. Geopositioning smartphones have attracted new application development, which utilizes the user's location information to provide valuable services. These applications are called LBS applications [25].

LBS need infrastructure like an internet network that can provide positioning tools for trough mobile devices [26]. Today, the tool that can supply access to LBS is mobile devices that users can send requests and retrieve results through. LBS need applications that providers develop just for them. These applications would download and install mobile devices like Personal Data Assistants (PDAs),

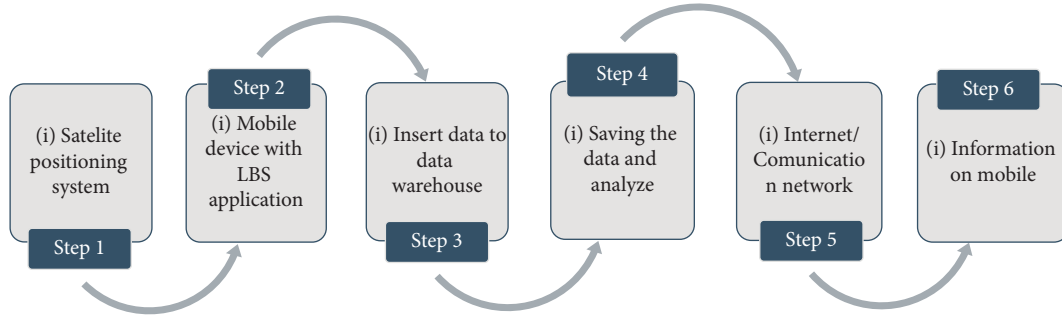


FIGURE 3: LBS process.

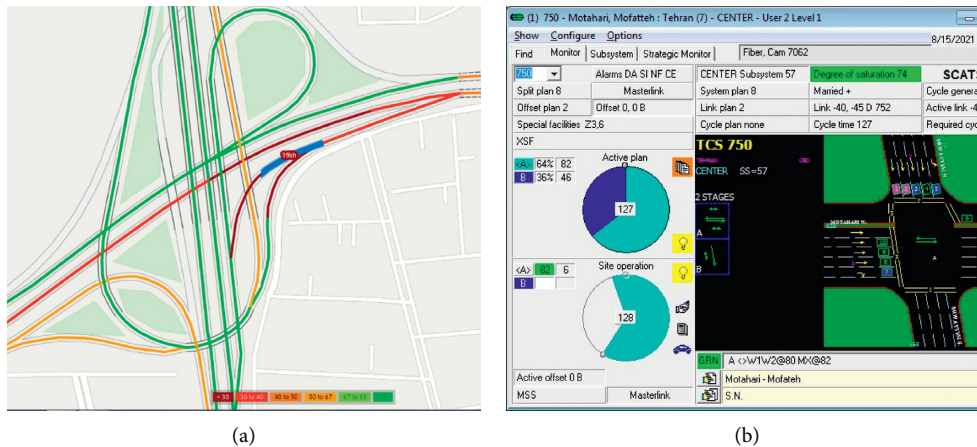


FIGURE 4: RISC Map View by Location-Based Service (LBS) data model (a), and SCATS software environment (b) for one sample (Motahhari-Mofatteh) used intersection.

laptops, and mobile phones [27]. Figure 3 shows the process that an LBS works. In step one, a request is sent through an application on a mobile device. Then in step two, requests and the user’s current location data is sent to the server. In steps three and four, the service server gets the necessary information from databases. Finally, in steps five and six, the required information is sent to the user [27].

In this research, eight independent variables and one dependent variable have been defined and counted. The independent variables used were: volume (maximum 15-minute count in hour), angle of access, total ramp density, number of lanes from the access point to the roadway, number of lanes in the roadway, on-street parking, deceleration and acceleration lanes, land use; and the dependent variables used in this research was spot speed as a traffic characteristic.

3.3. Traffic Characteristics. Traffic characteristics in this research included speed and volume data. These data were collected and recorded with the help of Tehran Traffic Control Company (TTCC) based on LBS. Global Positioning System (GPS) tracking data and SCATS data at intersections. LBS collected speed data. Tracking data calibrated by floating car and volume data was collected based on visually counted and rechecked by recorded video and SCATS data, especially for arterials with

intersections. An example of LBS’s collecting speed data, which Rajman Information Structures Company (RISC) developed, and volume data by SCATS software is shown in Figure 4.

The data collection was started on Monday, February 26, 2018, and was completed on Tuesday, May 28, 2019. These data were collected on Mondays and Tuesdays of each week between 9 a.m. and 11 a.m. for about more than one year in case of typical weather conditions. In this study, about 8600 data points were recorded for speed and volume by data sources.

3.4. Modeling. There are several ways to check the normality of speed data. In this study, skewness, kurtosis, and histogram, Kolmogorov-Smirnov, and Shapiro–Wilk tests were examined [28]. They confirmed the normality of the speed data. The histogram of speed for the normality test is shown in Figure 5.

With linear regression, we can estimate the linear equation’s coefficients and for this, we used one or more independent variables to calculate the dependent variable value [28]. The speed data, the dependent variable in this research, is quantitative and numerical and the used model is linear regression. SPSS software was used for modeling and a significant level of 95% was considered. A chi-square test was used to compare observational speed and predicted speeds. Finally, the relationship between all variables is shown in the correlation matrix.

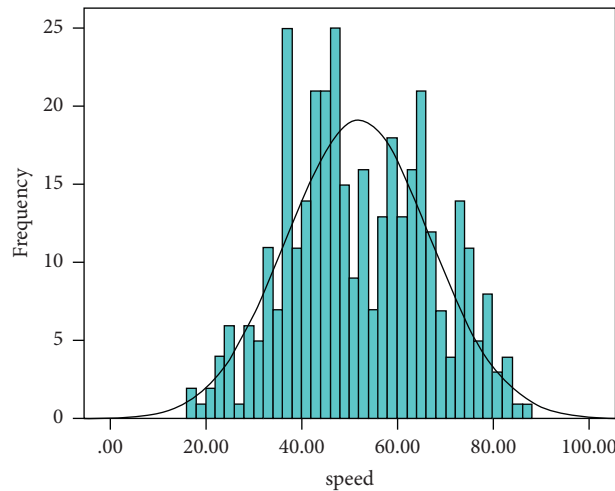


FIGURE 5: Histogram of gathered speed for normality test.

TABLE 2: Linear regression model, the effect of independent variables on the dependent variable separately.

Tests	Independent variable	Dependent variable	Standard regression coefficient (β)	Significant level (p -value < 0.05)
1	Angle of access	Spot speed	-0.49	Ok
2	Total ramp density	Spot speed	-0.25	Ok
3	Number of lanes from the access point to the roadway	Spot speed	-0.17	Ok
4	Number of lanes in the roadway	Spot speed	-0.014	Not ok
5	Volume	Spot speed	-0.15	Ok
6	On-street parking	Spot speed	-0.67	Ok
7	Deceleration and acceleration lanes	Spot speed	0.58	Ok
8	Land use	Spot speed	-0.51	Ok

TABLE 3: Linear regression model, the simultaneous effect of two or more independent variables on the dependent variable.

Tests	Independent variable	Dependent variable	Standard regression coefficient	R^2 adjust (%)
9	<ul style="list-style-type: none"> • Angle of access • Volume 	Spot speed	-0.48-0.116	26
10	<ul style="list-style-type: none"> • Angle of access • Volume • Total ramp density • Volume 	Spot speed	-0.46-0.091-0.12	27
11	<ul style="list-style-type: none"> • Total ramp density • On-street parking • Total ramp density 	Spot speed	0.10-0.12-0.64	47
12	<ul style="list-style-type: none"> • On-street parking • Deceleration and acceleration lanes • Total ramp density 	Spot speed	-0.13-0.48 0.24	50
13	<ul style="list-style-type: none"> • Angle of access • Volume • On-street parking 	Spot speed	-0.1-0.14-0.099-0.56	50

4. Results

Initially, the independent variables were fitted separately to the linear regression model's spot speed-dependent variable. The results are presented in Table 2.

Then, using the multiple linear regression analysis, the simultaneous effect of independent variables on speed was

measured. In this test, two to four independent variables' simultaneous effect is shown on the dependent variable. The number of independent variables causes an increasing number of prediction models. The results of the test are shown in Table 3.

In the final multiple regression analysis, which is the purpose of the research, the simultaneous effects of all

TABLE 4: Results of the final regression model.

Factors		Standard regression coefficient	<i>t</i>	<i>P</i> -value	VIF
Angle of access		$\beta = -0.059$	-1.13	0.26	1.9
Total ramp density		$\theta = -0.069$	-1.58	0.11	1.3
Number of lanes from access point to the roadway		$\gamma = 0.039$	-0.88	0.37	1.4
Number of lanes in roadway		$\delta = 0.072$	1.57	0.11	1.5
Volume		$\mu = -0.099$	-1.97	0.049	1.9
On-street parking		$\pi = -0.5$	-8.2	<0.001	1.2
Deceleration and acceleration lanes	0	—	—	—	—
	1	$\tau = -0.23$	-4.06	<0.001	2.2
Land use	Commercial or administrative	—	—	—	—
	Residential	$\rho = 0.019$	0.27	0.7	2.9

TABLE 5: ANOVA table.

Model	Sum of square	df	Mean square	<i>F</i>	Sig
Regression	42039.87	8	5254.98	46.97	<0.001
Residual	39269.41	351	111.87	—	—
Total	81309.28	359	—	—	—

$R^2 = 0.52$; R^2 adjust = 0.5; Durbin-Watson = 1.4

TABLE 6: The correlation matrix.

Correlation matrix	A	B	C	D	E	F	G	H	L
A	1	-0.49	-0.25	-0.66	-0.16	-0.01	-0.15	0.57	-0.51
B	-0.49	1	0.23	0.56	0.17	0.12	0.07	-0.65	0.47
C	-0.25	0.23	1	0.16	-0.12	-0.39	0.22	-0.14	0.22
D	-0.66	0.56	0.16	1	0.35	0.15	0.03	-0.65	0.64
E	-0.16	0.17	-0.12	0.35	1	0.42	-0.08	-0.3	0.22
F	-0.01	0.12	-0.39	0.15	0.42	1	-0.06	-0.21	0.13
G	-0.15	0.07	0.22	0.03	-0.08	-0.06	1	-0.07	0.53
H	0.57	-0.65	-0.14	-0.65	-0.3	-0.21	-0.07	1	-0.54
I	-0.51	0.47	0.22	0.64	0.22	0.13	0.53	-0.54	1

TABLE 7: Chi-square test based on design speed.

Functional systems	Local frequency (percent)	Collector frequency (percent)	Minor arterial frequency (percent)	Major arterial frequency (percent)	Total frequency (percent)	Chi-square test result
Observed	8 (2.25%)	109 (30.25%)	142 (39.45%)	101 (28.05%)	360 (100%)	$\chi^2 = 291.5$
Predicted	186 (51.67%)	45 (12.5%)	129 (35.83%)	0 (0%)	360 (100%)	$P < 0.001$

TABLE 8: Chi-square test based on permissible speed.

Functional systems	Local frequency (percent)	Collector frequency (percent)	Minor arterial frequency (percent)	Major arterial frequency (percent)	Total frequency (percent)	Chi-square test result
Observed	8 (2.25%)	109 (30.25%)	142 (39.45%)	101 (28.05%)	360 (100%)	$\chi^2 = 139.6$
Predicted	5 (1.38%)	225 (62.5%)	129 (35.82%)	1 (0.3%)	360 (100%)	$P < 0.001$

independent variables on speed are measured, and the final result is shown in (3) and Tables 4 and 5. The prediction rate of the model is 52%.

$$y = \alpha + \beta x_1 + \theta x_2 + \gamma x_3 + \delta x_4 + \mu x_5 + \pi x_6 + \tau x_7 + \rho x_8, \tag{3}$$

where y is the dependent variable, which is Spot speed, x_1 is the angle of access, x_2 is the total ramp density, x_3 is the

number of lanes from the access point to the roadway, x_4 is the number of lanes in the roadway, x_5 is the volume (maximum 15-minute count in an hour), x_6 is the on-street parking, x_7 is the deceleration and acceleration lanes, and x_8 is the land use.

The effect of all the variables used in this study on each other is shown in the correlation matrix. The correlation matrix is shown in Table 6. In this matrix, A is the spot speed, B is the angle of access, C is the total ramp density, D is the

on-street parking, E is the number of lanes from the access point to the roadway, F is the number of lanes in the roadway, G is the volume, H is the deceleration and acceleration lanes, and L is the land use.

To compare the criteria before and after modeling, chi-square test was used. A comparison of speed frequencies has been made based on the design speeds and permissible speed, and the results are shown in Tables 7 and 8.

5. Discussion and Conclusion

According to the tests, it can be concluded that the most influential independent variable is the on-street parking, which, due to the negative standard coefficient, has an inverse effect on the dependent variable. Among the independent variables, land use had the lowest impact on the dependent variable. The standard regression coefficient of this variable is positive. It had a direct impact on the dependent variable. In multiple linear regression models, the number of lane variables from the access point to the roadway, the number of lanes in the roadway, and land use were variables that directly correlated with other variables. The relationship of the dependent variable with other variables was indirect.

By analyzing the on-street parking data, an unexpected result was obtained. In the parts of the roadway where there was on-street parking but was not being used, the speed was increased compared to the previous one and one of the reasons is that users use these lanes as passing lanes.

The observational speeds before the study were compared with the predicted speeds after modeling using the chi-square test. According to Table 7, at first, these values were compared with the design speed, with the initial values for local streets equal to 8 (2.25%), for collector streets equal to 109 (30.25%), for minor arterial streets equal to 142 (39.45%), and for s , 101 (28.05%). After modeling and comparing with design speed, their values were as follows: 186 (51.67%) for local streets, 45 (12.5%) for collector streets, 129 (35.83%) for minor arterial streets, and zero for principal arterials. Then, these values were compared with the permissible speed. According to Table 8, the results of the chi-square test based on permissible speed are as follows: for local streets it is equal to 5 (1.38%), for collector streets it is equal to 225 (62.5%), for minor arterial streets it is equal to 129 (35.82%), and for principal arterials it is equal to one (0.3%).

The predicted and the Nash–Sutcliffe test measured observational speeds to measure the modeling's prediction accuracy. The Nash–Sutcliffe model efficiency coefficient value was obtained at 96.8%. This means that the predicted speeds were close to the observational speeds, and the regression model's prediction has been corrected. However, in Tehran, the actual function of the roadways differed from their nominal function. All speed numbers entered in the chi-square test were coded. Finally, only one number among the speeds was in the design speeds range and the principal arterials' permissible speeds.

According to the chi-square test results, two general results can be extracted: Firstly, in Tehran, as our case study,

each of the urban FCS does not match the current situation of traffic characteristics and geometric design features by considering roadside land use. The actual functional classification is remarkably different from the conventional functional classification named in the references. In most cases, the obtained classification was one to two categories less than the conventional nominal classification. Secondly, considering the SCATS and LBS data encode the real-time traffic flow and average speeds, it can be used as a new method to determine the more accurate FCS of each segment in the urban transportation network.

Data Availability

The datasets are available from the corresponding author on reasonable request.

Conflicts of Interest

There are no conflicts of interest.

References

- [1] S. Argyroudis, K. Pitilakis, and A. Anastasiadis, "Roadway network seismic risk analysis in urban areas: the case of Thessaloniki-Greece," in *Proceedings of the Geoline Conference*, Lyon, France, May 2005.
- [2] M. Malenkovska Todorova, R. Donceva, and J. Bunevska, "Role of functional classification of highways in road traffic safety," *Transport Problems*, vol. 4, pp. 97–104, 2009.
- [3] A. Book, "AASHTO Green Book 2016. AASHTO," January 2018, <https://www.fhwa.dot.gov/design/standards/151112.cfm>.
- [4] T. Nordfjærn, K. S. Egset, and M. Mehdizadeh, "Winter is coming": Psychological and situational factors affecting transportation mode use among university students," *Transport Policy*, vol. 81, pp. 45–53, 2019.
- [5] M. Mehdizadeh and A. Ermagun, "I'll never stop driving my child to school": on multimodal and monomodal car users," *Transportation*, vol. 47, no. 3, pp. 1071–1102, 2020.
- [6] AASHTO 6th Edition, L. R. F. D., *Bridge Design Specifications*, American association of state highway and transportation officials, Washington, DC, USA, 1998.
- [7] S. Gao and G. Mai, "Mobile GIS and location-based services," B. Huang, T. J. Cova, and M.-H. Tsou, *Comprehensive Geographic Information Systems*, pp. 384–397, Elsevier, Oxford, UK, 2018.
- [8] J. Raper, G. Gartner, H. Karimi, and C. Rizos, "A critical evaluation of location based services and their potential," *Journal of Location Based Services*, vol. 1, no. 1, pp. 5–45, 2007.
- [9] P. Sadhukhan, N. Mukherjee, and P. K. Das, "Location-based services for smart living in urban areas," in *Precision Positioning with Commercial Smartphones in Urban Environments*, pp. 53–69, Springer, Manhattan, NY, USA, 2021.
- [10] P. Kiefer, H. Haosheng, R. Martin, and V. D. W. Nico, *Progress in Location Based Services 2018*, Springer, Manhattan, NY, USA, 2018.
- [11] G. Gartner and H. Huang, *Progress in Location-Based Services 2016*, Springer, Manhattan, NY, USA, 2017.
- [12] I. Gholampour, H. Mirzahosseini, and Y.-C. Chiu, "Traffic pattern detection using topic modeling for speed cameras based on big data abstraction," *Transportation Letters*, pp. 1–8, 2020.
- [13] S. Afandizadeh Zargari, A. Memarnejad, and H. Mirzahosseini, "A structural comparison between the

- origin-destination matrices based on local windows with socioeconomic, land-use, and population characteristics,” *Journal of Advanced Transportation*, vol. 2021, Article ID 9968698, 17 pages, 2021.
- [14] H. Mirzahosseini, G. Iman, S. Maryam, and Z. Lei, “How realistic is static traffic assignment? Analyzing automatic number-plate recognition data and image processing of real-time traffic maps for investigation,” *Transportation Research Interdisciplinary Perspectives*, vol. 9, 2021.
- [15] S. A. Zargari, N. Dehghani, and H. Mirzahosseini, “Optimal traffic lights control using meta heuristic algorithms in high priority congested networks,” *Transportation Letters*, vol. 10, no. 3, pp. 172–184, 2018.
- [16] S. Afandizadeh Zargari, N. Dehghani, H. Mirzahosseini, and M. Hamedi, “Improving SCATS operation during congestion periods using internal/external traffic metering strategy,” *Promet - Traffic & Transportation*, vol. 28, no. 1, pp. 41–47, 2016.
- [17] F. Gerry, “Urban Roadway Classification: Before the Design Begins,” *Urban Street Symposium Conferences Proceedings*, 1999.
- [18] K. E. Hedges, *Architectural Graphic Standards*, John Wiley & Sons, Hoboken, NJ, USA, 2017.
- [19] Y. Qin, Z. Wang, C. Xiang, E. Hashemi, A. Khajepour, and Y. Huang, “Speed independent road classification strategy based on vehicle response: theory and experimental validation,” *Mechanical Systems and Signal Processing*, vol. 117, pp. 653–666, 2019.
- [20] Y. Qin, C. He, X. Shao, H. Du, C. Xiang, and M. Dong, “Vibration mitigation for in-wheel switched reluctance motor driven electric vehicle with dynamic vibration absorbing structures,” *Journal of Sound and Vibration*, vol. 419, pp. 249–267, 2018.
- [21] S. Azafer and M. Bensaibi, “Seismic vulnerability classification of roads,” *Energy Procedia*, vol. 139, pp. 624–630, 2017.
- [22] M. Friedrich, “Functional structuring of road networks,” *Transportation research procedia*, vol. 25, pp. 568–581, 2017.
- [23] S. Steiniger, M. Neun, and A. Edwardes, “Foundations of location based services,” *Lecture Notes on LBS*, University of Zurich, vol. 1, no. 272, Zurich, Switzerland, 2006.
- [24] H. Huang and G. Gartner, “Current Trends and Challenges in Location-Based Services,” *Multidisciplinary Digital Publishing Institute*, vol. 7, 2018.
- [25] D. Parmar and U. P. Rao, “Towards privacy-preserving dummy generation in location-based services,” *Procedia Computer Science*, vol. 171, pp. 1323–1326, 2020.
- [26] W. Schwinger, C. Grün, and B. Pröll, “A light-weight framework for location-based services,” in *OTM Confederated International Conferences “on the Move to Meaningful Internet Systems”* Springer, Berlin, Heidelberg, 2005.
- [27] A. Kushwaha and V. Kushwaha, “Location based services using android mobile operating system,” *International Journal of Advances in Engineering & Technology*, vol. 1, no. 1, p. 14, 2011.
- [28] G. A. Morgan, L. L. Nancy, W. G. Gene, and C. B. Karen, *IBM SPSS for Introductory Statistics: Use and Interpretation: Use and Interpretation*, Routledge, Oxfordshire, UK, 2019.

Research Article

A Hybrid Machine Learning and Optimization Model to Minimize the Total Cost of BRT Brake Components

Saeed Najafi-Zangeneh ¹, Naser Shams Gharneh ¹, Ali Arjomandi-Nezhad ²,
and Erfan Hassannayebi ³

¹Industrial Engineering Department, Amirkabir University of Technology, Tehran 15875-4413, Iran

²Industrial Engineering and Productivity Research Center, Amirkabir University of Technology, Tehran 15875-4413, Iran

³Department of Industrial Engineering, Sharif University of Technology, Tehran 14588-89694, Iran

Correspondence should be addressed to Erfan Hassannayebi; hassannayebi@sharif.edu

Received 3 March 2021; Revised 5 September 2021; Accepted 12 October 2021; Published 22 October 2021

Academic Editor: Dongjoo Park

Copyright © 2021 Saeed Najafi-Zangeneh 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.

Public transport is amongst critical infrastructures in modern cities, especially megacities, home to millions of people. The reliability of these systems is highly crucial for both citizens and service providers. If service providers overlook system reliability, a considerable amount of expenses will be wasted. Several factors such as vehicle failure, accident, lack of budget weather factors, and traffic congestion cause unreliability, among which vehicle failure plays a prominent role. The brake system is the most vulnerable and vital component of a public transportation bus. Brake reliability depends on driver's expertise, component quality, passenger loading, line situation, etc. Driver's expertise and components' quality are the most important factors for brake system reliability. This study aims to implement a hybrid machine learning and optimization model to minimize the total investment and reliability-related costs in a bus rapid transit (BRT) system. A regression analysis method is proposed to capture the main attributes of a joint brake system, including the level of education, training, and drivers' experience. The failure rate is modeled as a linear function of ETE and the quality of brake system subcomponents using a Lasso regression model. MILP optimization is then provided for optimizing the total expected costs for a bus rapid transit (BRT) system. Furthermore, a practical case is studied to investigate whether this optimization can reduce costs. The results confirm the efficiency of the hybrid optimization approach.

1. Introduction

Nowadays, cities are growing in size, and their populations are increasing rapidly. As citizens need to travel inside their cities more frequently, public transportation systems are getting ever-increasing importance in society. Many passengers travel by bus rapid transit (BRT), a left-side door bus operating in a fully separated lane. BRT reliability studies are pivotal because an interruption in such systems would result in passenger dissatisfaction and stakeholders would have to deal with vast economic losses. To overcome this challenge, the reliability of this transportation system is analyzed and then optimized. Reliability refers to the probability that a device performs its purpose adequately for the period intended under the operating conditions encountered [1]. A

high level of reliability would be an excellent incentive for citizens to choose public transport [2]. Several works analyzed in detail in the next section aim to quantify and enhance the reliability of urban bus systems as a backbone to public transport.

There are several reasons for BRT system irregularity, including suboptimal scheduling, accident, bus failure, etc. Based on the analysis of historical data, the main reason for BRT irregularity and latency is bus failures, which is due to brake failure in most cases. Not only is brake failure the primary reason for bus failure, but also it completely interrupts the bus. The driver cannot even take the bus to the repair shop. Therefore, brake component reliability optimization is vital in enhancing overall reliability. However, system owners have limited financial resources; therefore,

such enhancements should be constrained to available budgets and prospective future costs. To the best of our knowledge, this paper, for the first time, presents an analytical cost-benefit optimization for the brake system reliability considering the total costs imposed on the owners. At the first step, the brake failure rate is modeled as a function of subcomponent types and the education, training, and experience (ETE) indicators of the drivers. The former represents that a high-quality subcomponent lasts longer, while the latter represents the effect of driver skills in a better brake system. This is followed by modeling the primary investment, operation, and maintenance costs, including repair, replacement, HR training, and salary. Finally, convex mixed-integer linear programming (MILP) is provided to decide on the type of brake subcomponents of buses acquired for BRT lines and the ETE indicator of their drivers. The objective is to minimize the total cost, including investment, driver salaries, replacement costs, and economic loss due to bus interruptions and failures. To summarize, the main contributions of this paper are as follows:

- (i) Modeling the brake failure rate based on subcomponents and associated drivers
- (ii) Modeling various brake-related investments, operation, and maintenance costs
- (iii) Optimizing subcomponents and driver planning to minimize total costs

The rest of this paper is organized as follows: related research materials are reviewed in Section 2; the general structure of the proposed approach is briefly introduced in Section 3; in Section 4, the failure rate is modeled as a function of the ETE indicator and the brake system quality; the formulation of the optimization problem is discussed in Section 5; a case study for a practical BRT system is presented in Section 6; results and Section 7 presents results and discussion, and finally, the conclusion is drawn in Section 8.

2. Related Works

Several types of research in the literature concentrated on the reliability of the public transportation system. Those works either modeled or proposed reliability enhancement solutions using online or offline methods. In the following sections, those research works are reviewed.

2.1. Reliability Modeling and Quantification. The first stage in reliability studies is modeling and quantification. Public transportation services are categorized into two groups: frequency-based and scheduling-based [3]. While scheduling-based services operate on predefined schedules, only headway time is of interest within the system management in frequency-based services. Moreover, passengers are divided into two types: commuters, who regularly travel for business or education purposes, and noncommuters, who use public transport for occasional travels in that specific paths. Also, they analyzed the methods of computing headway and expected waiting time. Liu and Sinha [4] introduced three reliability metrics: “travel time reliability,” “headway

reliability,” and “passenger wait time reliability.” In [5], a set of reliability indicators from the viewpoint of customers are introduced. The latter expects that the indicators satisfy the four attributes “measurability,” “ease of availability,” “speed of availability,” and “interpretability,” in addition to being customer-oriented. In [6], the quality of service and transit reliability for older people (more than 65 years), counted as vulnerable users, are computed. A data-driven reliability study of public transportation for the Netherlands is presented in [7]. The automatic vehicle location (AVL) data are used for offline measuring time reliability in [8]. In [9], the percent of passengers receiving regular service (PPR) and percent of passengers receiving punctual service (PPP) using AVL data are computed. Three performance measures of punctuality index based on routes (PIR), deviation index based on stops (DIS), and evenness index based on stops (EIS) have been introduced and implemented for the Beijing transportation system in [10]. The probability that the public transport system performance is within the acceptable range for Beijing’s transport system reliability is computed. The impact of ridership on the reliability of the public transportation system is modeled in [11]. A review of all influential factors in reliability, in addition to reliability metrics, is briefly discussed in [12]. It divides the factors into two groups: demand-side factors, including traffic flow, passenger route-wise demand, and directional flow at inter-sections, and supply-side factors, including facility design, accidents, driver behavior, traffic management scheme, vehicle breakdown, and weather. Next, the reliability of the Ahmedabad city is computed using gathered GPS data. A methodology for estimating the value of travel time reliability is presented in [13]. Bunker [14] presented a probabilistic reliability model for sections (the distance between stops). Although that model investigates the financial aspects of reliability, it does not discuss the improvement strategy.

2.2. Reliability Enhancement. Optimal reliability enhancement can substantially reduce system expenses. Moosavi et al. [15] categorized reliability enhancement strategies into three main groups: prioritizing, operational, and control. Prioritization approaches are those that give priority to the public within the city. Dedicating a separate lane to buses is an example of prioritization policies. Operational strategies include long-term accomplishments such as driver training and restructuring of bus routes (offline methods). Control strategies are real-time decisions such as skipping stops (online methods). The impacts of various control strategies on transportation reliability are then simulated. The paper does not model the economic aspects of reliability. Therefore, the approach does not give the stakeholder a vision for the financial benefits of reliability enhancement. An analytical control strategy optimization is suggested in [16]. This approach ensures the global optimality of final results. AVL data are utilized in [17–19] to identify routes that need assistance and reliability enhancement. Wang et al. [20] proposed a data-driven bus scheduling optimization to enhance the reliability of transportation systems. In [21], how headway variations cause an extra cost to passengers

and how total cost (operator and user costs) will be optimized by bus stop placement and dispatching headway are discussed.

The papers mentioned above discussed the effect of path and control strategies on public transport reliability. However, for a BRT service in which a specific lane is dedicated to buses, the unwise path selection and suboptimal control strategies are not the primary cause of unreliability. These systems are highly reliable, especially in peak hours [22]. Breakdowns of buses are the main cause of interruptions and compromises to the service. A bus consists of several vulnerable components, of which brake systems are responsible for most interruptions. We checked the accuracy of this fact by comparing the reasons for a practical BRT system interruption. The historical data confirm that brake failure accounts for most of the buses' interruptions. Data analysis in [23] also confirms that the main factor that causes an urban bus being downtime in repair is the brake system failure. A fuzzy rule-based study of the Istanbul BRT system also indicates that the brake component is one of the vital components for retaining the reliability of the BRT system [24].

Furthermore, upon brake malfunctions, a severe risk is imposed on passengers and drivers. In this regard, the brake system functionality is also a key to safety [25]. Hence, analyzing the reliability of brake systems and their impact is an important subject.

Yusupov [25] presented a serial reliability model for brake systems. The reliability of the brake system is then computed based on the reliability of each subcomponent. A maximum likelihood estimation (MLE) method is presented in [26] for estimating the failure distribution of brake subcomponents. Moreover, the reliability of the brake system is computed using a fault tree. A Petri-net model for computing the reliability of mechatronic systems is presented in [27]. The critical reliability metrics, which are failure rate, mean time between failures, mean time to repair, and the brake system availability, are modeled in [28]. The shape of the brake piston ring is redesigned to improve the reliability in [29]. Yusupov et al. [30] first identified the subcomponents with the credible value of failure rates. Then, the relationship between these values and the brake component reliability was modeled. Finally, the method was simulated for the ABS brake system. None had studied the brake system reliability impact on the overall bus reliability.

To conclude, none of those above papers analyzed and optimized the brake system reliability as part of the whole. As a result, financial studies and the cost-benefit optimization for reliability enhancement are also missed. To fulfill this research gap, this paper presents a model for brake failure. The costs BRT systems endure due to brake failures are modeled, and the BRT system's total costs related to brake failures are optimized.

3. Proposed Methodology: Big Picture

To enhance the reliability of the brake components, first, the influential factor should be identified, as depicted in Figure 1. The drivers' expertise and subcomponents' quality

are discussed in Section 4. Experts can provide an approximately precise score for drivers and subcomponents. The relationship between the failure rate and these scores can be modeled with machine learning (ML). Increasing the score for these factors inevitably causes a decrease in the failure rate. However, this increase requires extra investment either in components or salary and training costs. Due to limitations in available budget, the total cost, covering reliability enhancement budget and interruption, and operation and management (O&M) costs should be optimized. Since O&M cost computation requires every subcomponent failure and replacement cost, the decomposition overall failure rate to subcomponent failure rate is necessary.

4. Failure Rate Model

To model the brake system failure rate, the features contributing failure rate value should be extracted. These are the features that must be modeled and qualified. The main reasons for brake system failure are low-quality brake components and careless drivers. Therefore, the features are the driver expertise score and the quality of the brake system. A machine learning model is then trained to estimate the failure rate based on these two features. These features are brake quality scores and drivers' expertise scores. Obviously, the better the quality of the brake and the more skillful the driver are, the less the failure rate of the brake component is. The following sections introduce both the features and model. This model is exploited inside our optimization problem in Section 5.

4.1. Brake Quality Score. Subcomponent types have a significant contribution to the failure rates of a component. The better the subcomponents, the longer the component survives. For example, the brake system is composed of several subcomponents, four of which are responsible for most failures: pedal, retarder, ABS, and pad. Each falls within one of the following quality bands: A (highest quality), B, and C (lowest quality), with scores of 15, 10, and 5, respectively. In the end, the sum of the subcomponent scores is scaled between 0 and 100.

4.2. Driver Expertise Score. Highly skilled drivers can better maintain and manage the brake system, and therefore, this can be regarded as an influential factor in calculating the failure rate. This paper introduces the ETE indicators representing the level of education, annual training hours, and total years of driving experience. To calculate ETE, the score of education level and experience are calculated according to Table 1. The values in Table 1 are based on the filled surveys. The training score is then calculated according to (1). In this formula, h_{\min} is the minimum hour of required training and S_h is the coefficient of training hour in ETE score. Finally, the sum of these three scores is scaled between 0 and 100.

$$\text{training_score} = S_h * (h - h_{\min}). \quad (1)$$



FIGURE 1: The general structure of the brake reliability-related cost minimization approach.

TABLE 1: Score table for education and years of experience.

Education level	Under high school	High school diploma	Associate degree	Bachelor	Master	Ph.D.
Score	2	3	4	6	8	10
Years of experience	0–5	5–10	10–15	15–20	20–25	25–30
Score	2	3	4	6	8	10

4.3. Failure Rate Model. Failure rate (f) is estimated as a function of ETE and brake quality score (Q) as shown in the following equation:

$$\hat{f} = g(\text{ETE}, Q). \quad (2)$$

There is no analytical formula that relates ETE and Q to the failure rate. Thus, a data-driven model is used instead. The g function is estimated using machine learning methods. Machine learning includes various models such as

linear regression, decision tree, and artificial neural network. In this paper, the *Lasso* method, which fits a linear function to an input-output relationship [31], is employed to model failure. The learner minimizes the mean square error between the actual and predicted output. To regularize the coefficients and prevent overfitting, a term of the first-order norm of the coefficients is added to the objective function according to (3) [31]. This trained linear failure rate model is later used in MILP cost optimization.

$$\text{Min} \left\{ \frac{1}{M} \sum_{m=1}^M [f_m - (\beta_0 + \beta_1 * ETE_m + \beta_2 * Q_m)]^2 + \lambda * (|\beta_0| + |\beta_1| + |\beta_2|) \right\}, \quad (3)$$

where β_1 and β_2 are the coefficients of ETE and Q in the linear fitted function, and λ is the regularization factor, which is a hyperparameter. Hyperparameters should be assigned a value before the training task. M is the total number of samples, and m is the index of samples. This optimization is solved via the scikit-learn package in the python programming language.

The brake component failure rate can be approximately decomposed into subcomponent failure rates by multiplying the failure rate with a fraction of the failure rate of that subcomponent.

5. Modeling and Optimizing Total Cost

This section presents the mathematical optimization model to minimize the investment and reliability-related costs in a BRT system under the risk of braking failure. The main idea is to optimize brakes and ETE factors before operating new buses in a BRT system. It is expected that the optimized operating plan could significantly improve the reliability of the operations and reduces the total cost of the BRT system. To do so, the main pillars of costs and constraints must first be identified. The total cost has four pillars:

- (i) Investment cost (IC): the amount of money used to buy subcomponents, subject to budget availability.
- (ii) Human resource cost (HRC): driver salaries and training costs, depending on driver education and experience.

(iii) Outage cost (OC): cost of an interruption in bus operations due to failure in brake components. This would undoubtedly incur costs as fewer passengers are served.

(iii) Replacement cost (RC): the cost of replacing a failed or damaged brake subcomponent with a new one.

The objective function is the sum of the investment, human resource, outage, and replacement costs, as asserted in equation (4). In the following sections, details of computing each cost and associated constraints are explained.

$$\text{Min}\{\text{IC} + \text{HRC} + \text{RC} + \text{OC}\}. \quad (4)$$

5.1. Investment Cost. The investment cost is the sum of subcomponent costs. Referring back to Section 4, the four subcomponents pedal, retarder, ABS, and pad are responsible for the majority of brake failures. The indices of p , r , a , and d are used as notations for the mentioned elements. The set of pedal types is symbolized as P , retarder types as R , ABS types as A , and pad types as D . Equation (5) represents the investment cost for N buses. The symbol $|\cdot|$ in this equation and successive equations refers to the size of the set. The binary variable $B(i,e)$ indicates whether a subcomponent of type e is bought for bus i . The parameter $C(\cdot)$ is the cost of subcomponents.

$$\text{IC} = \sum_{i=1}^N \left(\sum_{p=1}^{|P|} B(i,p) * C(p) + \sum_{a=1}^{|A|} B(i,a) * C(a) + \sum_{r=1}^{|R|} B(i,r) * C(r) + \sum_{d=1}^{|D|} B(i,d) * C(d) \right). \quad (5)$$

Since only one subcomponent type can be installed in a bus, constraints (6)–(9) should be satisfied.

$$\sum_{p=1}^{|P|} B(i,p) = 1 \quad \forall i, \quad (6)$$

$$\sum_{a=1}^{|A|} B(i,a) = 1 \quad \forall i, \quad (7)$$

$$\sum_{r=1}^{|R|} B(i,r) = 1 \quad \forall i, \quad (8)$$

$$\sum_{d=1}^{|D|} B(i,d) = 1 \quad \forall i, \quad (9)$$

The supplier can provide a limited quantity for each type of subcomponent. The situation is asserted in (10)–(13). The

parameter Max_e is the maximum number of subcomponents (of general type e) that can be supplied.

$$\sum_{i=1}^N B(i,p) \leq Max_p \quad \forall p, \quad (10)$$

$$\sum_{i=1}^N B(i,a) \leq Max_a \quad \forall a, \quad (11)$$

$$\sum_{i=1}^N B(i,r) \leq Max_r \quad \forall r, \quad (12)$$

$$\sum_{i=1}^N B(i,d) \leq Max_d \quad \forall d, \quad (13)$$

There is a limited amount of investment budget as stated in (14):

$$IC \leq IC_{\max} \quad (14)$$

5.2. *Human Resource Cost.* Driver salaries and training during Y years constitute the total human resource cost. Equation (15) formalizes this fact for N buses. This formula neglects the fixed HR costs. In this equation, ed is the index of education that belongs to set $ED = \{\text{Under high school, High school diploma, Associate degree, Bachelor, Master, Ph.D.}\}$. The binary variable $EDU_{i,ed}$ indicates whether the level of education of the i^{th} bus driver is equal to ed . The

index x represents the index of experience level. It can take quantitative values of Table 1. The set of these values is denoted as X . $EXP_{i,x}$ is a binary variable, which equals one if the driver of the i^{th} bus has an experience level of x . The continuous variable h_i is the total training hours of the i^{th} bus driver. C_h is the annual cost per hour of training. $C(ed)$ and $C(x)$ are the additional monthly income that system owners should pay to a driver with an education level of ed and experience of x .

$$HRC = Y \cdot \left\{ \sum_{i=1}^N \left[C_h * (h_i - h_{\min}) + 12 \sum_{e=d=1}^{|ED|} EDU_{i,e,d} * C(e,d) + 12 \sum_{x=1}^{|X|} EXP_{i,x} * C(x) \right] \right\} \quad (15)$$

Constraint (16) asserts there is a lower and upper band for training hours.

$$h_{\min} \leq h_i \leq h_{\max} \quad \forall i. \quad (16)$$

Each driver has only one specific level of education and experience. This fact is mathematically modeled in (17) and (18).

$$\sum_{e=d=1}^{|ED|} EDU_{i,e,d} = 1 \quad \forall i, \quad (17)$$

$$\sum_{exp=1}^{|X|} EXP_{i,x} = 1 \quad \forall i. \quad (18)$$

Since regulations and policies limit the number of employed drivers who possess a specific level of education, constraint (19) sets the maximum number of drivers within each level of education. In this equation, Max_{ed} is the maximum number of drivers with an education level of ed that policies allow to hire.

$$\sum_{i=1}^N EDU_{i,e,d} \leq Max_{ed} \quad \forall ed. \quad (19)$$

The transportation service company would prefer not to dedicate a tremendous amount of money to HR. Therefore,

the human resource cost is bounded as represented in constraint (20).

$$HRC \leq HRC_{\max} \quad (20)$$

5.3. *Outage Cost.* Equation (21) states that the outage cost is the multiplication of total duration years (Y), the brake system failure rate of bus i (f_i), the average time a bus stays in a repair shop due to brake failure (μ), and the bus interruption cost per hour (I_i).

$$OC = \sum_{i=1}^N (Y * f_i * \mu * I_i). \quad (21)$$

The failure rate is estimated with a linear model, as discussed in Section 4. It is stated in (22):

$$f_i = \beta_0 + \beta_1 * ETE_i + \beta_2 * Q_i \quad \forall i. \quad (22)$$

ETE and Q , introduced in more detail in Section 4, are calculated through equations (23) and (24). In these equations, S_x and S_{ed} are the scores of experiences and education for the experience level of x and education level of ed according to Table 1 in Section 4. According to Table 1, the maximum ETE and Q scores are 40 and 60. To scale these scores between 0 and 100, they are multiplied by ratios 100/40 and 100/60. These two coefficients can change if a different scoring schema is used.

$$ETE_i = \frac{100}{40} \left[\sum_{e=d=1}^{|ED|} S_{e,d} * EDU_{i,e,d} + \sum_{x=1}^{|X|} S_x * EXP_{i,x} + S_h * (h_i - h_{\min}) \right] \quad \forall i, \quad (23)$$

$$Q_i = \frac{100}{60} \left[\sum_{p=1}^{|P|} S_p * B(i,p) + \sum_{a=1}^{|A|} S_a * B(i,a) + \sum_{r=1}^{|R|} S_r * B(i,r) + \sum_{d=1}^{|D|} S_d * B(i,d) \right] \quad \forall i. \quad (24)$$

The total available ETE is limited due to issues such as a limited number of high-quality candidates. Similarly, the

total quality of the brake system is limited. Constraints (25) and (26) restate this fact.

$$\sum_{i=1}^N \text{ETE}_i \leq \text{SUM_ETE}_{\max} \quad \forall i, \quad (25)$$

$$\sum_{i=1}^N Q_i \leq \text{SUM_Q}_{\max} \quad \forall i. \quad (26)$$

5.4. Replacement Cost. Replacement cost is the sum of the expected cost of replacing each subcomponent after it fails. This fact is mathematically asserted in equation (27). The replacement cost for each component equals the cost of a single component multiplied by the expected damages over Y years. It can be rewritten as (28)–(31).

$$\text{RC} = \sum_{i=1}^N \left[\sum_{p=1}^{|P|} \text{RC}_{i,p} + \sum_{a=1}^{|A|} \text{RC}_{i,a} + \sum_{r=1}^{|R|} \text{RC}_{i,r} + \sum_{d=1}^{|D|} \text{RC}_{i,d} \right] \quad \forall i, \quad (27)$$

$$\text{RC}_{i,p} = Y * f_i * \nu_P * B(i, p) * C(p) \quad \forall i, p, \quad (28)$$

$$\text{RC}_{i,a} = Y * f_i * \nu_A * B(i, a) * C(a) \quad \forall i, a, \quad (29)$$

$$\text{RC}_{i,r} = Y * f_i * \nu_R * B(i, r) * C(r) \quad \forall i, r, \quad (30)$$

$$\text{RC}_{i,d} = Y * f_i * \nu_D * B(i, d) * C(d) \quad \forall i, d. \quad (31)$$

In (28), the parameter ν_P is the relative failure frequency of the pedal. This parameter is approximated by analyzing historical data. It can be estimated using historical data. Other variables inside (28)–(31) are introduced in previous sections. In (28), the multiplication of the continuous variables f_i and $B(i, p)$ is nonlinear. The same happens in (29)–(31) for ABS, retarder, and pad, respectively. To linearize these equations, a conversion, introduced in [32], is used. According to this conversion, equation (32) is linearized by replacing it with (33) and (34) [32]. This conversion is applied to (28)–(31) for them to linearize.

$$\text{multiplication} = \text{binary} * \text{continuous}, \quad (32)$$

$$0 \leq \text{multiplication} \leq \text{binary} * \text{continuous}_{\max}, \quad (33)$$

$$\begin{aligned} &\text{continuous} + (\text{binary} - 1) * \text{continuous}_{\max} \\ &\leq \text{multiplication} \leq \text{continuous}. \end{aligned} \quad (34)$$

To summarize, the optimization problem is modeled as a mixed-integer linear program (MILP) with the objective function of equation (4) and constraints (5)–(27) and linearized (28)–(31). Decision variables are the types of subcomponents chosen for bus brakes and driver education and experience and training.

6. Case Study

To verify the efficiency of the method, a real case study of a BRT service is presented. The first BRT system in Tehran, Iran, was initiated in 2007. Currently, ten routes are operating in Tehran. Buses operate in specially dedicated routes

in which other vehicles are not allowed. Moreover, in the case of a junction, BRT buses have priority. Additional routes are planned and added as required. The data for brake system failures, subcomponent types, and drivers of 183 buses were collected. However, costs were modified for security reasons. This case will decide the types of subcomponents and ETEs for ten buses planned to be exploited in three BRT lines for 20 years. Subcomponent prices are shown in Table 2. Recall from Section 4 that types A, B, and C components have 15, 10, and 5 scores, respectively. Due to the supplier limitations, no more than two subcomponents of type A can be provided. The average repair times for each subcomponent and relative failure frequencies are listed in Table 3. The brake system average repair time is the average repair time for each subcomponent weighted by the relative failure frequencies. Each hour of training per year costs 6.7 USD. The maximum training hours per year for each driver is 120 hours. The salaries of drivers are listed in Table 4. The company's policy allows a maximum of one Ph.D. driver, two masters, and three holding any other degrees.

The transportation company has a budget of 1,800,000 USD for driver salaries and training over 20 years. Similarly, no more than 9000 USD is available for the brake system of these ten buses. Each hour of bus interruption costs 201.289 USD for line 1, 196.2 USD for line 2, and 150 USD for line 3. Therefore, the maximum total score of 900 is considered available for both subcomponents and driver ETEs.

7. Results and Discussion

As demonstrated in Table 5, the training is at the maximum possible level. It is mainly because training is relatively cheaper. Subcomponents of type A are only installed at buses 4 and 5, belonging to line 1. On the contrary, buses 9 and 10 possess subcomponents of less quality. This is because the interruptions in line 3 result in lower outage costs. In the optimal strategy, the salary and brake investment costs are 1,799,427.795 and 8,970.461 USD, close to their maximum values. The total expected replacement and outage costs are 2,857,811.418 and 5,690,168.350 USD, respectively. Therefore, the total cost is 10,356,378.86 USD.

7.1. Sensitivity Analysis. The transport company may hypothesize whether increasing the brake system investment or training and salaries would lower total costs over 20 years. To investigate this, several cases of sensitivity analysis are performed. First, the effect of brake investment limitation is investigated. Next, an analysis is performed to identify whether an enhancement in the HR cost limitation would change the optimal total cost. It is assumed that the sum of brake investment and HR costs is constant. Finally, the effects on the total expected cost are evaluated.

7.1.1. Brake Investment Limitations. If constraint (14), which limits the brake investment cost, is omitted, the total cost would be 10,064,524.85 USD, and the

TABLE 2: Price of subcomponents (USD).

Subcomponent	Type A	Type B	Type C
Retarder	247.678	201.238	154.798
Pad	77.399	68.111	61.919
ABS	773.99	619.19	495.35
Pedal	120.3591	100.399	20.8235

TABLE 3: Repair times and relative failure frequencies.

Subcomponent	Repair time (hr)	Relative failure frequency
Retarder	2.5785	0.1766
Pad	2.8197	0.2470
ABS	0.783	0.3114
Pedal	4.258033	0.2649

TABLE 4: Salary per month of drivers (USD).

Education level	Under high school	High school diploma	Associate degree	Bachelor	Master	Ph.D.
Salary	544.892	557.276	603.715	650.155	743.034	1083.591
Years of experience	0-5	5-10	10-15	15-20	20-25	25-30
Salary	80.49535	100.7430	140.9907	182.9102	241.9814	322.4767

TABLE 5: Results.

Bus #	Edu.	Training	Exp.	ETE	Retarder	Pad	ABS	Pedal	Q
1	High school diploma	120	5-10	65.0	B	B	C	B	58.33
2	Bachelor	120	5-10	72.5	B	B	C	B	58.33
3	Associate degree	120	5-10	67.5	B	B	C	B	58.33
4	Bachelor	120	5-10	72.5	A	A	B	A	91.66
5	Bachelor	120	5-10	72.5	A	A	B	A	91.66
6	High school diploma	120	5-10	65.0	B	B	C	B	58.33
7	Under high school	120	5-10	62.5	B	B	C	B	58.33
8	High school diploma	120	5-10	65.0	B	B	C	B	58.33
9	Under high school	120	0-5	60.0	B	B	C	B	58.33
10	Under high school	120	0-5	60.0	B	B	C	C	50.00

investment cost would be 10350.356 USD, which is 1350.365 USD more than the current investment budget. Therefore, if the transportation service provides an additional 1350.365 USD financial resources for investing in the brake system, the total cost would decrease by 291,854.01 USD. Figure 2 depicts the effect of changing the brake investment cost on the total expected cost over 20 years.

7.1.2. HR Cost Limitations. Similar to the procedure used in the previous section, constraint (20) is omitted to analyze the effects of HR cost limitations. In this case, the HR cost would be 2587994.854 USD, and the total expected cost would be 9275496.72 USD. Therefore, an increase of 787,994.854 USD in the HR cost would result in 1,080,882.14 USD benefits in the total expected cost. Since the interval of spending HR and total expected costs are almost simultaneous, the BRT service can revise its policy based on the results of this optimization. The variation of total expected cost during 20 years versus the HR cost during the same interval is depicted in Figure 3.

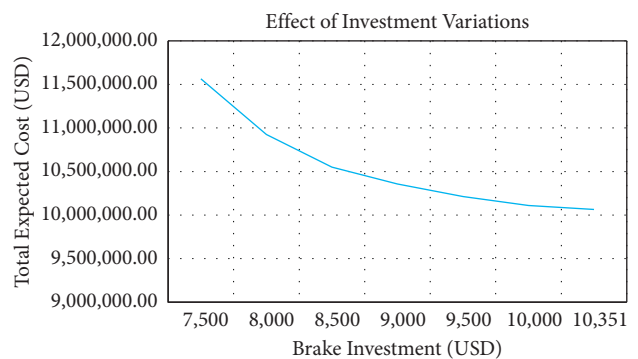


FIGURE 2: Brake investment sensitivity analysis.

7.1.3. HR Cost Limitation and Brake Investment Budget Joint Analysis. As seen in previous sections, an increase in HR limitation or brake investment would decrease the expected cost. If the transport company questions whether decreasing one limitation in favor of the other would reduce the expected cost, another sensitivity analysis should be performed. Table 6 provides more insights into this question. It

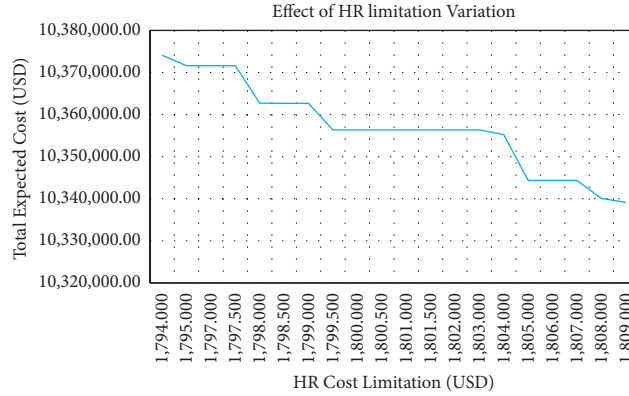


FIGURE 3: HR cost sensitivity analysis.

TABLE 6: HR cost and brake investment limitations trade-off.

Case #	Brake investment limitation	HR cost limitation	Total cost
1	8,000	1,801,000	10,924,940.31
2	8,500	1,800,500	10,551,420.18
3 (Base case)	9,000	1,800,000	10,356,378.86
4	9,500	1,799,500	10,213,323.76
5	10,000	1,799,000	10,118,665.99

TABLE 7: Related notations.

		Indexes	
a	Index of ABS types	p	Index of pedal type
d	Index of pad types	r	Index of retarder types
ed	Index of level of education	x	Index of level of experience
i	Index of buses		
		Sets	
A	Set of ABS types	P	Set of pedal type
D	Set of pad types	R	Set of retarder types
ED	Set of the of education	X	Set of levels of experience
		Parameters	
β_0	Constant value in failure rate model	I_i	Cost of an hour of interruption of bus i
β_1	Coefficient of E_{TE} in failure rate model	Max_e	Maximum available of item e (e could be a subcomponent type or education level or experience level)
β_2	Coefficient of Q in failure rate model	S_e	The score of item e (e could be a subcomponent type or education level, or experience level)
M	Number of samples for failure rate modeling	v_g	The relative failure rate of item e (e could be a pedal, pad, retarder, or ABS)
N	Number of buses	Y	Total years of planning
$C(e)$	Cost of item e (e could be a subcomponent type or education level or experience level)		
C_h	The annual cost of an hour of training	μ	Average buses' brake repair time
		Variables	
$B(i,e)$	A binary variable indicating whether item e is bought for bus i (e could be a pedal, pad, retarder, or ABS)	HRC	Human resource cost
$EDU_{i,ed}$	Binary variable indicating whether education level of the driver of bus i is ed	IC	Investment cost
E_{TE}_i	Education, training, and experience score of bus i 's driver	OC	Outage cost
$EXP_{i,x}$	Binary variable indicating whether the experience level of the driver of bus i is x	RC	Replacement cost
f_i	The failure rate of bus i	Q_i	Brake of i^{th} bus's quality score

can be observed that increasing brake investment limitation while keeping the sum of brake investment and HR cost constant would decrease the total expected cost. Notice that HR cost is spent in the broader interval of time. Therefore, supplying the financial resources for HR costs is easier.

8. Conclusions

This paper presented a joint brake system and driver employment and training optimization for buses in BRT systems. The objective function was to minimize the brake reliability-related costs plus investment costs. It has been observed that both qualities of the brake system subcomponents and driver ETE (education, training, and experience) indices are influential factors for the failure rate and, in consequence, the total expected cost. However, there are limited financial resources for these two factors, which should be modeled. Also, overspending on these two factors may put an unnecessary extra cost on the shoulders of service providers. Therefore, sensitivity analysis and optimization should be performed. A case study has been presented and analyzed to verify the efficiency of the method. The results assert that better subcomponents and drivers should be dedicated to bus lines with more interruption costs per hour. It has also been shown that if enough budgets are provided for brake systems, the total expected cost will decrease noticeably.

Furthermore, sufficient spending for the ETE would reduce costs. Providing a budget for the brake system is a challenging task. However, the ETE expenses are spread over many years; therefore, they are more practical to provide. Moreover, the saved money, which should have been expended as interruption losses, can be dedicated to ETE. The results have been presented to the abovementioned practical BRT system owner. After analyzing the strategy, they agreed to implicate HR employment, training, and subcomponent supply results in their planning and operations programs.

Nevertheless, considering the role of other factors, including seasonal factors and loading, in brake system reliability results in a more precise cost modeling and optimization in practice. For the future stream of research, co-optimizing the total expected brake-related expenses with repair staff employment is suggested. The optimal number of repair staff is employed to decrease expected outage durations and expected outage cost in consequence.

9. Summary of Notions

Table 7 contains a summary of all indices, variables, and parameters that have been mentioned throughout this paper.

Data Availability

The bus failure data and driver specifications are not publicly published due to the safety and security of third parties.

Conflicts of Interest

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

Acknowledgments

This study was funded by the Industrial Engineering and Productivity Research Center of the Amirkabir University of Technology.

References

- [1] R. Billinton and R. N. Allan, *Reliability Evaluation of Engineering Systems*, Plenum Press, New York, NY, USA, 1992.
- [2] R. Liu, H. Yu, P. Wang, and H. Yan, "A short-turn dispatching strategy to improve the reliability of bus operation," *Journal of Advanced Transportation*, vol. 2020, Article ID 5947802, 13 pages, 2020.
- [3] M. Ansari Esfeh, S. C. Wirasinghe, S. Saidi, and L. Kattan, "Waiting time and headway modelling for urban transit systems—a critical review and proposed approach," *Transport Reviews*, vol. 41, pp. 1–23, 2020.
- [4] R. Liu and S. Sinha, "Modelling urban bus service and passenger reliability," in *Proceedings of the Third International Symposium on Transportation Network Reliability*, The Hague, Netherlands, July 2007.
- [5] B. Barabino, N. A. Cabras, C. Conversano, and A. Olivo, "An integrated approach to select key quality indicators in transit services," *Social Indicators Research*, vol. 149, pp. 1–36, 2020.
- [6] F. Maltinti, N. Rassa, M. Coni et al., "Vulnerable users and public transport service: analysis on expected and perceived quality data," in *Proceedings of the International Conference on Computational Science and Its Applications*, pp. 673–689, Cagliari, Italy, July 2020.
- [7] N. van Oort, D. Sparing, T. Brands, and R. M. P. Goverde, "Data driven improvements in public transport: the Dutch example," *Public transport*, vol. 7, no. 3, pp. 369–389, 2015.
- [8] B. Barabino, M. Di Francesco, and S. Mozzoni, "An offline framework for the diagnosis of time reliability by automatic vehicle location data," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 3, pp. 583–594, 2016.
- [9] B. Barabino, C. Lai, C. Casari, R. Demontis, and S. Mozzoni, "Rethinking transit time reliability by integrating automated vehicle location data, passenger patterns, and web tools," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 4, pp. 756–766, 2016.
- [10] X. Chen, L. Yu, Y. Zhang, and J. Guo, "Analyzing urban bus service reliability at the stop, route, and network levels," *Transportation Research Part A: Policy and Practice*, vol. 43, no. 8, pp. 722–734, 2009.
- [11] J. Paudel, "Bus ridership and service reliability: the case of public transportation in Western Massachusetts," *Transport Policy*, vol. 100, pp. 98–107, 2021.
- [12] A. Kathuria, M. Parida, and C. R. Sekhar, "A review of service reliability measures for public transportation systems," *International Journal of Intelligent Transportation Systems Research*, vol. 18, pp. 1–3, 2019.
- [13] D. Nam, D. Park, and A. Khamkongkhun, "Estimation of value of travel time reliability," *Journal of Advanced Transportation*, vol. 39, no. 1, pp. 39–61, 2005.
- [14] J. Bunker, "Planning for transit system reliability using productive performance and risk assessment," in *Proceedings of the Transportation Research Board (TRB) 92nd Annual Meeting Compendium of Papers*, pp. 1–16, Washington, NJ, USA, March 2013.
- [15] S. M. H. Moosavi, A. Ismail, and C. W. Yuen, "Using simulation model as a tool for analyzing bus service reliability and

- implementing improvement strategies,” *PLoS One*, vol. 15, no. 5, p. e0232799, 2020.
- [16] M. M. Nesheli and A. Ceder, “Improved reliability of public transportation using real-time transfer synchronization,” *Transportation Research Part C: Emerging Technologies*, vol. 60, pp. 525–539, 2015.
- [17] B. Barabino and M. Di Francesco, “Diagnosis of irregularity sources by automatic vehicle location data,” *IEEE intelligent transportation systems magazine*, vol. 13, no. 2, 2021.
- [18] J. Lin, P. Wang, and D. T. Barnum, “A quality control framework for bus schedule reliability,” *Transportation Research Part E: Logistics and Transportation Review*, vol. 44, no. 6, pp. 1086–1098, 2008.
- [19] B. Barabino, M. Di Francesco, G. Maternini, and S. Mozzoni, “Offline framework for the diagnosis of transfer reliability by automatic vehicle location data,” *IEEE Intelligent Transportation Systems Magazine*, 2021.
- [20] Y. Wang, D. Zhang, L. Hu, Y. Yang, and L. H. Lee, “A data-driven and optimal bus scheduling model with time-dependent traffic and demand,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 9, pp. 2443–2452, 2017.
- [21] L. Zhao, S. I. Chien, L. N. Spasovic, and X. Liu, “Modeling and optimizing urban bus transit considering headway variation for cost and service reliability analysis,” *Transportation Planning and Technology*, vol. 41, no. 7, pp. 706–723, 2018.
- [22] A. Sajedinejad and E. Hasannayebi, “Implementation of operational city bus systems in order to organize public transportation,” *Geographical Researches Quarterly Journal*, vol. 31, no. 4, pp. 60–73, 2017.
- [23] A. A. Zavgorodniy, V. A. Maksimov, N. V. Pozhivilov, and G. A. Krylov, “Analysis of failures of urban buses in the conditions of the transit agency “mosgortrans,”” *IOP Conference Series: Materials Science and Engineering*, vol. 832, no. 1, p. 12074, 2020.
- [24] M. Erdoğan and İ. Kaya, “A systematic approach to evaluate risks and failures of public transport systems with a real case study for bus rapid system in Istanbul,” *Sustainable cities and society*, vol. 53, p. 101951, 2020.
- [25] R. R. Yusupov, “Designed brake system reliability assessment techniques application on the basis of the real one,” *IOP Conference Series: materials Science and Engineering*, vol. 695, no. 1, 2019.
- [26] L. Liu, X. Liu, X. Wang, Y. Wang, and C. Li, “Reliability analysis and evaluation of a brake system based on competing risk,” *Journal of Engineering Research*, vol. 5, no. 3, 2017.
- [27] A. Mihalache, F. Guerin, M. Barreau, and A. Todoskoff, “Reliability analysis of mechatronic systems using censored data and Petri nets: application on an antilock brake system (abs). InRAMS’06,” in *Proceedings of the Annual Reliability and Maintainability Symposium*, pp. 140–145, Newport Beach, CA, USA, January 2006.
- [28] E. N. Bohr, C. P. Ukpaka, and B. Nkoi, “Reliability analysis of an automobile brake system to enhance performance,” *International Journal of Production Engineering*, vol. 4, no. 2, pp. 47–56, 2018.
- [29] D. Nikitin, A. Asoyan, and L. Nikitina, “A method for reliability improvement in air brake system of compressed air cars,” *Transportation Research Procedia*, vol. 36, pp. 533–539, 2018.
- [30] R. R. Yusupov, A. D. Jarzemskiy, A. V. Alekseev, M. G. Korchazhkin, L. A. Berdnikov, and A. A. Pikulkiin, “The use of methods for assessing reliability of the designed brake system on the basis of the existing one,” *Journal of Physics: Conference Series*, vol. 1177, p. 12012, 2019.
- [31] N. N. Thach, L. H. Anh, and H. N. Khai, “Applying Lasso linear regression model in forecasting Ho chi minh city’s public investment,” *Data Science for Financial Econometrics*, pp. 245–253, 2021.
- [32] G. P. McCormick, “Computability of global solutions to factorable nonconvex programs: Part I - convex underestimating problems,” *Mathematical Programming*, vol. 10, no. 1, pp. 147–175, 1976.

Research Article

Evaluation and Analysis Model of the Length of Added Displaced Left-Turn Lane Based on Entropy Evaluation Method

Binghong Pan ¹, Jinfeng Ying ¹, Shasha Luo ¹, Yang Shao ², Shangru Liu ¹,
Xiang Li¹ and Zhenjiang Xie ¹

¹Highway School, Chang'an University of Shaanxi Province, Xi'an 710064, China

²School of Modern Posts (Logistics School) and Institute of Posts, Xi'an University of Posts and Telecommunications, Xi'an 710121, China

Correspondence should be addressed to Binghong Pan; panbh@chd.edu.cn

Received 12 June 2021; Revised 25 August 2021; Accepted 24 September 2021; Published 12 October 2021

Academic Editor: Erfan Hassannayebi

Copyright © 2021 Binghong Pan 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.

As the number of vehicles continues to grow in China, the problem of urban traffic congestion gets more serious, particularly at intersections. As a new type of unconventional intersection, the displaced left-turn (DLT) intersection has been widely recognized to improve the efficiency of intersections with heavy left-turn traffic flows. Converting an existing intersection into an intersection with additional DLT lanes is a valuable solution. However, the studies on DLT intersections mainly focus on performance, applicability, and safety. The guidelines on the specific engineering design process mostly come from experience, and the study on the considering multifactor design method is still insufficient. Therefore, this paper proposed an evaluation and analysis model of the lengths of added DLT lanes based on the entropy evaluation method, in which VISSIM and Surrogate Safety Assessment Model (SSAM) software were adopted for simulation. A design process for the length of the added DLT lanes was proposed with this model. An urban intersection in Xi'an was taken as a case study, and the application of the model and the design process was studied in detail. After selecting four evaluation indicators, the model was applied to calculate and analyze the optimal length of the added DLT lanes under 45 different traffic volume combinations. The recommended lengths of different situations were within the range recommended in the guidebook published by Federal Highway Administration. The results of the case study proved that the model proposed in this paper was advanced, reasonable, and practical.

1. Introduction

Since the beginning of the 21st century, the process of urbanization has accelerated and the size of cities has grown in China. Subsequently, the number of vehicles continues to rise, and the urban road traffic congestion problem is getting ever more serious. By the end of 2020, the "2020 China Urban Traffic Report" showed that the traffic congestion problem was still severe, among which the average speed of vehicles was basically around 30 km/h during the peak commuting hours [1]. This undoubtedly presented that the traffic efficiency of urban roads in peak hours still needed to be improved immediately, and the reduction of traffic efficiency also has a certain impact on the economic and social development of the whole city. Therefore, it is still a

significant need for the study to alleviate the problem of urban traffic congestion.

As an important joint of a road network, intersections are essential for improving the traffic efficiency of the entire road network. Some researchers have pointed out that the operational efficiency of an entire urban road network is often affected by some bottleneck sections. Moreover, urban intersections are often the bottleneck sections of urban roads [2]. To alleviate the congestion problem of urban road intersections, many scholars have studied how to improve the efficiency of intersections with different aspects, such as the new geometric form of intersections, signal timing optimization, and new traffic management technologies. Some scholars have studied the performance of the new U-turn intersections [3, 4], fan-shaped intersections [5], double-

torsional intersections [6], and parallel flow intersections [7]. The different characteristics of various new alternative intersections have also been summarized [8]. For signal timing control, some scholars have proposed some simple intersection signal control strategies based on different methods [9, 10]. Then, with the development of Internet of Vehicles and autonomous driving technology, some scholars have proposed signal models with the trajectory data of autonomous driving and Internet of Vehicles to optimize signal control [11]. In addition, as the “people-oriented” concept has gradually become a consensus, the signal control method of optimization has been proposed to comprehensively consider the total delay of pedestrians and vehicles passing an intersection [12]. There have also been attempts to improve the traffic conditions at intersections with the new traffic management. For traffic management, some scholars have used Internet of Things technology to propose a real-time optimization control model of the intersection traffic flow [13, 14], while some researchers have also studied how to use the Internet of Things technology to enhance the efficiency of intersections [15].

In addition to the above methods of the overall improvement of the intersection, left-turning traffic flows are essential to improving the crossing capacity and safety in the range of intersections because left-turning traffic flows are most likely to cause conflicts when running in intersections. Some scholars have conducted a detailed investigation and analysis on the operation of left-turning vehicles at intersections [16]. There were also a lot of results of solving the problem of left-turning traffic flows at intersections. To be more specific, some scholars have studied the improvement effect of VISSIM-based simulation and evaluated the operational impact of left-turn waiting areas at signalized intersections [17, 18]. Some studies have proposed to set up reasonable modeling for left-lane line extensions [19]. A method for the dynamic use of the left-turn lane for opposite through traffic to improve the efficiency of a signalized intersection has also been proposed [20]. Some scholars have studied the effect of U-turn facilities [21], while other scholars have specialized in optimizing the phase of left-turn signals at intersections to improve left-turning traffic problems [22].

The displaced left-turn (DLT) intersection is the new type of unconventional intersection and focuses on improving the operation of left-turning traffic flows. The DLT intersection is also referred to as the continuous flow intersection (CFI) in many studies. The core design concept of a DLT intersection is to set up a subintersection before left-turning vehicles enter the main intersection so that left-turning vehicles can change to the outside of the opposite lanes in advance and eliminate the conflicts between left-turning traffic flows and the opposite straight traffic flow at the main intersection [23]. Due to the elimination of conflicts between left-turning traffic flows and opposite straight traffic flows, straight and left-turning vehicles at the main intersection can be green at the same time. To date, some scholars have carried out meaningful studies on DLT intersections, such as the applicability of displaced left-turn intersections, and have put forward two forms: partial and complete DLT intersections [7]. Through observation data of

actual DLT intersections, the advantages of DLT intersections with improving traffic safety in turning can be found [24]. In addition, other scholars have also studied the safety of DLT intersections by using before-and-after comparison group and cross-sectional analysis methods and then pointed out that DLT intersections need to be equipped with traffic signs and traffic-calming facilities to achieve better use results [25]. Some scholars have put forward signal timing optimization models for DLT intersections applying the Monte Carlo method [26] and traffic progression method [27]. Others have investigated a dynamic and optimized method of traffic signal timing parameters to improve the integrated performance index of DLT intersections [28]. Besides, researchers have developed a left-hand excursion plane crossing design process [29]. Because of the increasing popularization of low-carbon travel in China, researchers have proposed a displaced left-turning bicycle lane based on the concept of a DLT lane [30]. Some scholars have put forward a new simplified DLT intersection (called CFI-Lite) on the basis of a DLT intersection and have verified its practicability [31]. Further, some scholars have made a comprehensive comparison of DLT intersections with other new type intersections with the operation and safety and have further explained the characteristics of DLT intersections [32]. Some researchers have investigated the coordination of consecutive DLT intersections under heterogeneous traffic conditions with a case study [33]. Other scholars have also evaluated alternative pedestrian and bicycle crossing schemes at continuous flow intersections [34]. There was a wealth of studies on the suitability, safety, and optimization of the DLT intersections.

However, the current design guidelines for unconventional DLT intersections were mostly from engineering experience and mathematical analysis methods [29]. They still lack detailed and specific guidance. At present, most of the studies on DLT intersections are about its applicability and performance or put forward the formula of queue length and delay on the basis of experience regression analysis. The design of intersections usually needs to take into account functions, such as traffic efficiency, safety, and environmental protection. Studies on considering the multifactor design method of reconstructing unconventional DLT intersections are still insufficient. The analytic hierarchy process (AHP) is often used to solve the multiobjective and comprehensive evaluation problem of the current engineering field. But this involves the participation of human factors, which is often controversial. The entropy method is a weighting method utilizing objective data and has been applied to the field of scientific research. Focused on this, this paper proposes a DLT lane lengths’ analysis and evaluation model based on the entropy method that can be used to guide the design. The usage of simulation analysis in models is an important scientific research method [35, 36]. Since VISSIM and SSAM are simulation software that have a wide range of applications in engineering and research [37–39], they have been applied to this design model. Taking an urban arterial road intersection in Xi’an as a case study, the proposed model is exactly applied to test its advancement, rationality, and practicability.

Xi'an, the capital of Shaanxi Province, is also one of the top five cities in China in the 2020 Commuter Peak Congestion Index. Therefore, this paper takes an intersection on an urban arterial road with a high left-turning traffic volume as a case study. Built on the current form of intersections, a new intersection with DLT lanes on the outside of the intersection is proposed, which includes the gradual widening of the intersection. In addition, different DLT lane lengths are proposed to be improved by using equal-length intervals.

The remainder of this paper is arranged as follows: Section 2 introduces the DLT lane length evaluation and analysis model based on the entropy method that can provide guidance for intersection reconstruction design. Section 3 shows the case study about an intersection on urban arterial roads in Shaanxi Xi'an and outlines the model calibration. Then, Section 4 provides the sensitivity analysis of the VISSIM simulation and explains the scheme evaluation and comparison. Last but not least, Section 5 puts forward the research conclusions of this paper.

2. Model

The purpose of this paper is to come up with a new evaluation and analysis model for the reconstruction of added DLT lanes intersection. It can make the design scientific and help the designers to determine the appropriate length of the externally increased DLT lanes.

2.1. Preliminaries. Before the establishment of the model, specific requirements should be put forward for the actual engineering conditions to which the model is applicable. The prerequisites for using the model are as follows:

- (i) Reconstruction of urban road intersection project and enough space for reconstruction
- (ii) The number of added DLT lanes is the same as the number of left-turn lanes at the existing intersection, so the impact caused by changes in the number of lanes can be ignored
- (iii) Regarding the independent urban cross intersection, the distance from the adjacent intersection upstream and downstream is large enough, and coordinated control is not considered

2.2. Entropy Method. In the process of analysis and calculation, how to operate a unified standard comparison to select the recommended length scheme is an essential issue that must be solved. And, this involves a scientific comparison between multiple parameters and multiple data.

The analytic hierarchy process (AHP), which is widely used in multiple parameters' comparison, is one of the weight evaluation methods. This includes expert scoring. However, the expert scores to determine the weight of indicators mainly rely on the subjective judgment of experts, which is controversial. In order to avoid this problem, the model in this paper adopts the entropy evaluation method (EEM) to calculate the weight.

The EEM is an effective method to solve multiparameter and multidata processing problems objectively. It comes from the concept of information entropy. Information entropy describes the average amount of information on the data, which means the more chaotic the system and the greater the amount of information carried. Comparing with other common weight calculation methods, the EEM cannot only realize the comparison of multiple parameters and multiple data between different schemes but also avoids the controversy caused by the participation of human factors. And, the EEM has been widely employed in many scientific fields such as electrical engineering [40], environment engineering [41], and water conservancy projects [42]. Therefore, the EEM is applied to the model in this paper as an important part.

The specific calculation process with EEM is as follows:

$$\begin{aligned} A &= A_{n \times k}, \\ B &= B_{n \times k}, \\ G &= G_{n \times k}, \end{aligned} \quad (1)$$

where A , B , and G represent the three different indexes, respectively, n means the total number traffic volume combination, and k represents the total scheme number. First, the same index of all the evaluated schemes should be converted into a matrix. Then, we take processing A matrix as an example.

Matrix A contains $n \times k$ simulation results (i represents the traffic volume combination number; j represents the scheme number):

$$A = \begin{bmatrix} A_1(1) & A_2(1) & \cdots & A_k(1) \\ A_1(2) & A_2(2) & \cdots & A_k(2) \\ \vdots & \vdots & A_j(i) & \vdots \\ A_1(n) & A_2(n) & \cdots & A_k(n) \end{bmatrix}. \quad (2)$$

Then, the best value of each row in matrix A is selected (the smaller the value of the index one, the better the performance, so the minimum value of each row is selected as the optimal. On the contrary, the larger the value, the better, so the maximum value of each row is selected). For instance, if the value in the A index is smaller, the better, then the minimum value of each row in the A matrix is the best. Therefore, a new matrix A_m is generated as follows:

$$A_m = \begin{bmatrix} \min A_j(1) \\ \min A_j(2) \\ \vdots \\ \min A_j(n) \end{bmatrix}, \quad j = 1 \text{ to } k. \quad (3)$$

The rest of the evaluation index matrices also repeat the above process:

$$\begin{cases} A_m(i) = \min A_j(i), \\ B_m(i) = \min B_j(i), \\ G_m(i) = \min G_j(i), \\ i = 1 \text{ to } n, j = 1 \text{ to } k. \end{cases} \quad (4)$$

The third step is to combine the above matrices into an overall matrix M with n rows and p columns, and the weight of each index can be calculated:

$$M = \begin{bmatrix} A_m(1, j) & B_m(1, j) & G_m(1, j) \\ A_m(2, j) & B_m(1, j) & G_m(1, j) \\ \vdots & \vdots & \vdots \\ A_m(n, j) & B_m(1, j) & G_m(1, j) \end{bmatrix} \quad (5)$$

$j = 1 \text{ to } k$

Each element in the matrix M can be expressed as

$$M = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ \vdots & \vdots & \vdots \\ m_{n1} & m_{n2} & m_{n3} \end{bmatrix} \quad (6)$$

The column vector in matrix M is represented as

$$m_j = [m_{1t}, m_{2t}, \dots, m_{kt}]^T, \quad t = 1, 2, \dots, q, \quad (7)$$

where q denotes the total number of the index type:

$$M = [m_1, m_2, \dots, m_q]. \quad (8)$$

Units for the different indexes in the matrix M are not the same. To further use the entropy method to process the data, all values should be standardized before the next step because the various indexes are not uniform. The absolute value is converted to the relative value to solve the normalization problem of the different property indexes. The specific calculation is as follows:

$$m'_{it} = \frac{m_{it} - \min\{m_{1t}, \dots, m_{nt}\}}{\max\{m_{1t}, \dots, m_{nt}\} - \min\{m_{1t}, \dots, m_{nt}\}}, \quad (9)$$

where m'_{it} represents the normalization value.

In equation (9), the greater the gap between the indexes m_{it} , the greater the effect of m_{it} . On the contrary, if the index value is the same as another, it means that the index value had no influence on the final evaluation.

We obtain the new matrix M' for the forward procession:

$$M' = \begin{bmatrix} m'_{11} & m'_{12} & m'_{13} \\ m'_{21} & m'_{22} & m'_{23} \\ \vdots & \vdots & \vdots \\ m'_{n1} & m'_{n2} & m'_{n3} \end{bmatrix} \quad (10)$$

The weight of index j of scheme i is calculated as follows:

$$p_{it} = \frac{m'_{it}}{\sum_{i=1}^n m'_{it}}, \quad i = 1 \text{ to } n, t = 1 \text{ to } 3. \quad (11)$$

The entropy value of index j is calculated as follows:

$$e_t = -h \sum_{i=1}^n p_{it} \ln(p_{it}), \quad (12)$$

where $h = 1/\ln(n)$ and satisfies $e_t \geq 0$. The entropy redundancy is calculated as follows:

$$d_t = 1 - e_t. \quad (13)$$

The weights of each index are calculated as follows:

$$p_t = \frac{d_t}{\sum_{t=1}^3 d_t}. \quad (14)$$

The weights of all indices are calculated and denoted as row vector W :

$$W = [p_1, p_2, p_3]. \quad (15)$$

2.3. Process of the Model. The specific content of the model purposed in this paper is shown in Figure 1:

The evaluation and analysis model of the reconstructed urban intersections with added DLT lanes consists of three parts. The first part is collecting existing intersection data. The collected data are mainly divided into two categories: traffic data and geometry of existing intersection. Traffic data is the basic functional goal that the new type of intersection must achieve, and the geometry of an intersection is the basic engineering condition for the reconstruction.

The second part is to design improved schemes and obtain various evaluation parameters of existing and improved intersection schemes. PTV VISSIM is widespread in the field of microscopic traffic simulation, and SSAM is officially designated by the Federal Highway Administration (FHWA) as the agency's safety evaluation software [43, 44]. The SSAM is a safety analysis software and classic in traffic simulation. At the same time, it can be compatible with the vehicle trajectory file obtained by VISSIM simulations and can evaluate the safety situation in the simulation process by identifying and analyzing the vehicle trajectory. In addition, the SSAM software also has built-in statistical analysis functions based on the frequency and severity of conflicts, which can help designers design safe transportation facilities. During this part, PTV VISSIM and SSAM software are used to construct simulation models of traditional intersections and improved schemes for simulation experiments.

Finally, the last part is sensitivity analysis and schemes' evaluation. The collected traffic volume is the traffic volume of the existing intersection for a limited period of time, and the actual traffic volume is constantly evolving. Therefore, it is necessary to simulate the performance of various traffic volume combinations and analyze and compare the performance of each scheme.

The calculation of the scheme evaluation is as follows:

From the matrix of equation (5), we can extract the scheme number j of each element to generate a new matrix J :

$$J = \begin{bmatrix} j_{11} & j_{12} & j_{13} \\ j_{21} & j_{22} & j_{23} \\ \vdots & \vdots & \vdots \\ j_{n1} & j_{n1} & j_{n1} \end{bmatrix} \quad (16)$$

Corresponding to each combination of traffic volume (each row), the score of each scheme can be calculated.

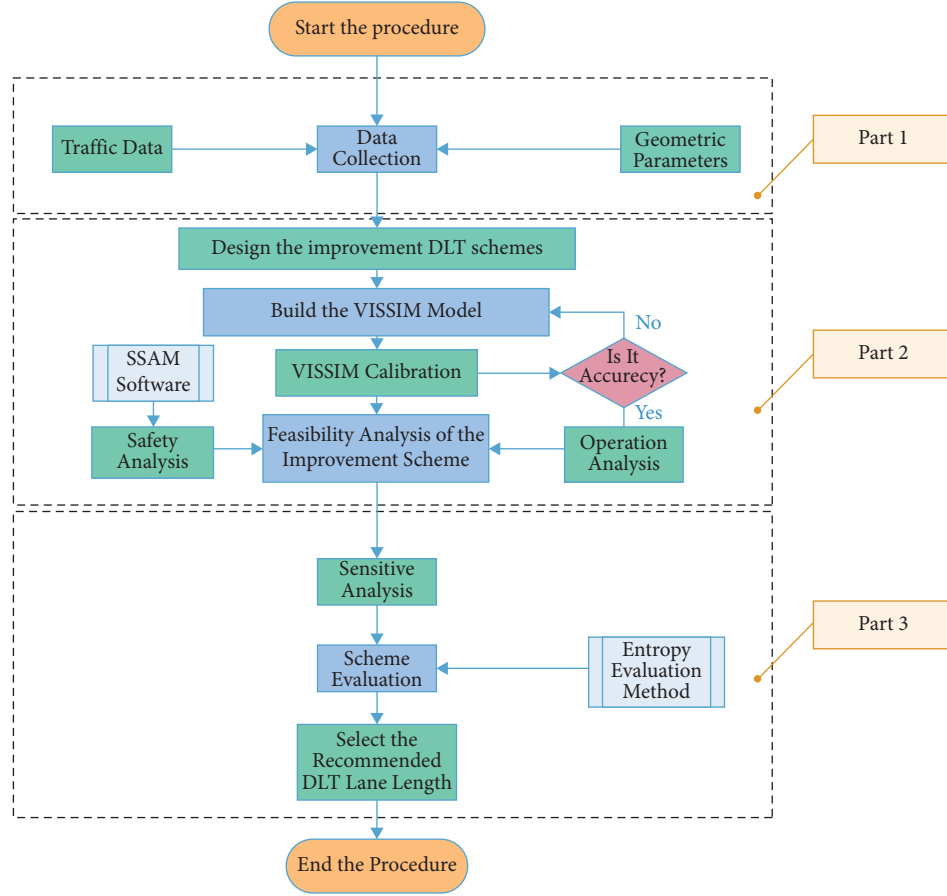


FIGURE 1: The flowchart of the evaluation and analysis model based on entropy evaluation method for the design process.

Taking the score of Scheme 1 as an example, the calculation is as follows:

$$x_{i1} = \begin{cases} \sum_{t=1}^3 p_t \cdot j_{1t}, & j_{1t} = 1, \quad i = 1, 2, 3, \dots, n, \\ 0, & j_{1t} \neq 1. \end{cases} \quad (17)$$

After calculating the score results of k schemes under n traffic volume conditions, we can generate a new matrix $X_{n \times k}$, as shown below:

$$X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1k} \\ x_{21} & x_{22} & \cdots & x_{2k} \\ \vdots & \vdots & x_{ij} & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nk} \end{bmatrix}. \quad (18)$$

Extract each row vector in matrix X as x_i :

$$x_i = [x_{i1}, x_{i2}, \dots, x_{ik}]. \quad (19)$$

Record the column with the highest score in each row vector as r_i :

$$r_i = j, \text{ if } x_{ij} = \max\{x_i\}, \quad j = 1, 2, \dots, k. \quad (20)$$

Finally, the column vector R of the recommended optimal length scheme under n different traffic volume conditions can be obtained:

$$R = [r_1, r_2, \dots, r_n]^T. \quad (21)$$

3. Case Study

In this part, we introduce the process and methodology of the case study in this paper.

3.1. Problem Statement. Xi'an, as an ancient capital with a thousand years of history, has become one of the cities with the most serious traffic problems in China. With the continuous development of modernization and urbanization, the number of vehicles of Xi'an city has continued to rise. According to China's congestion rankings on 2020 by Baidu map, Xi'an city is the fourth of this rank. Relieving the congestion problem of the intersections in Xi'an has the essential reference for other cities.

The focus is on an intersection of Xi'an city. As shown in Figure 2, the intersection is located where Electronic Road and Taibai South Road meet; as can be seen, this is the junction of two major arteries in Xi'an city. This junction consists of a signalized intersection. And, there is a lot of left-



FIGURE 2: The location of the investigated intersection. The investigated intersection is a traditional cross intersection.

turning traffic at this intersection, especially in the east-west direction.

Left-turning traffic has always been an essential issue of efficiency and safety at intersections. Thus, we focused on the left-turning traffic problem at this intersection in the east-west direction. Because of the heavy left-turning traffic of this intersection, the left-turning traffic flows from the east and west entrances often have high delays when passing intersections. It takes a lot of time for drivers to go through the intersection, especially during the evening rush hour. The manner in which to improve the operation of left-turning traffic flows at this intersection has become a problem that urgently needs to be solved.

3.2. Data Collection. Figure 3 shows the present intersection of Taibai South Road and Electronic Road in Xi'an. The traffic volume of left-turning traffic in the east-west direction is very high, especially in the evening rush hours. Left-turning vehicles in the east-west direction queue for a long time when passing through the intersection. At this intersection, the speed limit for the basic section in the east-west direction is 70 km/h and for the basic section in the north-south direction is 60 km/h. The east entrance of the intersection has five lanes from east to west, and there are four lanes from west to east; the west entrance of the intersection has five lanes from west to east and three lanes from east to west. There are collector-distributor lanes on both sides. The north entrance has five lanes from north to south and three lanes from south to north; meanwhile, the south entrance has five lanes from south to north and three lanes from north to south. The width of each lane is 3 m.

Step 1. Selecting data collection time.

Figure 4 shows a 24-hour traffic congestion coefficient map of Xi'an on 21 October 2020, indicating the peak and

low-traffic periods of the day. It can be seen from the figure that the morning peak time was from 7:00 a.m. to 9:00 a.m., while the evening peak time was from 5:00 p.m. to 7:00 p.m., and the noon trough time was from 11:00 a.m. to 1:00 p.m. The experimental data were collected in one hour during the morning and evening peak and midday trough periods.

Step 2. Preparing experimental equipment.

Data collection required the use of Unmanned Aerial Vehicles (UAV), mobile phones, radars, laptops, radar data and power cables, a drone battery and a controller, and a mobile power supply (for radar and laptop charging).

Step 3. Choice of instrument installation position.

As shown in Figure 5, when measuring the speed of vehicles, in order to ensure that the speed of each lane of the entrance in each direction of the intersection could be measured as accurately as possible, the installation of the radar should be very close to the direction of the vehicles, and the erection position on the roadside should be as high as possible. The height of the radar should exceed 2 m to ensure that there are no fixed obstacles in the measurement range; when there was a suitable overpass near the intersection, it was best to measure the vehicle directly on the overpass.

Step 4. Specific experimental operations.

First, we turned on the remote control of the drone; then, the drone should be turned on. The power cord of the radar and laptop were connected to the mobile power supply to ensure normal power supply, and the data cable of the radar was inserted into the USB port of the laptop. Then, the following items were clicked on in sequential order: "Check" to see if the radar is working properly, "Settings" to synchronize the time of the radar and the notebook, and the original data to delete it. Finally, the investigation could be started, and the radar would officially begin to measure data.

Step 5. Inspecting during data collection.

During the radar data collection period, the real-time inspection interface of the software open should always be retained, and the radar measurement should be checked every 5 mins to ensure it is normal.

Step 6. Processing analysis.

When the data collection was finished, the data were downloaded as a spreadsheet with a designated name, and the video data taken by the drone were copied. Tables 1–3 show the traffic volume data during the morning, midnoon, and evening periods. As shown in the table, in the evening rush hour, the east-west entrance left-turn traffic has the highest proportion and is the most congested period of urban traffic.

3.3. Design of Improvement DLT Schemes. Left-turning traffic flows at intersections often conflict with oncoming direction traffic flows. To improve the efficiency of left-turning traffic in the east-west direction, adding DLT lanes is one of the



FIGURE 3: The actual situation of the investigated intersection. The investigated intersection is located in the center of Xi'an Yanta district. The east-west street is Dianzi Road, and the north-south street is Taibai South Road.

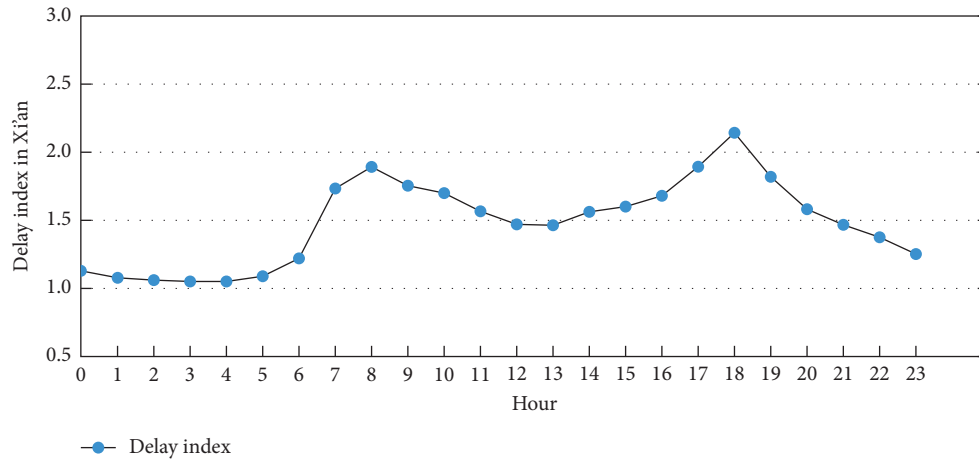


FIGURE 4: The congestion index in Xi'an on 2020.10.21. The real-time congestion index can be gathered from Autonavi Company web page at <https://report.amap.com/detail.do?city=610100>.

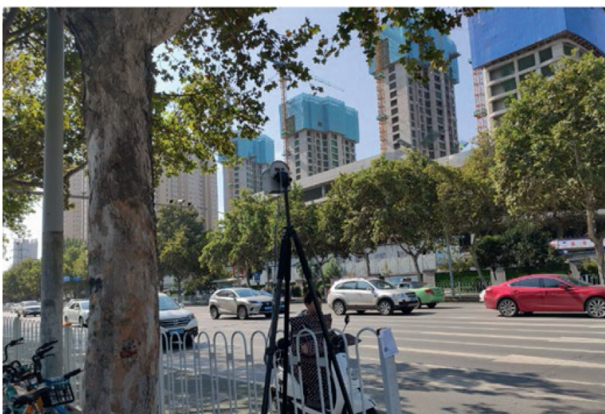


FIGURE 5: The radar survey at the intersection. This figure was taken at 34.2174°N, 108.9125°E. The figure shows the radar erect at the south entrance of the intersection. When measuring, the radar speed measurement direction must be as parallel as possible to the driving direction of the vehicles and facing the front of the vehicles and must the same for the other three entrances.

effective solutions. Advanced design schemes for extending the DLT lanes on the outside of the current east-west entrance were put forward in this paper.

As shown in Figure 6, the specific design method of the advanced schemes with added DLT lanes is as follows:

- (i) Survey and collect the design data and construction parameters of existing intersections, including the number of lanes, lane width, transition section length, and existing signal timing plans. Figure 6(a) shows the existing intersection. There is a 60 m long transition section at the west entrance. Both the east entrance and the west entrance include two left-turn lanes. According to the usage conditions of the model, the above parameters are consistent with the status quo.
- (ii) As shown in Figure 6(b), keep the number of basic lanes and the number of left-turn lanes at the existing intersection unchanged, and change the

TABLE 1: Collected data during the peak hour of 8:00 a.m. to 9:00 a.m. on 21 October 2020.

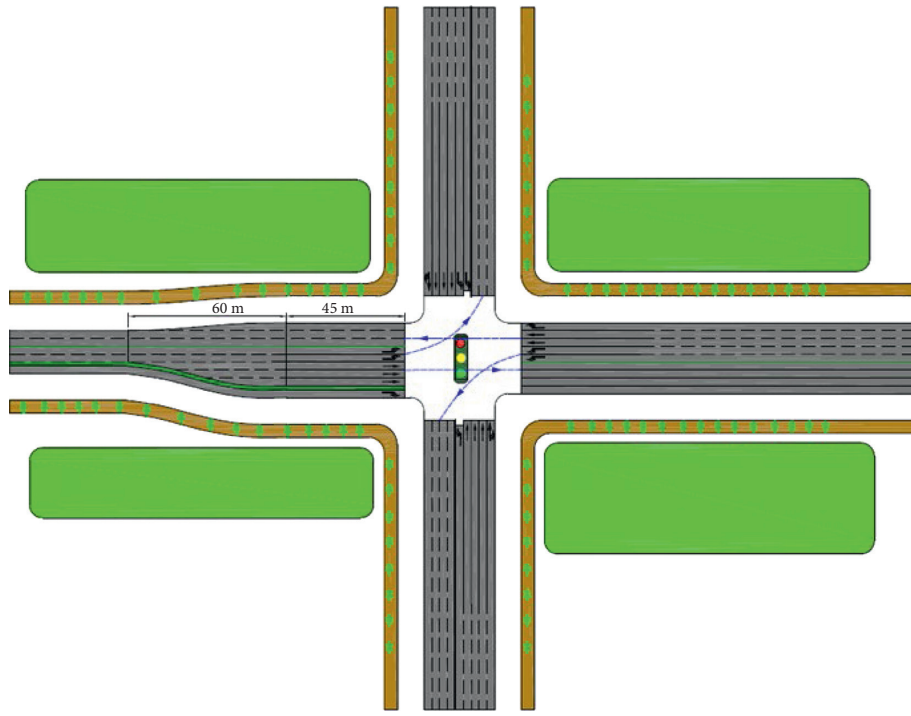
	Flow	Flow number	Car (veh/h)	Truck (veh/h)	Bus (veh/h)	Max speed (km/h)	Min speed (km/h)	Average speed (km/h)
East entrance	Through	1	546	12	42	60	0	19
	Left turn	2	366	12	6	84.5	0	25
	Right turn	3	84	0	36	62	0	21.5
West entrance	Through	4	708	0	48	63	0	22
	Left turn	5	372	0	0	69	0	29
	Right turn	6	84	0	0	62.8	0	24
South entrance	Through	7	1458	24	48	52.4	0	12
	Left turn	8	132	6	0	46	0	23
	Right turn	9	246	6	0	50	0	13
North entrance	Through	10	1554	12	48	58.5	0	18
	Left turn	11	150	0	18	50	0	25
	Right turn	12	294	6	0	53.5	0	18.5

TABLE 2: Collected data during the peak hour of 11:00 a.m. to 12:00 a.m. on 21 October 2020.

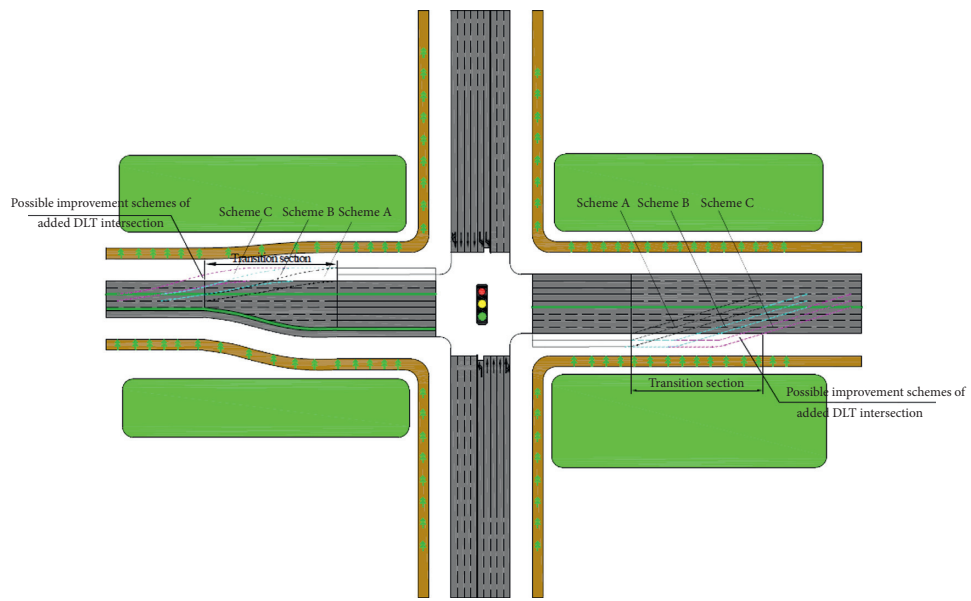
	Flow	Flow number	Car (veh/h)	Truck (veh/h)	Bus (veh/h)	Max speed (km/h)	Min speed (km/h)	Average speed (km/h)
East entrance	Through	1	332	21	28	62	0	18
	Left turn	2	282	21	7	83.5	0	25
	Right turn	3	233	14	28	64	0	20
West entrance	Through	4	600	56	49	60	0	24.5
	Left turn	5	268	0	0	70	0	30
	Right turn	6	106	21	0	62.6	0	25
South entrance	Through	7	1956	49	57	52.2	0	9.9
	Left turn	8	148	0	0	48	0	22
	Right turn	9	282	21	0	51	0	13
North entrance	Through	10	2196	49	71	56.5	0	17.9
	Left turn	11	268	7	21	52	0	24.5
	Right turn	12	226	0	7	54.5	0	18

TABLE 3: Collected data during the peak hour of 5:00 p.m. to 6:00 p.m. on 21 October 2020.

	Flow	Flow number	Car (veh/h)	Truck (veh/h)	Bus (veh/h)	Max speed (km/h)	Min speed (km/h)	Average speed (km/h)
East entrance	Through	1	474	20	60	54	0	18.7
	Left turn	2	294	7	0	63.7	0	26.6
	Right turn	3	160	20	27	56	0	21.7
West entrance	Through	4	787	20	47	61.2	0	29.5
	Left turn	5	313	13	0	70.6	0	37
	Right turn	6	133	7	0	63	0	31.5
South entrance	Through	7	1481	47	87	22.7	0	9
	Left turn	8	100	7	7	29.9	0	21.6
	Right turn	9	253	7	0	24	0	12
North entrance	Through	10	1661	67	67	46.8	0	15.1
	Left turn	11	220	0	40	54	0	20.16
	Right turn	12	253	0	0	48	0	17.2



(a)



(b)

FIGURE 6: Continued.

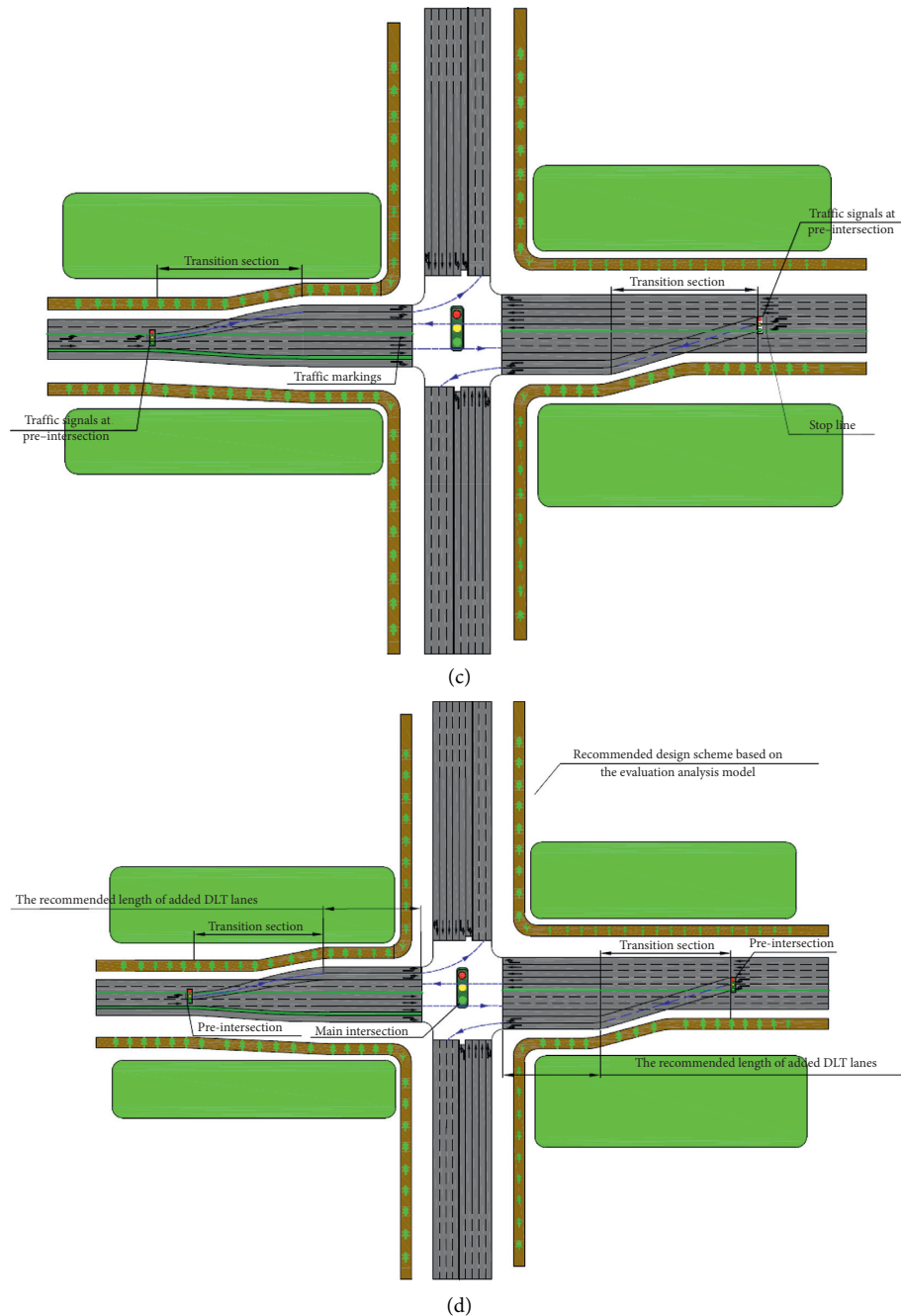


FIGURE 6: The design method of the DLT scheme. (a) Existing traditional situation. (b) Put forward some possible improvement schemes. (c) Traffic facilities design. (d) Determine the final recommended design scheme.

left-turn lanes at the original entrance to allow through vehicles to pass. Then, add DLT lanes to the outside of the opposite straight lanes, and set a smooth transition section to ensure a reasonable transition. After that, put forward some possible improvement reconstruction schemes according to the actual land conditions and the length of the existing left-turn lane.

(iii) Complete the design of traffic facilities for each possible improvement scheme, including the design

of traffic signs and markings, main intersection signal timing, and preintersection signal timing. This safeguards that the vehicles can safely and smoothly pass through the intersection (Figure 6(c)).

(iv) Analyze the operation and safety of possible improvement schemes and the existing intersection. Then, the sensitivity analysis and evaluation of possible improvement schemes by using the evaluation and analysis model based on the entropy

evaluation method are carried out. Finally, the recommended design scheme with the recommended length of the extra DLT lane is obtained (Figure 6(d)).

The results of simulation experiments for each scheme are shown in Figure 7.

In the case study of this paper, there were two DLT lanes at the east entrance and also two at the west entrance. It kept the number of DLT lanes consistent with the left-turn lanes in the current situation. The main variable included in this study was the length of the DLT lanes. The present length of the entrance was 45 m, while the longest length that could be expanded at this stage was 145 m. The lengths of the DLT lanes in schemes 1–5 changed at equal intervals, which are 45, 70, 95, 120, and 145 m.

After adding DLT lanes at the intersection, was it effective? If there were improvements in some fields, what were the percentages of advancement? What was the recommended length of the DLT lanes on the operating conditions? To solve these problems, we used the entropy-based model proposed in this paper for definite evaluation and analysis.

3.4. VISSIM Calibration. The parameters of the VISSIM simulation model needed to be calibrated before the simulation of an intersection improved scheme, so as to assure the accuracy of the VISSIM simulation model. The specific calibration of the VISSIM model is as follows. First, the parameters were input into the VISSIM software according to the current field investigation at the intersection of South Taibai Road and Dianzi Road, such as operating speed and vehicle composition proportion. Next, we selected the governing car following model in VISSIM. The governing car following model is the classic model proposed by German Professor Wiedemann, which belongs to the psycho-physiological model. There are two types in the Wiedemann model by default: Wiedemann99 and Wiedemann74. Previous studies found that Wiedemann74 was more consistent with the vehicle behavior on urban roads, and the Wiedemann74 model was also selected for simulation in this study. After selecting the Wiedemann74 model, key parameters such as the average stopping distance and the desired time headway were modified, relying on the actual survey data. Then, the simulation was calibrated with the capacity indicators, which could comprehensively reflect the similarity between the effect of the whole simulation model and the actual intersection situation.

Then, the VISSIM simulation model was run 30 times under each circumstance, and the average value of the traffic volume of each inlet and direction was taken. Finally, the relationship between the traffic volume data obtained from the simulation and the traffic volume data obtained from the actual survey at the intersection was compared. The error was characterized by the mean absolute percent (MAPE) value. According to relevant studies, when the MAPE value of the traffic volume in each direction is less than 15%, the simulation results of the VISSIM simulation model are considered to be effective. The MAPE was calculated as follows:

$$\text{MAPE} = \frac{1}{q} \sum_{i=1}^q \left| \frac{C_v^i - C_f^i}{C_f^i} \right|, \quad (22)$$

where q denotes the 12 different flows in this study, C_v^i is the capacity (number of vehicles crossing the intersection per unit time) simulated in the VISSIM model (veh/h), and C_f^i is the capacity of the investigation (veh/h).

After running the realistic model of the intersection in VISSIM 30 times, the calculated results of the average hourly traffic volume, the actual collected traffic volume at the intersection, and the MAPE value are summarized in Table 4.

As shown in Table 4, the MAPE value of traffic flow in the 12 directions was 6.43% in total. Because the traffic volume error in total was less than 15%, the constructed model met the requirements of simulation accuracy. Thus, the calibration accuracy of the VISSIM model was reasonable [45, 46].

3.5. Operation Analysis. The east-west direction is the main road. Because of the high volume of left-turning traffic during peak hours, it is also the direction for the implementation of the DLT schemes; the north-south direction is the secondary road, and the north-south direction remains the actual crossroad mouth form. It is the main development goal of the current intersection that increasing traffic capacity and improving delays. And, environmental protection is also one of the common goals of today's engineering projects. Therefore, the indexes selected in this case were the capacity, delays, number of stops, and NOx emissions.

Capacity refers to the number of vehicles passing through the entire intersection in a unit hour. Delays mean the difference between the actual travel time and the expected travel time. In this study, delays include stop delay and travel delay, and it is calculated as follows:

$$D = d_1 + d_2, \quad (23)$$

where D denotes the total delay, d_1 denotes the stop delay, and d_2 denotes the travel delay. The number of stops indicates the average number of stops for each vehicle passing the intersection; NOx emissions represent the total amount of nitrogen oxide emitted by vehicles passing through the intersection within an hour. There were five groups of advanced models and one existing intersection model, and each group ran 30 random seeds to obtain more scientific and reliable results from the statistical significance. Therefore, 180 simulations were performed in the case of existing traffic volume. The results are stated below in Table 5.

It was evident that, in the simulation results, Scheme 0 was much higher than the other situations in terms of the delays, number of stops, and NOx emissions. They all had a similar trend, with the degree of development gradually increasing from Scheme 1 to Scheme 4, but with no distinction when the improvement changes from Scheme 4 to Scheme 5. This indicated that the difference resulting from changing the length of the added DLT lanes had a regular influence on the results. However, the capacity for each

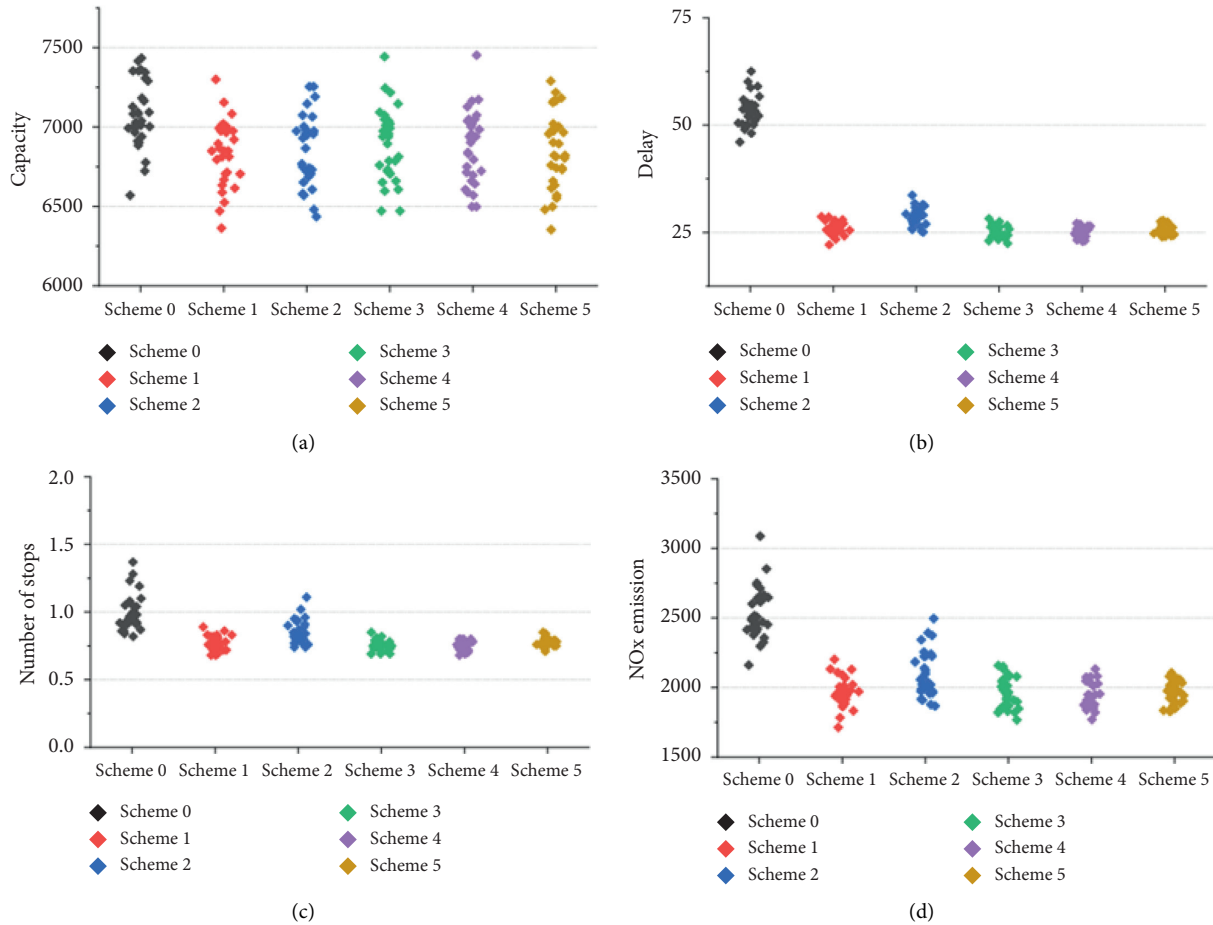


FIGURE 7: Simulation results of the experiments. (a) Capacity. (b) Delays. (c) Number of stops. (d) NOx emissions.

TABLE 4: VISSIM simulation calibration results with the collected data. MAPE, mean absolute percent.

Flow number	Investigate capacity (veh/h)	Simulated capacity (veh/h)	Individual MAPE (%)	Total MAPE (%)
$i = 1$	554	486	12.27	
$i = 2$	301	270	10.30	
$i = 3$	207	198	4.35	
$i = 4$	847	756	10.74	
$i = 5$	326	288	11.66	
$i = 6$	140	144	2.86	
$i = 7$	1615	1971	22.04	6.43
$i = 8$	114	108	5.26	
$i = 9$	260	279	7.31	
$i = 10$	1795	2088	16.32	
$i = 11$	260	225	13.46	
$i = 12$	253	288	13.83	

scheme did not differ between the six schemes. This meant there were no obvious advantages between the different schemes under the present investigated traffic volume.

3.6. Safety Analysis. A complete evaluation should conclude with both an operation and safety evaluation. With the calibration of VISSIM already completed, the microscopic vehicle simulation of the VISSIM model already has a high degree of agreement with the actual

situation. We took the number of cross-conflicts present in the one-hour intersection vehicle operation video investigated under actual conditions and compared it with the safety simulation results. Then, we found that there were 4 cross-conflicts in the video. The deviation between simulation and reality was within the acceptable range. Therefore, the simulation results of SSAM were considered reliable.

The safety analysis was applied to the six schemes with evening peak hours' data. The results are shown below.

TABLE 5: VISSIM results of the six selected schemes with evening peak hour data.

Item	Capacity (veh/h)		Average delays (s)		Number of stops		NOx emission (gallons/h)	
Scheme 0	7083		53.23		0.99		2543.31	
Scheme 1	6840	-3.43%	25.9	51.34%	0.77	22.22%	1965.42	22.72%
Scheme 2	6849	-3.30%	28.74	46.01%	0.85	14.14%	2098.35	17.50%
Scheme 3	6885	-2.80%	25.24	52.58%	0.76	23.23%	1955.52	23.11%
Scheme 4	6876	-2.92%	25.02	53.00%	0.75	24.24%	1943.46	23.59%
Scheme 5	6849	-3.30%	25.46	52.17%	0.77	22.22%	1963.8	22.79%

Table 6 shows the results of the safety analysis of the six schemes, including the existing traditional scheme. The results show that compared to the new type of intersection with additional DLT lanes, the traditional intersection had significantly higher safety risks in the rear end and lane change. Compared to the proposed improvement plan, the value for the item "rear end" was 70%~100% higher, and the value for the item "lane change" was 100%~200% higher than the improvement plan. However, due to the sub-intersections set up at DLT intersections, the traditional intersections in the column of the crossing had advantages and their values were lower than the other five improved schemes. In general, the sum of the three potential safety risks shows that the improved DLT intersection scheme should outperform the existing traditional intersections in terms of safety performance. Moreover, the five improved schemes achieved relatively close safety effects.

4. Results and Discussion

4.1. Sensitivity Analysis. Sensitivity analysis can reflect the improvement ratio of each advancement scheme compared to the present scheme. Capacity, delays, number of stops, and NOx emissions are the four indexes. Based on the maximum capacity under different lane numbers in the Highway Capacity Manual (HCM) [47], multiplying by the certain coefficients, different traffic volume parameters were obtained. All volume parameters in VISSIM are shown in Table 7.

The next step was to calculate the signal timing plan. As Figure 8 shows, the signal timing of the existing intersection consists of four stages and the phasing of the improvement schemes consists of three stages. The added DLT lanes eliminate the conflicts between left-turning vehicles and straight vehicles at the main intersection, and the straight and left-turning vehicles on the same entrance at the main intersection can be released at the same time.

Synchro7 is a signal optimization software using the NEMA signal timing structure. We input the number of signal stages, the traffic volume of each direction, and lane allocation plan in synchro7; then, the synchro7 calculated and output the recommended signal timing plans. Afterward, we input the signal timing calculated by synchro7 into VISSIM.

For the development schemes of the different DLT lanes lengths, the improvements in the indicators between them and the present intersection scheme were compared and analyzed one by one. The DLT lane length was 45 m in

Scheme 1, 70 m in Scheme 2, 95 m in Scheme 3, 120 m in Scheme 4, and 145 m in Scheme 5. And, the traditional intersection was represented as Scheme 0.

Figure 9 shows the improvement in the capacity under the five development schemes. As shown in this picture when the traffic volume in the east-west direction was less than 1029 veh/h, there was no obvious advancement effect of the five advanced schemes compared to the traditional cross intersection. When the east-west traffic volume continued to develop, the advantage of setting DLT lanes began to show. In Scheme 1, when the east-west traffic volume reached the maximum value of 3430 veh/h in the test process, the effect of improving the capacity was most obvious, reaching a maximum of 22%. In Scheme 2, when the east-west traffic volume reached the maximum combined traffic volume of 3430 veh/h, the enhancement effect peaked at 31.5%. In the case of Scheme 3, as the input traffic volume in the east-west direction increased after exceeding 1029 veh/h, the improvement rates also increased. In this simulation experiment, when the maximum combined traffic volume was 3430 veh/h, the maximum percentage of the enhancement effect was able to reach 36%. For Scheme 4, the simulation results show that, with the growth of the traffic volume in the east-west direction after exceeding 1029 veh/h, the advantages of the Scheme 4 became gradually clear. When the traffic volume in the east-west direction reached the maximum value of 3430 veh/h, the maximum percentage of the improvement effect was able to reach 42%. The maximum improvement percentage under Scheme 5 was 36%, which was generated when the input traffic volume in the east-west direction was 3430 veh/h and the input traffic volume in the north-south direction was 3656 veh/h.

In summary, there was no improvement in the capacity of all five development schemes in the case of low traffic in the east-west direction. When the traffic volume was greater than 1029 veh/h, the enhancement effect gradually increased as the traffic volume increased. From Scheme 2 to Scheme 5, as the lengths of the DLT lanes increased, the maximum improvement effect of the capacity gradually increased, while in terms of the change from Scheme 4 to Scheme 5, as the lengths of the DLT lanes increased from 120 to 145 m, the maximum advancement effect of the capacity slightly reduced.

Figure 10 shows the advance in the average vehicle delay index under the five scenarios. In the case of Scheme 1, the average vehicle delay upgrade ratios were concentrated between 5% and 30% and the maximum improvement percentage was 31%. In Scheme 2, the average delay upgrade

TABLE 6: Safety analysis of the six simulations by the surrogate safety assessment model (SSAM).

Scheme number	Item	Crossing	Rear end	Lane change	Total
0	Present intersection	5	120	52	177
1	DLT 45 m	9	69	25	103
2	DLT 70 m	15	71	19	105
3	DLT 95 m	12	60	20	92
4	DLT 120 m	8	63	19	90
5	DLT 145 m	13	65	17	95

TABLE 7: The traffic volume combinations of the sensitivity analysis (veh/h).

Item	Value
East/west volume	686/1029/1372/1715/2058/2401/2744/3087/3430
North/south volume	1828/2285/2742/3199/3656

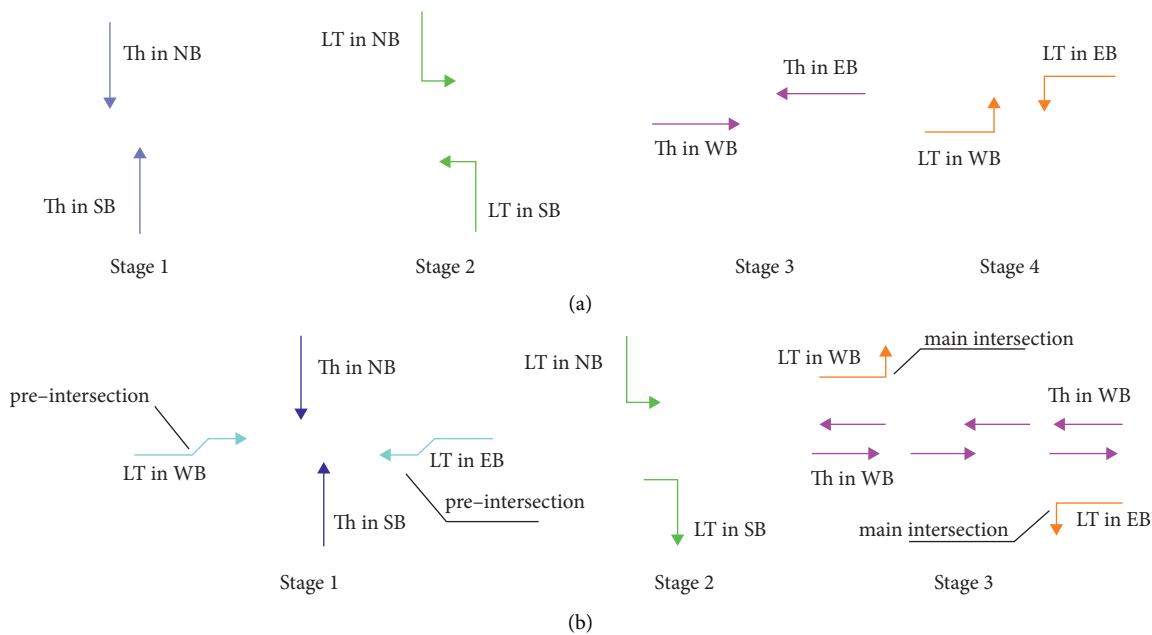


FIGURE 8: Signal timing stage diagram for existing intersection and improvement schemes with added DLT lanes. (a) Existing intersection. (b) Improvement intersections with added DLT lanes. EB, WB, NB, and SB denote Eastbound, Westbound, Northbound, and Southbound, respectively. LT and Th denote Left-Turn and Through, respectively.

percentages were concentrated between 10% and 35%. The maximum improvement percentage was 37% when the traffic volume in the east-west direction was 2744 veh/h and 1828 veh/h in the north-south direction. In Scheme 3, there was also positive growth in each traffic volume. The average percentage of the vehicle delay improvement was up to 41% under the same traffic volume with Scheme 2. In the case of Scheme 4, the average delay upgrade percentages were mainly in the range of 15%–45%. And, the highest advancement percentage was 46.4% also in the case where Scheme 2 reaches its maximum improvement ratio. In Scheme 5, the advance ratios were also concentrated between 15% and 45% compared to a traditional intersection and reached a maximum of 45.8% in the same case as Scheme 2.

In other words, the comparative analysis showed that, as the added DLT lanes gradually increased from 45 to 120 m,

the improvement percentage of the average vehicle delay also increased. When the DLT lanes exceeded 120 m and increased to 145 m, there was a slight decrease.

Figure 11 shows the improvement in the number of stops for the five advanced schemes compared to the traditional situation. For all the five schemes when the input traffic volume in the east-west direction was less than 1029 veh/h, a development in the average number of stops was not evident, but when the traffic volume in the east-west direction exceeded 1029 veh/h and gradually increased, the improvement effect of the scheme became gradually distinct. In the case of Scheme 1, the number of stops' upgrade ratios was concentrated between 5% and 40%. When the traffic volume in the east-west direction was 2744 veh/h and 1828 veh/h in the north-south direction, the advancement percentage was up to 43.8%. In Scheme 2, the range of the upgrade ratios was similar to that of Scheme 1, while the

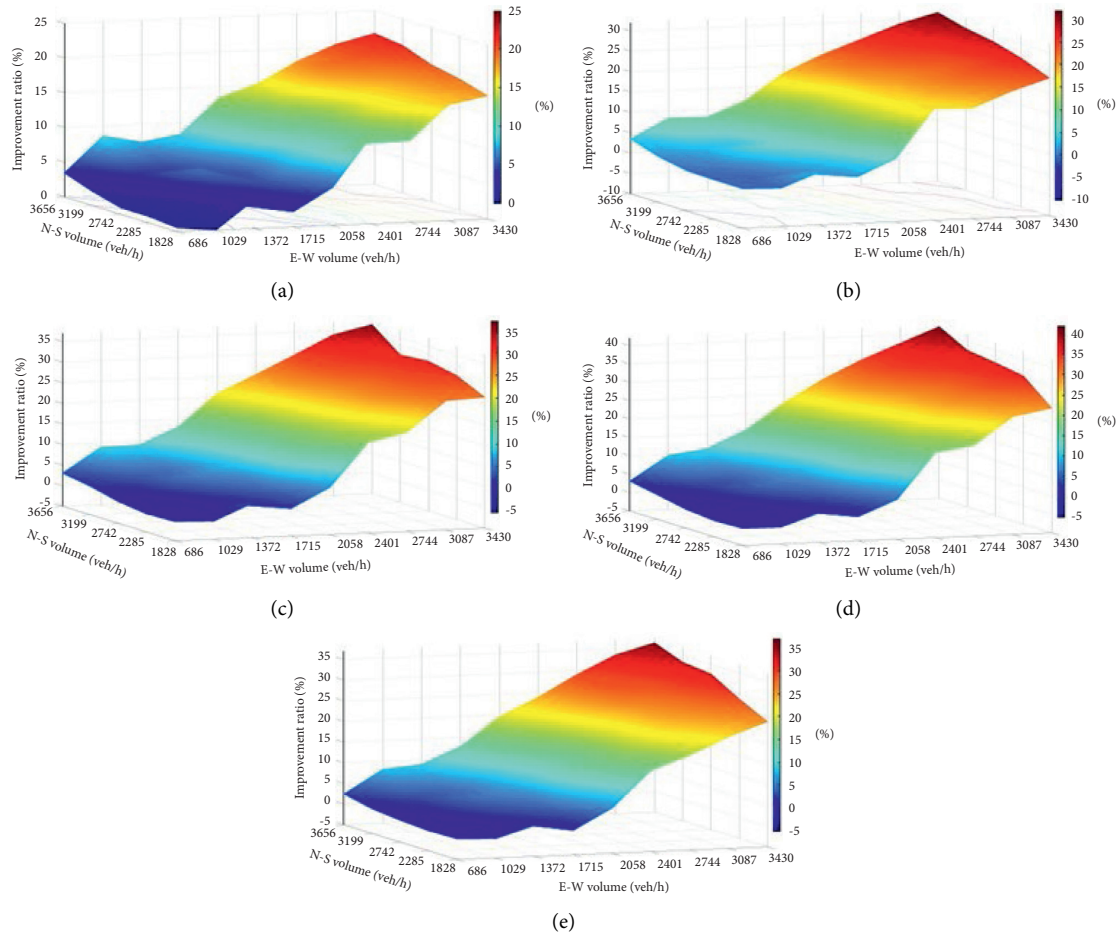


FIGURE 9: Improvement ratios of capacity. (a) Scheme 1. (b) Scheme 2. (c) Scheme 3. (d) Scheme 4. (e) Scheme 5. E-W volume means the same traffic volume of east entrance and west entrance. N-S volume means the same traffic volume of north entrance and south entrance.

development percentage was up to 47.6% at the same traffic volume as Scheme 1. In the case of Scheme 3, the average number of stops' improvement percentages is mainly in the range of 5%–50%, and the upgrade ratio was up to 53.8% at the same traffic volume as Scheme 2. In Scheme 4, the upgrade ratio reached a maximum of 60% under the same traffic volume as the three previous schemes. In Scheme 5, like the four advanced schemes mentioned above, the ratios increased with the increase in east-west traffic volume. The biggest enhancement percentage occurred in the same circumstance. And, the increasing percentage at this time was 55.7%.

In short, for all five advanced schemes, regardless of the lengths of the DLT lanes, there was no significant growth in the number of stops for vehicles when the traffic volume in the east-west direction was low. On the one hand, as the traffic volume in the east-west direction gradually increased, advantages of increasing the DLT improvement program gradually emerged. On the other hand, a horizontal comparison showed that, as the lengths of the DLT lanes increased from 45 to 120 m, the enhancement effect increased with the increase in length; however, when the lengths of the DLT lanes increased from 120 to 145 m, and there was a small decrease in the improvement in the average number of stops.

Figure 12 shows the improvements in the NOx emissions indicators of the five progressive schemes compared to the traditional intersection. It was found that the enhancement effects of the five improved schemes were roughly the same as those of the current traditional intersections. The upgrade effects were not obvious in the 45 combinations of traffic volume, and negative changes were shown in most cases. The maximum improvement percentages of the schemes appeared when the traffic volume in the east-west direction was 2058 veh/h and 1828 veh/h in the north-south direction. The improvement percentages gradually increased from Scheme 1 to Scheme 4, i.e., 10.6%, 12.9%, 18.2%, and 21%. Meanwhile, the maximum percent of Scheme 5 was 17.6%. Generally speaking, because the traffic capacity of the advanced scheme was higher than that of the traditional scheme, the nitrogen oxide content emitted by vehicles in the entire intersection area did not significantly improve compared to the traditional intersection scheme for the same duration. There was even a slight increase in emissions.

Corresponding to the east-west traffic flow with DLT lanes, this part focuses on comparing and analyzing the differences in the travel time of the east-west imported left-turning vehicles under different traffic volume combinations for the east-west traffic flow with DLT lanes.

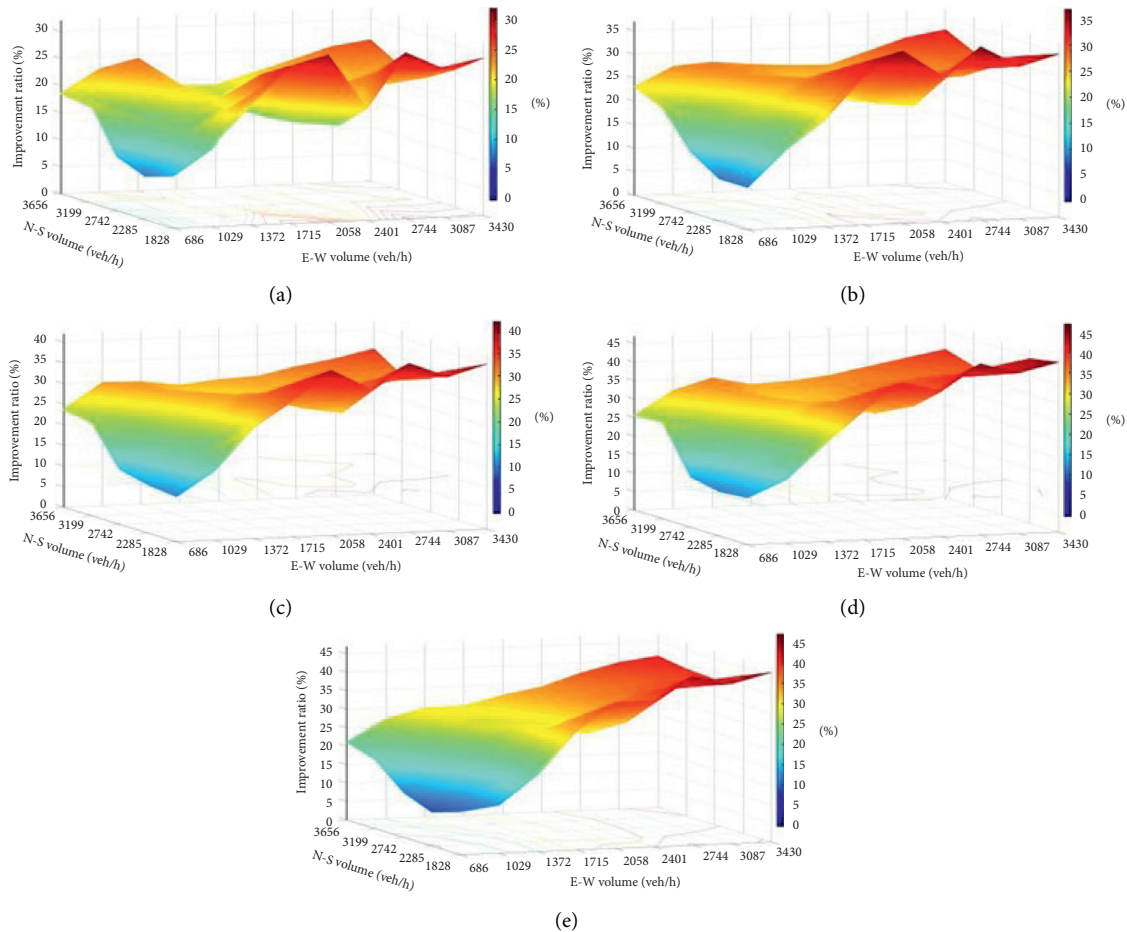


FIGURE 10: Improvement ratios of delays. (a) Scheme 1. (b) Scheme 2. (c) Scheme 3. (d) Scheme 4. (e) Scheme 5. E-W volume means the same traffic volume of east entrance and west entrance. N-S volume means the same traffic volume of north entrance and south entrance.

Figure 13 shows the travel time of left-turning vehicles at the west entrance. The travel time was recorded from the point where the vehicle reached 100 m before the main intersection to the point where the vehicle drove 100 m away from the main intersection. In Scheme 1, the simulation experiment results showed negative changes under all 45 traffic volume combinations. This is because the DLT lane has subintersections and left-turning vehicles pass through the entire level intersection, meaning that vehicles need to pass through two signal-controlled intersections. On the contrary, it also showed that the length of the 45 m DLT lane was set too short to reduce the travel time of left-turning vehicles. In Scheme 2, the travel time of left-turning vehicles at the west entrance could be obviously improved under most traffic conditions, with the highest advancement percentage reaching 30%. In Scheme 3, the improvement effect of the driving time of left-turning vehicles at the west entrance was more distinct than that in the second scheme. The enhancement percentage under most traffic conditions was more than 30%, and the largest advancement percentage reached 62.5%. In Scheme 4, the enhancement percentage of the driving time of left-turning vehicles at the west entrance also exceeded 30% in most cases, and the maximum improvement percentage reached 65.8%. When the lengths of

the DLT lanes increased to 145 m, the maximum enhancement percentage of the driving time of left-turning vehicles at the west entrance was 55.8%.

Figure 14 shows the improvement in the driving time of left-turning vehicles at the east entrance. In Scheme 1, the driving time of left-turning vehicles at the east entrance was similar to the situation at the west entrance, and Scheme 1 showed a negative change compared to the traditional scheme. Additionally, the largest negative upgrade ratio even reached -90% . In Scheme 2, the driving time of left-turning vehicles showed a positive development in most cases, and the percentage of enhancement was concentrated in the range of $10\% \sim 20\%$. In the case of Scheme 3, the advancement effect of the driving time of left-turning vehicles was more obvious. Under various traffic volume combinations, the driving time of left-turning vehicles improved, and the growth percentage was concentrated in the $30\% \sim 50\%$ range. In the case of Scheme 4, the driving time of left-turning vehicles also improved under all circumstances, and the increasing percentage was concentrated in the $40\% \sim 60\%$ range. When using Scheme 5, the development under the 45 traffic volume combinations was positive, and the advancement percentage was mostly between 40% and 70%.

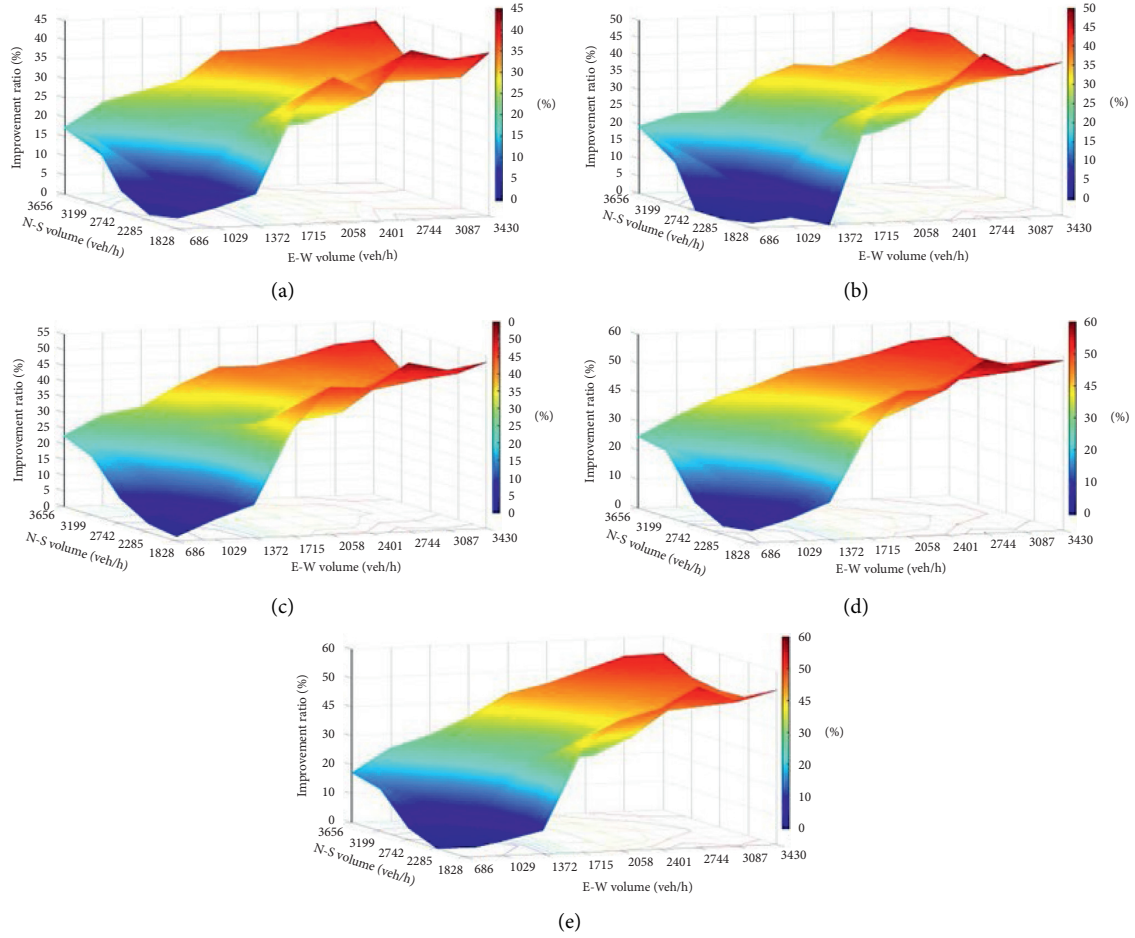


FIGURE 11: Improvement ratios of the number of stops. (a) Scheme 1. (b) Scheme 2. (c) Scheme 3. (d) Scheme 4. (e) Scheme 5. E-W volume means the same traffic volume of east entrance and west entrance. N-S volume means the same traffic volume of north entrance and south entrance.

4.2. *Weight of Four Indexes Based on Entropy Method.* As mentioned above, four commonly used indicators were selected for the scheme comparison and selection of this part:

$$\begin{aligned}
 C &= C_{45 \times 6}, \\
 D &= D_{45 \times 6}, \\
 S &= S_{45 \times 6}, \\
 N &= N_{45 \times 6},
 \end{aligned}
 \tag{24}$$

where C denotes the matrix of capacity, D denotes the matrix of average vehicle delays, S denotes the matrix of the number of stops, and N denotes the matrix of NOx emissions.

The above four indexes' simulation results of the six schemes with 45 different traffic combinations were calculated step by step according to equations (2)–(15). The weights of the four indexes are shown in following Table 8:

4.3. *Scheme Comparison.* According to equation (6), processing the simulation results generates the matrix J . And, in this case study, the first row J_1 of the matrix J could be obtained as follows:

$$J_1 = [2, 5, 5, 1]. \tag{25}$$

Then, the first row X_1 of the matrix X can be obtained from equations (17) and (18) as follows:

$$X_1 = [0.3341, 0.1595, 0, 0, 0.5064, 0]. \tag{26}$$

The first column represents the score of Scheme 0, the second column represents the score of Scheme 1, and so on; the last column represents the score of Scheme 5. And, the results in the first row show the recommended length of the DLT lanes for Scheme 4 under the first traffic volume combination.

The first element of the matrix R is

$$r_1 = 4. \tag{27}$$

Finally, the matrix R contains 45 scheme numbers that represent the best scheme for each combination of traffic volumes. Matrix R was transposed into a new 9×5 matrix R' , and the final result is shown in Figure 15:

Figure 15 shows a comparison of the six scenarios for all 45 traffic volume combinations, and the color block in the figure shows a clear regularity. In the lower-left corner,

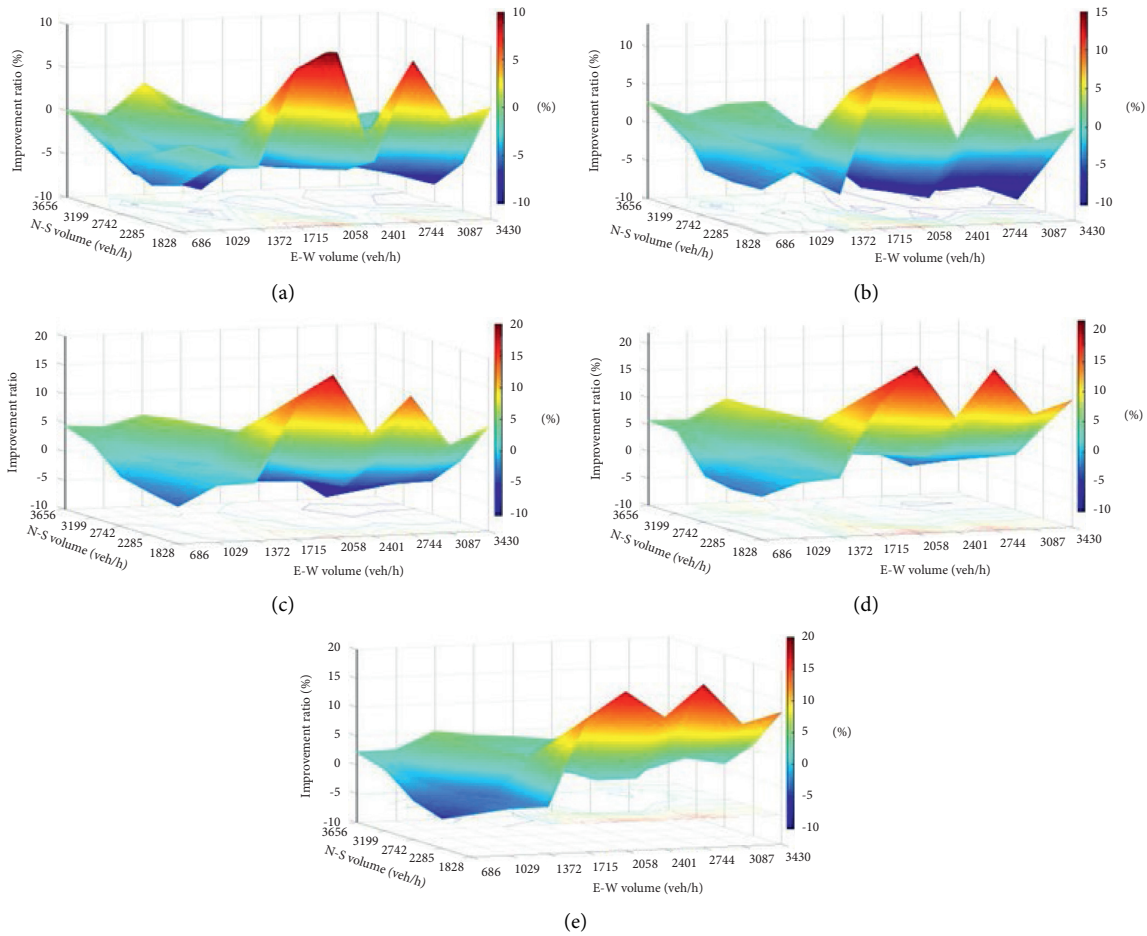


FIGURE 12: Improvement ratios of the NOx emissions. (a) Scheme 1. (b) Scheme 2. (c) Scheme 3. (d) Scheme 4. (e) Scheme 5. E-W volume means the same traffic volume of east entrance and west entrance. N-S volume means the same traffic volume of north entrance and south entrance.

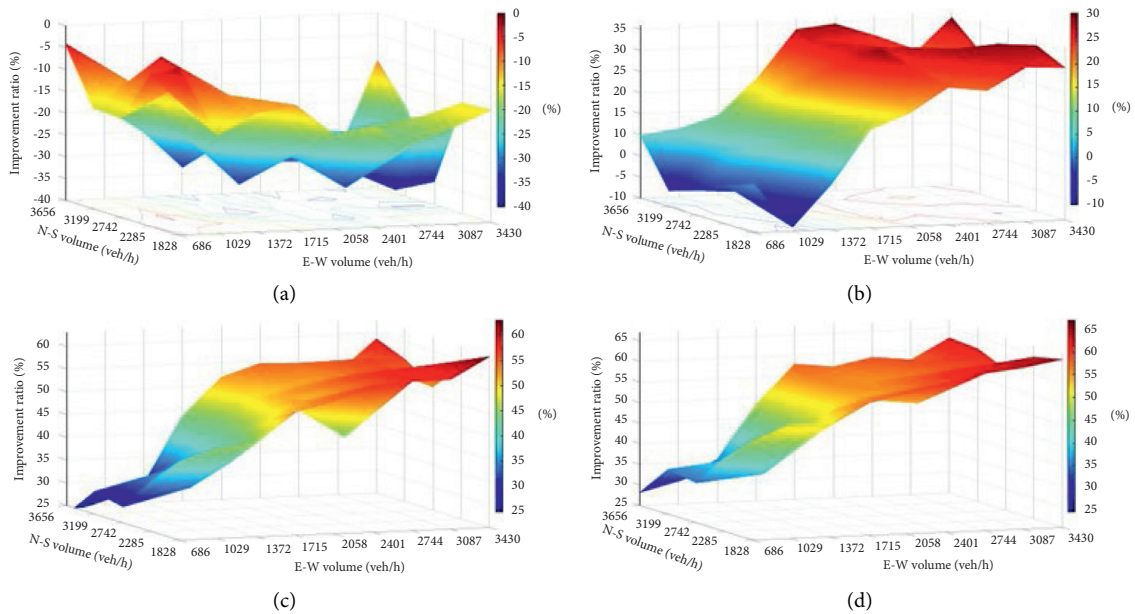


FIGURE 13: Continued.

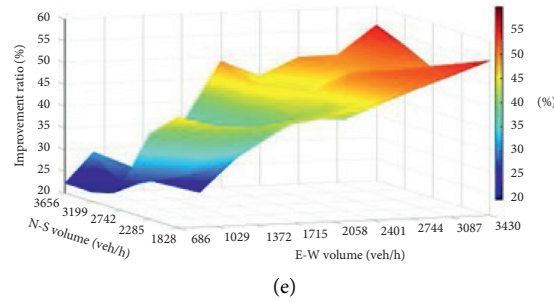


FIGURE 13: Improvement ratios of the left-turning time at the west entrance. (a) Scheme 1. (b) Scheme 2. (c) Scheme 3. (d) Scheme 4. (e) Scheme 5. E-W volume means the same traffic volume of east entrance and west entrance. N-S volume means the same traffic volume of north entrance and south entrance.

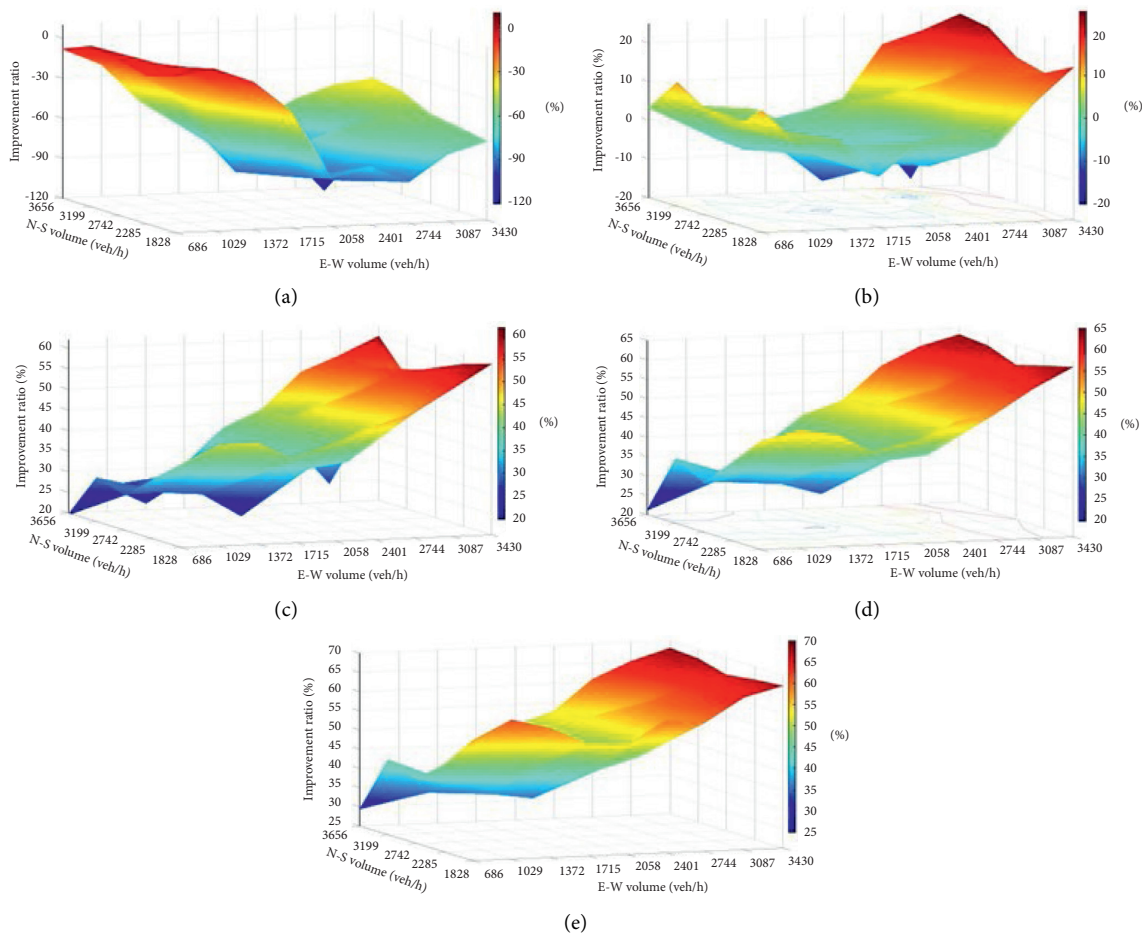


FIGURE 14: Improvement ratios of the left-turning time at the east entrance. (a) Scheme 1. (b) Scheme 2. (c) Scheme 3. (d) Scheme 4. (e) Scheme 5. E-W volume means the same traffic volume of east entrance and west entrance. N-S volume means the same traffic volume of north entrance and south entrance.

TABLE 8: Weights of the four indexes.

Index	C	D	S	N	Sum
Weight	0.1595	0.2905	0.2159	0.3341	1

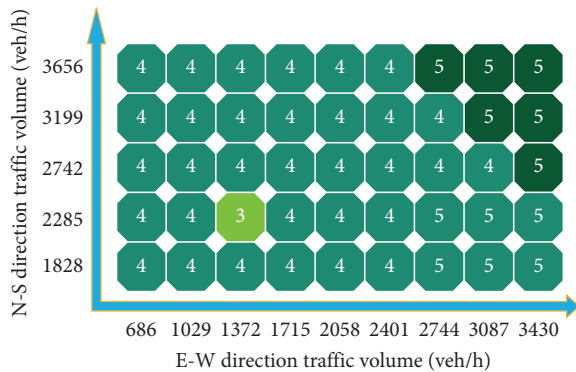


FIGURE 15: Recommended schemes' matrix diagram. The figure shows the recommended schemes in the different traffic combinations.

either Scheme 4 or Scheme 3 performed well when the traffic volume in both the east-west and north-south directions was small. However, when the traffic volume in both the east-west and north-south directions was getting larger, Scheme 5 showed its advantages in improving pollutant emission and became the recommended scheme. In general, Scheme 4 was the recommended solution for most cases because of its superior overall performance. In the final results of the recommended scheme, Scheme 3/Scheme 4/Scheme 5 = 1: 38:6. The recommended length obtained using the evaluation and analysis model met the recommended range of 300–500 feet in the guidebook on DLT intersections published by Federal Highway Administration [48], indicating that the results obtained from the model are scientific and reasonable.

5. Conclusions

The expanding urbanization exacerbates the problem of urban road congestion in China. Upgrading urban road intersections is one of the keys to solving this problem. The DLT intersection has been widely recognized as an unconventional intersection, which can improve traffic efficiency when the left-turn traffic flow is heavy. However, the current design of this unconventional intersection mainly relies on empirical methods and mathematical methods, and there are many deficiencies needed to be further overcome.

This paper investigated a model for determining the recommended length of added DLT lanes by using the VISSIM traffic simulation model and SSAM security evaluation model. Both VISSIM and SSAM were calibrated to ensure reliability, which could provide meaningful support for designing more efficient and safer DLT intersections. Considering the multiobjective decision on intersection design, the EEM was utilized in the model not only to solve the problem of multiparameter processing but also to ensure objectivity and avoid the controversy of artificial interference. The research carried out in this paper took a traditional cross intersection located in Xi'an, Shaanxi Province, China, as a case study. Only the west entrance had obvious transition and widening sections. The specific situation was shown in Figure 3. There was a large amount of left-turning

traffic flows in the east-west direction of the present intersection. Then, we designed development schemes with different lengths of added DLT lanes to deal with this problem. They performed better in efficiency and safety. The results showed that the development intersections with added DLT lanes significantly reduced the number of conflicts compared with the existing intersection, especially in the "rear end" and "lane change." Another important finding was that the change in the number of conflicts was not obvious as the length of the DLT lanes changed. Afterward, we conducted a sensitivity analysis of the improved schemes and the existing scheme under 45 different traffic combinations.

Using the model, the length of the 120 m long added DLT lanes at the crossing intersection which was recommended under most of the traffic volume combinations in this case. This was probably because setting 120 m added DLT lanes provided the superb balance of increasing capacity, reducing delays, and environmental protection.

And, the recommended length should not be less than 95 m under all of the 45 traffic conditions. This was probably because the additional DLT lanes were too short to meet the demand of vehicles that need to turn left, and the new schemes added subintersection, which might have negative changes compared to the traditional intersections.

The research proves that the added DLT lane length evaluation and analysis model proposed in this paper can eliminate the controversy of subjective human factors and achieve multiobjective optimization projects. It also proves that the evaluation and analysis model is advanced, reasonable, and maneuverable, and the considering multifactor design process for determining the recommended length of added DLT lanes with applying the analysis model also has good practicability. It can provide meaningful guidance for the designers in the design of the reconstructed DLT intersections.

Some issues in this paper that need to be further enriched and improved.

- (i) Budget constraints can be added to the model to select the recommended scheme
- (ii) In the future, the model also needs to consider how to compare schemes of different lane lengths when the number of DLT lanes' changes

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 there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This study was supported by Scientific Research Program funded by Shaanxi Provincial Education Department (Program no. 21JK0908).

References

- [1] Baidu Map, *2020 China Urban Transport Report*, Baidu Map Publications, Shenzhen, China, 2021.
- [2] J. Lioris, R. Pedarsani, F. Y. Tascikaraoglu, and P. Varaiya, "Platoons of connected vehicles can double throughput in urban roads," *Transportation Research Part C: Emerging Technologies*, vol. 77, pp. 292–305, 2017.
- [3] Y. Xiang, Z. Li, W. Wang, J. Chen, H. Wang, and Y. Li, "Evaluating the operational features of an unconventional dual-bay U-turn design for intersections," *PLoS One*, vol. 11, no. 7, Article ID e0158914, 2016.
- [4] Y. Shao, X. Han, H. Wu, H. Shan, S. Yang, and C. G. Claudel, "Evaluating the sustainable traffic flow operational features of an exclusive spur dike U-turn lane design," *PLoS One*, vol. 14, no. 4, Article ID e0214759, 2019.
- [5] H. Naghawi and W. Idewu, "Analysing delay and queue length using microscopic simulation for the unconventional intersection design Superstreet," *Journal of the South African Institution of Civil Engineers*, vol. 56, pp. 100–107, 2005.
- [6] J. G. Bared, P. K. Edara, and R. Jagannathan, "Design and operational performance of double crossover intersection and diverging diamond interchange," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1912, no. 1, pp. 31–38, 2005.
- [7] A. Dhattrak, P. Edara, and J. G. Bared, "Performance analysis of parallel flow intersection and displaced left-turn intersection designs," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2171, no. 1, pp. 33–43, 2010.
- [8] J. Bared, W. Hughes, R. Jagannathan, and J. E. Hummer, *Alternative Intersections/interchanges: Informational Report (AIIR) (No. FHWA-HRT-09-060)*, Federal Highway Administration, Office of Research, McLean, VA, USA, 2010.
- [9] J. Wu, A. Abbas-Turki, A. Correia, and A. E. Moudni, "Discrete intersection signal control," in *Proceedings of the IEEE International Conference on Service Operations and Logistics, and Informatics*, Philadelphia, PA, USA, August 2007.
- [10] L. Zhang, Q. Zhao, L. Wang, and L. Zhang, "Research on urban traffic signal control systems based on cyber physical systems," *Journal of Advanced Transportation*, vol. 2020, Article ID 8894812, 16 pages, 2020.
- [11] C. Yu, Y. Feng, H. X. Liu, W. Ma, and X. Yang, "Integrated optimization of traffic signals and vehicle trajectories at isolated urban intersections," *Transportation Research Part B: Methodological*, vol. 112, pp. 89–112, 2018.
- [12] C. Yu, W. Ma, K. Han, and X. Yang, "Optimization of vehicle and pedestrian signals at isolated intersections," *Transportation Research Part B: Methodological*, vol. 98, pp. 135–153, 2017.
- [13] Y. Li and Q. Liu, "Intersection management for autonomous vehicles with vehicle-to-infrastructure communication," *PLoS One*, vol. 15, no. 7, Article ID e0235644, 2020.
- [14] J. Olsson and M. W. Levin, "Integration of microsimulation and optimized autonomous intersection management," *Journal of Transportation Engineering Part A-System*, vol. 146, no. 9, 2020.
- [15] S. Ilgin Guler, M. Menendez, and L. Meier, "Using connected vehicle technology to improve the efficiency of intersections," *Transportation Research Part C: Emerging Technologies*, vol. 46, pp. 121–131, 2014.
- [16] J. Appiah, F. A. King, M. D. Fontaine, and B. H. Cottrell, "Left turn crash risk analysis: development of a microsimulation modeling approach," *Accident Analysis & Prevention*, vol. 144, Article ID 105591, 2020.
- [17] H. Chen, N. Zhang, and Z. Qian, "VISSIM-based simulation of the left-turn waiting zone at signalized intersection," *International Conference on Intelligent Computation Technology and Automation*, vol. 1, pp. 736–740, 2008.
- [18] Z. Yang, P. Liu, Z. Z. Tian, and W. Wang, "Evaluating the operational impact of left-turn waiting areas at signalized intersections in China," *Transportation Research Record*, vol. 2286, no. 1, pp. 12–20, 2008.
- [19] Q. Bai, Z. Gao, Z. Qu, and C. Tao, "Modeling for left-lane line extensions at signalized intersections with permitted left-turning phase," *Journal of Transportation Engineering Part A-Systems*, vol. 146, no. 8, 2020.
- [20] Y. Zheng, X. Hua, W. Wang, J. Xiao, and D. Li, "Analysis of a signalized intersection with dynamic use of the left-turn lane for opposite through traffic," *Sustainability*, vol. 12, no. 18, 2020.
- [21] M. M. A. Al-Omari, M. Abdel-Aty, J. Lee, L. Yue, and A. Abdelrahman, "Safety evaluation of median U-turn crossover-based intersections," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2674, no. 7, pp. 206–218, 2020.
- [22] Y. Xuan, C. F. Daganzo, and M. J. Cassidy, "Increasing the capacity of signalized intersections with separate left turn phases," *Transportation Research Part B: Methodological*, vol. 45, no. 5, pp. 769–781, 2011.
- [23] R. Jagannathan and J. G. Bared, "Design and operational performance of crossover displaced left-turn intersections," *Geometric design and the effects on traffic operations*, vol. 1881, no. 1, pp. 1–10, 2004.
- [24] W. Qu, Q. Sun, Q. Zhao, T. Tao, and Y. Qi, "Statistical analysis of safety performance of displaced left-turn intersections: case studies in San Marcos, Texas," *International Journal of Environmental Research and Public Health*, vol. 17, no. 18, 2020.
- [25] A. Abdelrahman, M. Abdel-Aty, J. Lee, L. Yue, and M. M. A. Al-Omari, "Evaluation of displaced left-turn intersections," *Transport Engineer*, vol. 1, no. 100006, 2020.
- [26] W. Suh and M. P. Hunter, "Signal design for displaced left-turn intersection using Monte Carlo method," *KSCCE Journal of Civil Engineering*, vol. 18, no. 4, pp. 1140–1149, 2014.
- [27] W. Qu, S. Liu, Q. Zhao, and Y. Qi, "Development of a progression-based signal-timing strategy for continuous-flow intersections," *Journal of Transportation Engineering Part A-Systems*, vol. 147, no. 3, pp. 1–11, 2021.
- [28] X. Hua, J. Yang, W. Wang, and H. Wang, "Operation optimization of signalized tandem intersections with displaced left turn," in *Proceedings of the Transportation Research Board 97th Annual Meeting*, Washington, DC, USA, January 2018.
- [29] J. Zhao, W. Ma, K. L. Head, and X. Yang, "Optimal operation of displaced left-turn intersections: a lane-based approach," *Transportation Research Part C: Emerging Technologies*, vol. 61, pp. 29–48, 2015.
- [30] J. Zhao, X. Gao, and V. L. Knoop, "An innovative design for left turn bicycles at continuous flow intersections," *Transportation Business: Transport Dynamics*, vol. 7, no. 1, pp. 1305–1322, 2019.
- [31] W. Sun, X. Wu, Y. Wang, and G. Yu, "A continuous-flow-intersection-lite design and traffic control for oversaturated bottleneck intersections," *Transportation Research Part C: Emerging Technologies*, vol. 56, pp. 18–33, 2015.
- [32] J. D. Reid and J. E. Hummer, "Travel time comparisons between seven unconventional arterial intersection designs," in *Geometric Design and the Effects on Traffic Operation 2001:*

- Highway Operation, Capacity, and Traffic Control*, pp. 56–66, no. 1751, Transportation Research Record, Washington, DC, USA, 2001.
- [33] S. Shokry, S. Tanaka, F. Nakamura, R. Ariyoshi, and S. Miura, “Bandwidth maximization approach for displaced left-turn crossovers coordination under heterogeneous traffic conditions,” *Journal of Traffic and Transportation Engineering*, vol. 6, pp. 183–196, 2018.
- [34] I. Ahmed, S. Warchol, C. Cunningham, and N. Roupail, “Mobility assessment of pedestrian and bicycle treatments at complex continuous flow intersections,” *Journal of Transportation Engineering Part A-Systems*, vol. 147, no. 5, pp. 1–10, 2021.
- [35] E. Hassannayebi, A. Sajedinejad, and S. Mardani, “Urban rail transit planning using a two-stage simulation-based optimization approach,” *Simulation Modelling Practice and Theory*, vol. 49, pp. 151–166, 2014.
- [36] E. Hassannayebi, M. Boroun, S. A. Jordehi, and H. Kor, “Train schedule optimization in a high-speed railway system using a hybrid simulation and meta-model approach,” *Computers & Industrial Engineering*, vol. 138, 2019.
- [37] H. Li, Z. Huang, X. Zou, S. Zheng, and Y. Yang, “VISSIM-based simulation and analysis of upstream segments in ramp areas for optimizing vehicle group lane-changing behaviors,” *Journal of Advanced Transportation*, vol. 2020, Article ID 5983161, 11 pages, 2020.
- [38] M. M. Morando, T. Q. Tian, L. T. Truong, and H. Vu, “Studying the safety impact of autonomous vehicles using simulation-based surrogate safety measures,” *Journal of Advanced Transportation*, vol. 2018, Article ID 6135183, 11 pages, 2018.
- [39] G. Tesoriere, T. Campisi, A. Canale, and T. Zgrablic, “The surrogate safety appraisal of the unconventional elliptical and turbo roundabouts,” *Journal of Advanced Transportation*, vol. 2018, Article ID 2952074, 9 pages, 2018.
- [40] G. Zhao and D. Wang, “Comprehensive evaluation of AC/DC hybrid microgrid planning based on analytic hierarchy process and entropy weight method,” *Applied Sciences-base*, vol. 9, no. 18, 2019.
- [41] T. Xie, M. Wang, C. Su, and W. Chen, “Evaluation of the natural attenuation capacity of urban residential soils with ecosystem-service performance index (EPX) and entropy-weight methods,” *Environmental Pollution*, vol. 238, pp. 222–229, 2018.
- [42] Y. Cui, P. Feng, J. Jin, and L. Liu, “Water resources carrying capacity evaluation and diagnosis based on set pair analysis and improved the entropy weight method,” *Entropy*, vol. 20, no. 5, pp. 359–379, 2018.
- [43] D. Koltovska, K. Bombol, and D. Ilievski, “Calibration and validation procedure of microscopic traffic simulation model: a case study,” in *Proceedings of the Second International Conference on Traffic and Transport Engineering (ICTTE)*, pp. 80–87, Belgrado, Serbia, April 2014.
- [44] Z. Li, M. V. Chitturi, D. Zheng, A. R. Bill, and D. A. Noyce, “Modeling reservation-based autonomous intersection control in VISSIM,” *Computers & Industrial Engineering*, vol. 2381, no. 1, pp. 81–90, 2013.
- [45] Federal Highway Administration Research and Technology, *Surrogate Safety Assessment Model (SSAM)*, Federal Highway Administration, Washington, DC, USA, 2020.
- [46] Federal Highway Administration (FHWA), *Techbrief Surrogate Safety Assessment Model (SSAM)*, Federal Highway Administration, Washington, DC, USA, 2008.
- [47] Transportation Researcher Board (TRB), *Highway Capacity Manual*, Transportation Researcher Board, Washington DC, USA, 6th edition, 2016.
- [48] Federal Highway Administration (FHWA), *Displaced Left-Turn Intersection Informational Guide*, Federal Highway Administration, Washington DC, USA, 2014.

Research Article

Short-Term Traffic Flow Prediction: A Method of Combined Deep Learnings

Chuanxiang Ren ¹, Chunxu Chai,¹ Changchang Yin ², Haowei Ji,¹ Xuezheng Cheng ², Ge Gao,¹ and Heng Zhang¹

¹College of Transportation, Shandong University of Science and Technology, Qingdao 266590, China

²College of Electrical Engineering and Automation, Shandong University of Science and Technology, Qingdao 266590, China

Correspondence should be addressed to Changchang Yin; ycc2009@sdust.edu.cn

Received 25 March 2021; Revised 16 June 2021; Accepted 21 June 2021; Published 5 July 2021

Academic Editor: Erfan Hassannayebi

Copyright © 2021 Chuanxiang Ren 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.

Short-term traffic flow prediction can provide a basis for traffic management and support for travelers to make decisions. Accurate short-term traffic flow prediction also provides necessary conditions for the sustainable development of the traffic environment. Although the application of deep learning methods for traffic flow prediction has achieved good accuracy, the problem of combining multiple deep learning methods to improve the prediction accuracy of a single method still has a margin for in-depth research. In this article, a combined deep learning prediction (CDLP) model including two paralleled single deep learning models, CNN-LSTM-attention model and CNN-GRU-attention model, is established. In the model, a one-dimensional convolutional neural network (1DCNN) is used to extract traffic flow local trend features and RNN variants (LSTM and GRU) with attention mechanism are used to extract long temporal dependencies trend features. Moreover, a dynamic optimal weighted coefficient algorithm (DOWCA) is proposed to calculate the dynamic weights of CNN-LSTM-attention and CNN-GRU-attention with the goal of minimizing the sum of squared errors of the CDLP model. Then, the neuron number, loss function, optimization algorithm, and other parameters of the CDLP model are discussed and set through experiments. Finally, the training set and test set for the CDLP model are established through the processing of traffic flow data collected from the field. The CDLP model is trained and tested, and the prediction results of traffic flow are obtained and analyzed. It indicates that the CDLP model can fit the change trend of traffic flow very well and has better performance. Furthermore, under the same dataset, the results from the CDLP model are compared with baseline models. It is found that the CDLP model has higher prediction accuracy than baseline models.

1. Introduction

With the economic development, the number of motor vehicles in the urban area has increased rapidly, and traffic congestion and traffic accidents have become increasingly serious. In order to mitigate the urban traffic problem, intelligent transportation systems have been widely implemented [1–4]. Among them, short-term traffic flow prediction is one of the core parts of an intelligent transportation system, which provides the basis for traffic management, traffic control, and traffic guidance and also provides support for traveler's decision-making. Prediction of short-term traffic flow has always been a hot topic for scholars in the field of traffic engineering.

For short-term traffic flow forecasting, early research mainly focused on statistical learning methods based on traditional mathematical models. Under the assumption of a certain probability distribution, the parameters of the statistical forecasting model are estimated through theoretical inference, and the model's forecasting results have a better strong explanatory. The traditional methods mainly include Kalman filter models, time series models, and nonparametric regression models.

Okutani and Stephanedes [5] proposed two prediction models based on the Kalman filter theory to predict the traffic flow of streets in Nagoya. In the models, the newest prediction error and the traffic data of multiple adjacent road

sections are considered to improve the prediction accuracy. Xie et al. [6] used the discrete wavelet decomposition method to denoise the traffic flow data and then established a Kalman filter model to predict the traffic flow, which reduced the interference of local noise on the original data and obtained better prediction results. Guo et al. [7] proposed an adaptive Kalman filter model, which uses the adaptive update method of variance to improve the parameters of the model, and verified that the prediction accuracy of the model is better than the traditional Kalman filter model through a large amount of highway traffic data. Emami et al. [8] proposed a fade memory Kalman filter model based on real-time data from the Internet of vehicles and Bluetooth detectors. This model considers the influence of weights and reduces the errors caused by the measurement method. Experiments show that the model can improve the accuracy of the forecast data.

The autoregressive integrated moving average (ARIMA) model is widely used in traffic flow prediction. Ahmed and Cook [9] investigated the ARIMA model in representing freeway time series data and found ARIMA was more accurate than moving average, double-exponential smoothing models. Hamed et al. [10] applied the ARIMA model to forecast traffic volume in urban arterials, and it turned out to be the most adequate model in reproducing all original time series and is computationally tractable. In addition to the ARIMA model, the autoregressive integrated moving average model with explanatory variables, seasonal autoregressive moving average model and other variant structures ARIMA models have also been applied in the field of traffic flow forecasting [11, 12].

The K-nearest neighbors (KNN) method does not require complex prior knowledge and precise function expressions. It has the advantages of a simple algorithm and good portability and has been applied in the field of traffic flow prediction. Zhang et al. [13] used the mean KNN and weighted KNN to establish traffic flow prediction models and comparative analysis was made. Cheng et al. [14] proposed an adaptive spatiotemporal KNN model, which comprehensively considers spatiotemporal weights, time windows, and other parameters, and simulation results demonstrated that the prediction effect of traffic flow has been further improved. The core content of the KNN is to design an appropriate search mechanism, and its prediction results rely on historical data. When the historical data are large, the search efficiency of this method will have a greater impact on the real-time performance of the prediction model.

The basic idea of the support vector machine (SVM) method for traffic flow prediction is to map the original traffic flow data to the high-dimensional feature space through the kernel function and to find the linearly divided plane from the mapping space to solve nonlinear problems in traffic flow data. Yang et al. [15] proposed a short-term traffic flow prediction model based on spatiotemporal correlation and adaptive multicore SVM for the nonlinearity and randomness of traffic flow. Luo et al. [16] used the method of least square SVM to predict the traffic flow, in which a hybrid optimization algorithm is proposed to select

the optimal parameters, and the experimental results show the model can improve the prediction ability and computational efficiency. Tang et al. [17] proposed a traffic flow prediction model that combines denoising schemes and SVM algorithms to improve the prediction accuracy. Results show the model outperforms that without denoising strategy. In addition to the traditional SVM model, variant SVM algorithms, such as seasonal SVM [18], which considers traffic data seasonality, and Online-SVR [19], which deals with special events, have also been applied in traffic flow prediction and good results are obtained.

The development and wide applications of traffic information collection technology, such as inductive detector, geomagnetic detectors, radio frequency identification technology, radar detection, video detection, and floating car detection [20–24], provide a large amount of data for traffic flow prediction. At the same time, with the rapid development of artificial intelligence technology, deep learning, which has powerful data feature mining and nonlinear data fitting capabilities, has been successfully applied in many fields, such as image processing and speech recognition [25–27], and gradually used in traffic parameter forecasting [28–31].

Moreover, the key point of traffic flow forecasting research has also shifted from traditional statistical learning forecasting methods and shallow neural networks [32–34] to deep learning forecasting methods. The shallow neural networks, which only have a single hidden layer, cannot learn the deeper features of traffic flow data and their prediction accuracy is often lower than that of the deep learning network. The deep learning methods have been gradually applied to the field of traffic flow prediction.

Deep belief network (DBN) is an earlier deep learning method used for traffic flow prediction. Huang et al. [35] designed a combined prediction model with unsupervised learning DBN at the bottom layer and multitask learning layer at the top layer for supervised prediction. The multitask learning layer can make full use of the weight sharing in DBN and outperform predicted results. Koesdwiady et al. [36] incorporated weather conditions and traffic flow data into the feature space at the same time and designed a DBN network for unsupervised pretraining, and relevant data from San Francisco are used to conduct experiments to verify the effectiveness of the proposed method. Xu and Jiang [37] proposed a DBN-support vector regression model for short-term traffic flow, in which DBN is used to learn the internal characteristics of traffic flow and support vector regression to predict the traffic flow. Experiments show that the model can effectively predict traffic flow and has fine prediction accuracy. Han and Huang [38] proposed a traffic flow prediction model combining DBN and a kernel extreme learning classifier, in which the internal characteristics of traffic flow data are extracted by DBN and the kernel extreme learner is used to predict traffic flow. Experiments show that the model can improve the accuracy of traffic flow prediction and reduce simulation time.

Convolutional neural network (CNN) is also a typical structure of deep learning. It is a feedforward neural network used to solve data problems similar to a grid structure. It can

accurately extract data features while reducing the complexity of the model. This efficient local feature extraction capability is conducive to better find the spatial correlation between traffic flow data, and then it is widely used in traffic flow prediction [39]. Zhang et al. [40] proposed a short-term traffic flow prediction model based on CNN, in which a spatiotemporal feature selection algorithm determines the optimal input data time lags and amounts of traffic flow data; then, CNN learns these spatiotemporal features. The effectiveness of the model was verified by comparing the prediction results with actual traffic data. An et al. [41] proposed a fuzzy-based CNN traffic flow prediction model, in which the fuzzy approach is used to represent the features of traffic accidents. The experimental results show that the model has superior performance. Liu et al. [42] proposed a CNN-attention model to predict traffic speed. Experimental results show that the model has a great advantage in traffic flow prediction and the impact of different traffic flow temporal and spatial data on traffic flow can be found through visualizing the weights generated by the attention model. Peng et al. [43] proposed a spatial-temporal incidence dynamic graph recurrent CNN to predict urban traffic passenger flow and experiments show that the predictive performance of this network is superior to traditional predictive methods.

LSTM network is a deep learning structure and also a variant of recurrent neural network (RNN). RNN can be applied to the relevant forecasting field of time series data [44]. However, RNN has a problem of the disappearance of the gradient, which can be overcome by LSTM [45]. LSTM has been applied in the field of traffic flow prediction. Ma et al. [46] applied the LSTM to establish a traffic speed prediction model. The results show that the LSTM network effectively captures the time correlation and nonlinearity of the traffic state, and the prediction accuracy is better than most statistics methods. Zhao et al. [47] proposed a traffic forecast model based on LSTM considering temporal-spatial correlation in traffic systems. The results validate that the model can obtain better prediction performance compared with other representative forecast models. Tian et al. [48] proposed a multiscale smoothing method to fill in the missing values in traffic flow data and established an LSTM model to predict traffic flow. Experiments show that the LSTM model has better prediction performance than other prediction methods. Zhao et al. [49] established the LSTM model to predict traffic flow speed and validated that the prediction accuracy is higher than that of the support vector regression prediction method. Wang et al. [50] constructed an LSTM encoding and decoding model based on the attention mechanism for time series prediction, which includes periodic mode and recent time mode. Experiments show that the model is effective and reliable in long-term prediction of time series.

In addition, combination algorithms for traffic flow prediction, especially deep learning algorithms, have received more attention from scholars and produced a series of achievements. Zhou et al. [51] combined LSTM and SVR to build a model for short-term traffic flow prediction, in which a genetic algorithm is used to optimize the parameters of

SVR. The results indicate that the prediction model has higher accuracy than LSTM and CNN. Zhang et al. [52] proposed a model for short-term traffic forecasting, which integrates a graph convolution operator and a residual LSTM structure. The model is evaluated on a traffic speed dataset and better prediction results than six baselines are obtained. Li et al. [53] developed a deep learning-based method, including CNN and LSTM, for real-time movement-based traffic volume prediction at signalized intersections. In the model, CNN is applied to learn the spatial features of traffic volume and LSTM to learn the temporal dependencies. Xia et al. [54] proposed a distributed LSTM weighted model combined with a time window and normal distribution to enhance the prediction capability for traffic flow. Furthermore, the experimental results indicate that the model achieved accuracy improvement.

In summary, the deep learning methods have been widely applied to short-term traffic flow prediction and achieved series of results. Moreover, from the above literature researches, it can be found that the combination of multiple deep learning methods, such as a combination of CNN and LSTM, can improve the performance of the prediction model. LSTM is a variation of RNN, which can obtain the time series characteristics of traffic flow. Meanwhile, there is another variant of RNN, namely GRU, which can also obtain the time series characteristics of traffic flow and make traffic flow prediction [55, 56]. The combined model of LSTM and GRU is used to predict traffic flow parameters, which has been discussed and applied in [57, 58], and its outstanding performance in both prediction accuracy and stability has been proved. In the two works of literature, LSTM and GRU are serial structures. LSTM is firstly used to learn the spatial-temporal characteristics of data, and then GRU is used to predict traffic parameters or LSTM is firstly used to predict value and then encoder with GRUs further captures the relationship between the input sequence and the output sequence. However, the sequential combination structure of LSTM and GRU does not simultaneously use the advantages of the two to complement each other, and it also lacks CNN's guidance on the local trend of traffic flow. It is necessary to apply the combination of three deep learning methods to study the prediction of traffic flow. In addition, the attention mechanism theory [59] has the function of improving the data extraction capabilities of deep learning by imitating human vision to assign weights to data features and has been widely used in image processing and speech recognition [60–63]. Applying it to CNN, LSTM, and GRU deep learnings for traffic flow prediction is also worthy of discussion.

In this article, a DOWCA is presented, and a combined prediction model with CNN, LSTM, GRU, and attention mechanism for short-term traffic flow is proposed and discussed. The main contributions of this study are as follows:

- (1) In order to build a combined traffic flow prediction model, a dynamic optimal weighted coefficient algorithm (DOWCA), is proposed, in which the weights of each single prediction method are

calculated dynamically following new prediction results added.

- (2) A combined deep learning model for short-term traffic flow prediction, namely CDLP, is established based on the CNN, LSTM, GRU, and attention mechanism, which includes paralleled CNN-LSTM-attention model and CNN-GRU-attention model. In CDLP, the dynamic weights for the two single models are calculated by DOWCA.
- (3) After parameter setting through experiment comparison and analysis, the CDLP model is trained and tested using traffic flow data from the field. The results indicate that the CDLP model outperforms baseline models.

The rest of the article is organized as follows. In Section 2, the methodologies of CNN, LSTM, GRU, and attention mechanism are introduced. In Section 3, a DOWCA is proposed and the CDLP model is constructed. In Section 4, the experiment results and analysis are presented. Finally, a brief conclusion and recommendations for future work are presented in Section 5.

2. Methodology

2.1. CNN. CNN is a feedforward neural network with a deep structure and mainly composed of convolution layer, pooling layer, and full connection layer [64]. Among them, the convolutional layer is the most important part of CNN, which uses the convolution kernel to carry out a convolutional calculation for data from the input layer and outputs the convolutional characteristics of the data. If the CNN model contains multiple convolutional layers, then the number of output characteristic parameters by the convolutional layer is large. In order to reduce the number of parameters, the pooling layer is often used to carry out subsampling operations on the convolutional features of the data to extract part of the information and prevent the model from overfitting. The fully connected layer is usually used at the end of the CNN model to reduce unnecessary feature loss, in which all features are integrated and calculated as the final output.

2.2. LSTM Network. LSTM is a variant structure of RNN, which can solve the problem of gradient disappearance and gradient explosion in RNN and can better realize the prediction of time series sequence. The LSTM network is composed of a series of basic cells. The basic cell structure is shown in Figure 1, which includes three gate structures: input gate, output gate, and forget gate.

The orange lines in Figure 1 represent the input gate. The main function of the input gate is to control the input process of all information at time t . The information input process mainly includes two parts. One part is the process of updating the current time information through the tanh function to obtain a new state vector, and the other part is superimposing the current input and the output information of the hidden layer at the

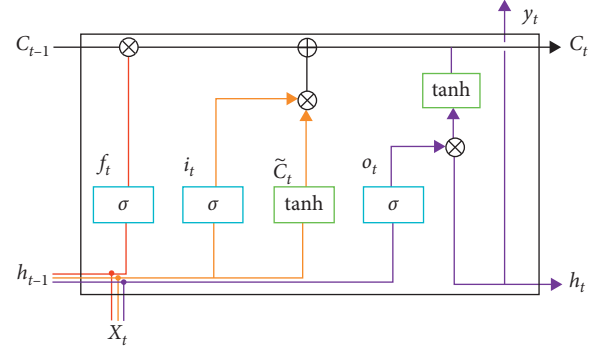


FIGURE 1: The basic unit structure of LSTM.

previous time through the sigmoid function. The specific implementation process can be expressed as follows:

$$\begin{aligned} i_t &= \sigma(W_i h_{t-1} + U_i x_t + b_i), \\ \tilde{C}_t &= \tanh(W_c h_{t-1} + U_c x_t + b_c), \end{aligned} \quad (1)$$

where W_i , W_c , U_i , and U_c are the weights of the input gates; b_i and b_c are the biases of the input gates; and σ and \tanh are activation function, and their formulas are as follows:

$$\begin{aligned} \tanh(x) &= \frac{e^x - e^{-x}}{e^x + e^{-x}}, \\ \sigma(x) &= \frac{1}{1 + e^{-x}}. \end{aligned} \quad (2)$$

The red lines in Figure 1 represent the forget gate, whose main function is to determine the redundant information to be discarded in the unit. The input of the forget gate includes input X_t and output h_{t-1} of the unit at the previous time. The output process is shown in formula (3).

$$f_t = \sigma(W_f h_{t-1} + U_f x_t + b_f), \quad (3)$$

where W_f and U_f are the weight of the forget gate and b_f represents the bias of the forget gate.

The forget gate uses the sigmoid function to superimpose the input values X_t and h_{t-1} , and the output value is limited to the range of $[0, 1]$; finally, the output value is multiplied by the output unit state C_{t-1} at the previous moment. When the output value is 0, it means that the information will be completely discarded. When the output value is 1, it means that the information will be completely retained.

The output information of the forget gate and the input gate is, respectively, multiplied and superimposed on each other to obtain the current unit output state. The specific calculation process is as follows:

$$C_t = f_t \otimes C_{t-1} + i_t \otimes \tilde{C}_t. \quad (4)$$

It can be seen from this formula that C_t represents the long-term memory of all historical information at the current moment.

The purple lines in Figure 1 represent the output gate. The output gate determines the output result of the entire basic cell, which is related to the cell output state C_t at the

current moment. First, use the sigmoid function to process part of the information of the input unit to obtain the output O_t of the output gate and then use the tanh function to process the information in C_t . After the two sets of processed information are multiplied, the final output h_t is obtained. The specific calculation formula is as follows:

$$\begin{aligned} O_t &= \sigma(W_o h_{t-1} + U_o x_t + b_o), \\ h_t &= O_t \otimes \tanh(C_t). \end{aligned} \quad (5)$$

2.3. GRU Network. Similar to LSTM, GRU is also a variant structure of the RNN algorithm, and it also has the function of dealing with the problem of gradient disappearance in RNN and ineffective long-term sequence memory. Compared with LSTM, GRU reduces the complexity of the structure by reducing the gates in the architecture. The cyclic structure of GRU consists of two gate structures, an update gate (purple lines) and a reset gate (red lines), and its cell structure is shown in Figure 2.

The update gate z_t can determine the memory information at the previous time and the remaining part of the information at the current time and continue to transfer the remaining information to the future time so as to obtain the long-term dependence in the entire network transmission process. The reset gate r_t is mainly used to obtain short-term time dependence, control the operation of the hidden state information h_{t-1} and the current input value x_t at the previous moment, and decide to forget the amount of information in the past.

Formulas (6)–(9) represent the calculation process of each state within each time step in GRU cell.

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z), \quad (6)$$

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r), \quad (7)$$

$$g_t = \tanh(W_g x_t + U_g (r_t \otimes h_{t-1}) + b_g), \quad (8)$$

$$h_t = z_t \otimes h_{t-1} + (1 - z_t) \otimes g_t, \quad (9)$$

where W_z , W_r , and W_g are input-related weight matrices; U_z , U_r , and U_g are cyclically connected weight matrices; and b_z , b_r , and b_g are related biases.

2.4. Attention Mechanism. Attention mechanism focuses on important information by assigning different weights to input features. The process of focusing on important information is shown as the calculation process of weight. The higher the importance of information is, the larger the weight is allocated. In the application of attention mechanism in deep learning model, the calculation process of context vector and weight involved is as follows.

The output hidden state of the deep learning model is supposed as $h_1, h_2, \dots, h_i, \dots, h_t$, and the context vector C_t can be calculated as follows:

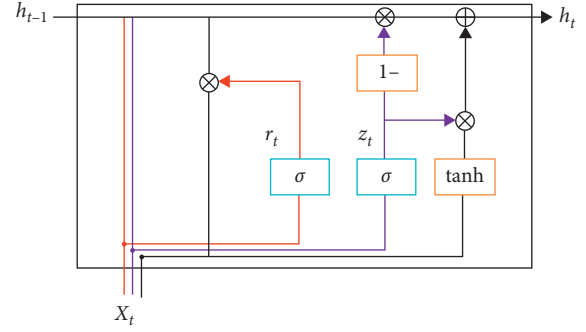


FIGURE 2: The basic unit structure of GRU.

$$C_t = \sum_{i=1}^t \alpha_{t,i} h_i. \quad (10)$$

In formula (10), $\alpha_{t,i}$ is the weight for h_i , and the sum of the weights is 1. It can be calculated as follows:

$$\alpha_{t,i} = \frac{\exp(e_{t,i})}{\sum_{i=1}^T \exp(e_{t,i})}, \quad (11)$$

where $e_{t,i}$ is an alignment model, and its calculation formula is as follows:

$$e_{t,i} = \tanh(W_a s_{t-1} + U_a h_i + b_a), \quad (12)$$

where W_a , U_a , and b_a are the network parameters of deep learning model and s_{t-1} can be calculated as follows:

$$s_{t-1} = g(s_{t-2}, y_{t-2}, c_{t-1}), \quad (13)$$

where $g(\cdot)$ denotes the deep learning network.

Based on formula (13), the output of the attention mechanism is expressed as follows:

$$y_t = \text{softmax}(s_t), \quad (14)$$

where softmax is activation function.

3. Model

3.1. Dynamic Optimal Weighted Coefficient Algorithm. Compared with a single prediction model, the combined prediction model can comprehensively utilize the advantages of multiple prediction models, improve the accuracy of prediction results, and has better robustness. In the combined prediction model, the calculation of the weighted coefficient of each single prediction model is the key. Generally, the optimal weighted coefficient algorithm (OWCA) is used, in which the weighted coefficient of each single prediction method is calculated with the goal of minimizing the sum of squared errors of the combined prediction [65–68]. The calculation principle is as follows.

Suppose there are m prediction methods; the prediction value of the i th method at time t is y_{it} , where $i = 1, 2, \dots, m$; $t = 1, 2, \dots, N$. Then, the prediction error e_{it} of the i th prediction method can be expressed by the following:

$$e_{it} = y_t - y_{it}. \quad (15)$$

Let l_1, l_2, \dots, l_m be the weighted coefficients of m prediction methods, respectively, and $l_1 + l_2 + \dots + l_m = 1$. The prediction result of the combined prediction method, labeled as \hat{y}_t , can be calculated as follows:

$$\hat{y}_t = l_1 y_{1t} + l_2 y_{2t} + \dots + l_m y_{mt}, \quad (16)$$

and the prediction error e_t for the combined prediction method at time t can be obtained:

$$e_t = \sum_{i=1}^m l_i e_{it}. \quad (17)$$

Let J represent the sum of squared errors of the combined prediction method, then the problem of solving the optimal weight at time t can be expressed as the following optimization model:

$$\text{Min } J = \sum_{t=1}^N \sum_{i=1}^m \sum_{j=1}^m l_i l_j e_{it} e_{jt}. \quad (18)$$

Formula (18) can be expressed in matrix form as follows:

$$\begin{aligned} \text{Min } J &= L^T E L \\ \text{s.t. } R^T L &= 1, \end{aligned} \quad (19)$$

where $L = (l_1, l_2, \dots, l_m)^T$ represents the weighted coefficient column vector; $R = (1, 1, \dots, 1)^T$ represents the m -dimensional column vector with all 1 elements; E is the combined prediction information error matrix, $E = (E_{ij})_{m \times m}$ and E_{ij} is expressed as follows:

$$E_{ij} = e_i^T e_j, \quad i, j = 1, 2, \dots, m, \quad (20)$$

where e_i represents the prediction error column vector of the i th single prediction method, and $e_i = (e_{i1}, e_{i2}, \dots, e_{iN})^T$.

If the prediction error vector group of m prediction methods is linearly independent, then the combined prediction information error matrix E is an invertible matrix. According to the Lagrange multiplier method [69], the optimal solution of model (18) can be obtained as follows:

$$L^* = \frac{E^{-1} R}{R^T E^{-1} R}, \quad (21)$$

where L^* is the optimal weight vector, namely, the optimal weighted coefficients of m prediction methods.

According to the OWCA and the historical prediction error of each single prediction method, the optimal weighted coefficient of each single prediction method can be obtained so as to carry out the combined prediction. In the OWCA, the weighted coefficient of each single prediction method is fixed. However, in the prediction of time data sequences, such as traffic flow, with the increase of time, the prediction results of each single prediction method also increase. More importantly, the prediction errors of each single prediction method also vary. If the weighted coefficient of each single prediction method is invariable, it cannot reflect the influence of the newly increased prediction results of each

single prediction method on the combined forecasting, which also affects the accuracy of the combined forecasting results.

Therefore, based on the optimal weighted coefficient algorithm, a dynamic optimal weighted coefficient algorithm, namely, DOWCA, is proposed. In the DOWCA, with the increase of time, the amount of historical prediction error data increases continuously, the weighted coefficient of each single prediction method, namely, the dynamic weighted coefficient, labeled as $l_{1t}, l_{2t}, \dots, l_{mt}$, is recalculated by the OWCA. The dynamic weighted coefficients are applied to each single prediction method and the combined prediction results are obtained. The whole process of the DOWCA is shown in Figure 3, and the pseudocode of DOWCA is shown in Algorithm 1.

3.2. Combined Deep Learning Prediction Model. CNN has the ability to obtain local trend features of data sequences, while LSTM and GRU have the ability to obtain long-term dependent features of data sequences. At the same time, the attention mechanism can make the deep learning model pay attention to important features. Based on this, a combined deep learning prediction model with CNN, LSTM, GRU, and DOWCA is designed for traffic flow prediction, namely, CDLP model.

In the CDLP model, CNN, LSTM, and attention are connected sequentially and become the sequential combination structure, which is named as CNN-LSTM-attention model, i.e., one single traffic flow prediction model in the CDLP model. Moreover, CNN, GRU, and attention are also designed as the sequential combination structure and named as CNN-GRU-attention model, i.e., another single traffic flow prediction model in the CDLP model. Then, the two sequential combination structures are paralleled and combined by DOWCA. From a layer standpoint, the CDLP model has three layers, input layer, hidden layer, and output layer. The hidden layer includes four layers, CNN layer, LSTM and GRU layer, attention layer, and dropout layer. The whole structure of the CDLP model is shown in Figure 4.

The input layer of the CDLP model is the processed traffic flow data sequence, including training set and test set, which is simultaneously inputted to two paralleled CNN layers in the hidden layer of the CDLP model.

The hidden layer of CDLP includes two CNN layers, LSTM and GRU layers, two attention layers, and two dropout layers in sequence. Moreover, all of them are paralleled. About the CNN layer, due to the periodicity and sequence of traffic flow data, 1DCNN is used and the output of 1DCNN is computed by the activation function ReLu. The formula of ReLu is as follows:

$$\text{ReLU}(x) = \begin{cases} x, & x > 0, \\ 0, & \text{otherwise.} \end{cases} \quad (22)$$

About LSTM and GRU layers, if too many network layers are selected, the calculation of the entire network will be large and more training time will be needed. According to [70], when both the accuracy of the prediction model and the training time are considered, the two LSTM network layers

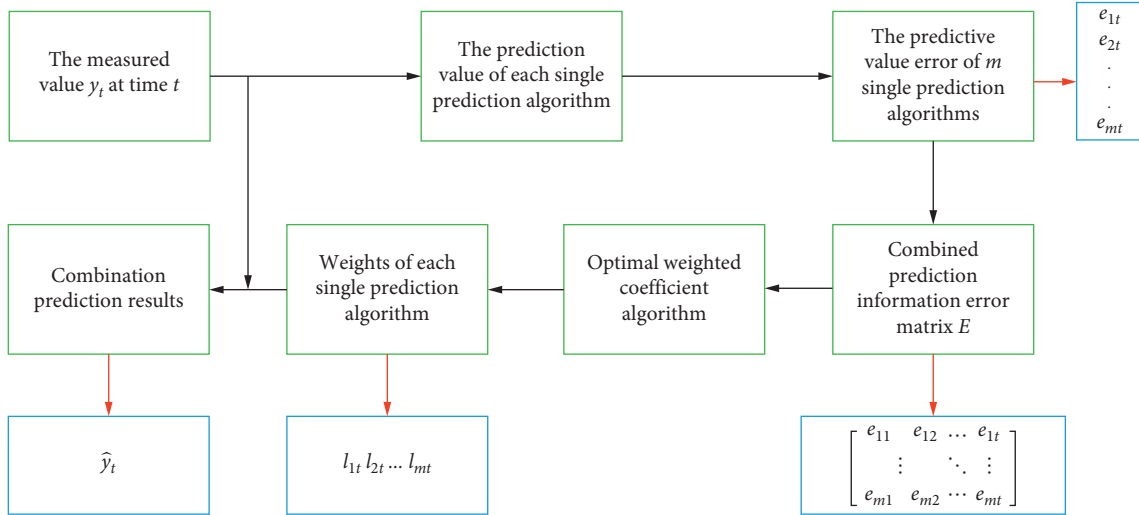
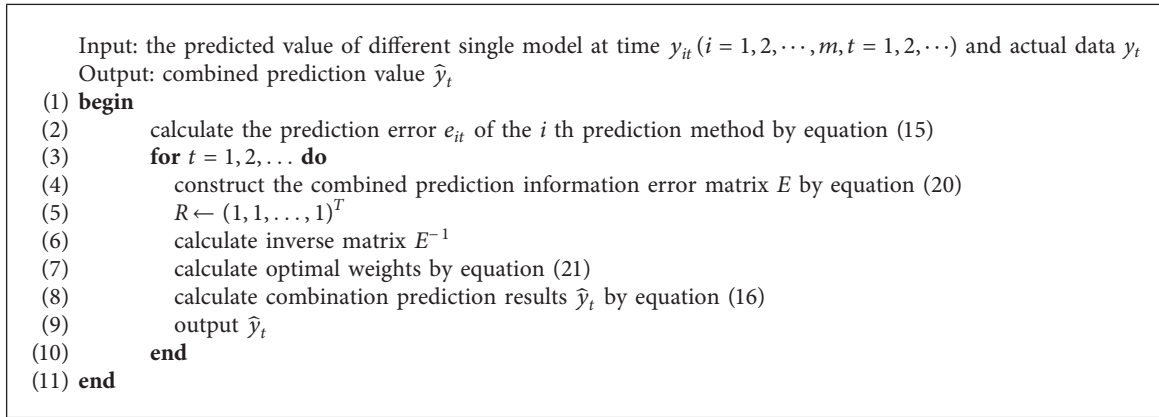


FIGURE 3: The process of DOWCA.



ALGORITHM 1: The pseudocode of DWOCA

are suitable, so two network layers in LSTM are selected. Similarly, two network layers are selected in GRU. The input of the first LSTM and GRU network layer is local trend features extracted by 1DCNN and its output is the state of the neural unit of the current LSTM and GRU layer. The second LSTM and GRU network layer mines the characteristics of the data and outputs the hidden layer state to the attention layer.

About the attention layer, the input state $h_1, h_2, \dots, h_i, \dots, h_t$ comes from LSTM and GRU. Correspondingly, $g(\cdot)$ in formula (13) denotes LSTM and GRU.

The last layer in the hidden layer, the dropout layer, is designed to prevent the occurrence of overfitting after the attention layer, which is the output from the hidden layer of CDLP to the output layer. Moreover, the input of the dropout layer is the output y_1, \dots, y_{t-1}, y_t from the attention layer.

The CDLP model is aimed to predict the traffic flow at the next moment based on the historical data. Therefore, the output layer includes two paralleled neural units, which are actually the outputs of two single models, CNN-LSTM-

attention model and CNN-GRU-attention model, respectively. The two output neural units are fully connected with the dropout layer. In addition, the output layer of the CDLP model also includes weight calculation for the outputs of the CNN-LSTM-attention model and CNN-GRU-attention model, in which the DOWCA is used. Finally, the traffic flow prediction values and the dynamic optimal weights for outputs of CNN-LSTM-attention and CNN-GRU-attention are obtained.

4. Experiment

4.1. Data Processing and Dataset. The traffic flow data at the intersection of Jiangxi Road and South Fuzhou Road in Qingdao, China, are collected through inductive detector as the dataset for the verification of the CDLP model. The original dataset contains three consecutive months of traffic flow data for each entrance road segment at the intersection from February 15, 2019, to May 15, 2019. The statistical time interval of these data is 5 minutes, and a total of 25920 pieces of data are obtained.

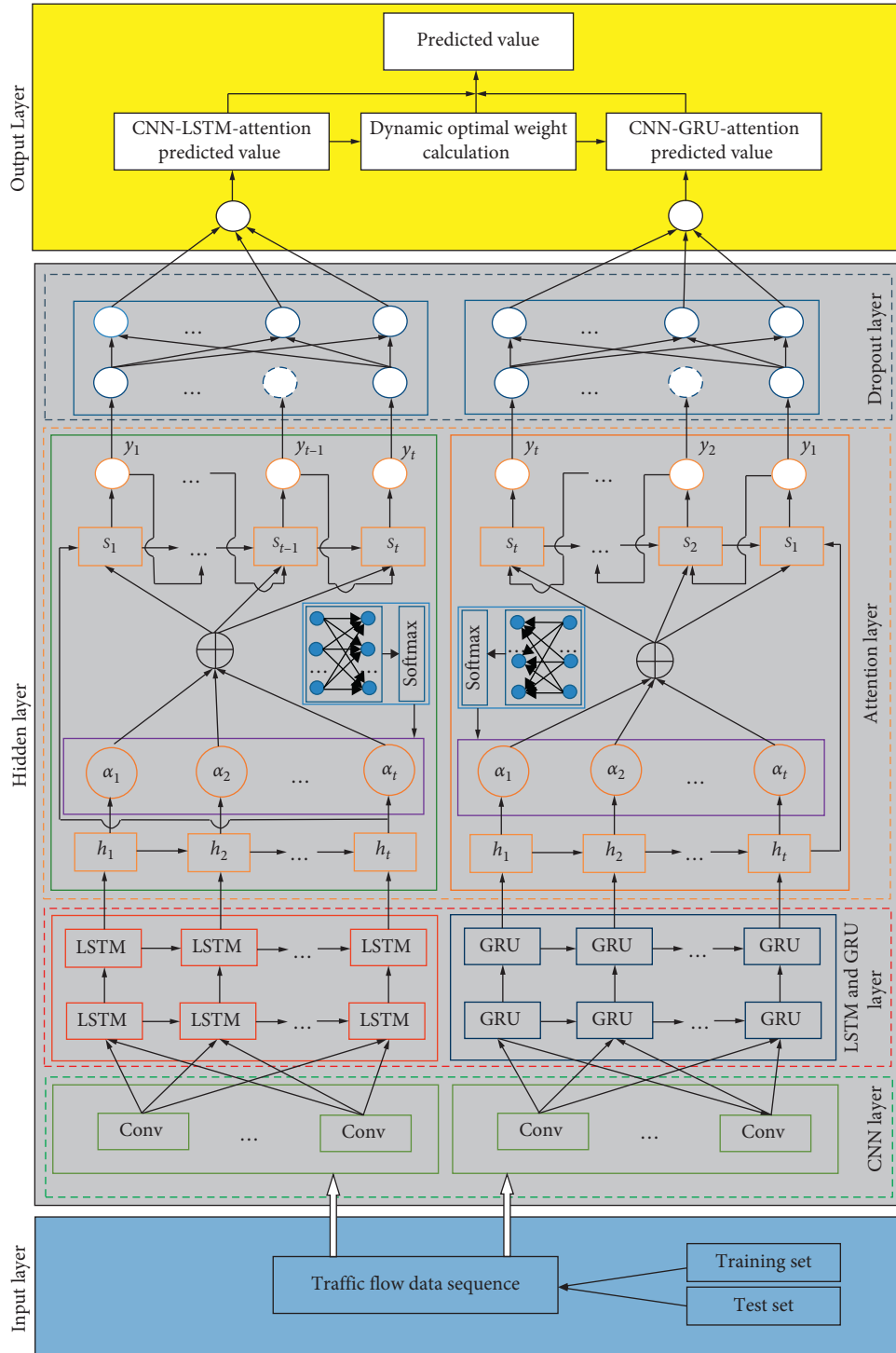


FIGURE 4: The structure of the CDLP model.

First, the abnormal and missing data in the original data are processed, in which the abnormal data are regarded as missing data. The Lagrangian interpolation method is used to process the missing data. In the process, four adjacent data before and after the missing datum are selected for interpolation to ensure the reliability of the interpolation data.

Then, the Min-Max method is used to normalize the data, and the calculation formula is as follows:

$$y' = \frac{y - y_{\min}}{y_{\max} - y_{\min}}, \quad (23)$$

where y_{\min} and y_{\max} are the minimum and maximum values of traffic flow, respectively and y and y' are the traffic flow data before and after being normalized, respectively.

The normalized data are divided into the training set and test set. The data from February 15, 2019, to May 1, 2019, are

used as the training set, and the dataset from May 2, 2019, to May 15, 2019, is the test set.

4.2. Experimental Environment and Selection of Evaluation Indicators. The hardware and software conditions in the experimental environment of this article are shown in Table 1.

In order to evaluate the traffic flow prediction performance of the CDLP model, three evaluation indicators are selected: MAPE, MAE, and RMSE. Their calculation formulas are as follows:

$$\begin{aligned} \text{MAPE} &= \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100, \\ \text{MAE} &= \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|, \\ \text{RMSE} &= \left(\frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{n} \right)^{1/2}, \end{aligned} \quad (24)$$

where n is the total number of samples in the test set, y_i is the i th actual value of sample, and \hat{y}_i is the predicted value of the i th sample.

4.3. CDLP Model Parameter Setting

4.3.1. Loss Function. The loss function quantifies how close a given neural network is to the ideal state it is trained on. The average absolute error function and the mean square error function are used as loss functions commonly. Because of the convenient calculation of the mean square error function, in the CDLP model, the mean square error function is selected as a loss function and the calculation formula is as follows:

$$L(y, \hat{y}) = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2, \quad (25)$$

where y_i is the actual value, n is the total number of samples, and \hat{y}_i is the prediction value.

4.3.2. The Neuron Number in the CDLP Model. The neuron numbers of the input layer and hidden layer should be set before the model is trained (the number of neurons in the output layer has been determined in Section 3.2). The following is the process of setting the number of neurons in the input layer and the hidden layer.

In order to obtain the appropriate neuron number of the input layer, 6, 12, 18, and 24 are selected, respectively, to train the model, and the optimal neuron number is obtained through error analysis of the test set. Similarly, for the setting of the neuron number of LSTM layers and GRU layers, four numbers of 16, 32, 64, and 128 are selected, respectively, to train the model. Moreover, the optimal neuron number is determined through the error analysis of the test set.

Regarding the error analysis of the test set, MAPE is selected as the main evaluation indicator, while MAE and

TABLE 1: Experimental environment.

Software and hardware configuration	Configuration parameter
CPU	Intel i5-8250U @1.60 GHz
RAM	8G
Programming language	Python 3.7.0
Deep learning framework	Tensorflow 1.14
Deep learning library	Keras 2.3.1

RMSE are used as auxiliary evaluation indicators. The evaluation indicator results of the test set under different neuron numbers in input and LSTM layers are obtained, which include the MAPE, MAE, and RMSE, as shown in Table 2.

From Table 2, it can be seen that when the neuron number of the input layer is set to 12 and the neuron numbers of the two LSTM layers are set to 128 and 128, respectively; the MAPE, MAE, and RMSE of the model test set are all the smallest. It indicates that the neuron numbers of the input layer and hidden layer are the best for the model training effect under this setting. Moreover, the neuron numbers of the two GRU layers are the same as those of LSTM, i.e., 128 and 128, respectively.

4.3.3. Optimization Algorithm. In the training process of the deep learning model, an optimization algorithm is used to iterate the model parameters to reduce the loss function value so that the training process of the model tends to be stable as the number of iterations increases. The optimization algorithms mainly include RMSprop and Adam. The two algorithms are applied to train the CDLP model and the better one is selected as an optimization algorithm according to the prediction results. After training of CDLP model under RMSProp algorithm and Adam algorithm, respectively, the results of three evaluation indicators are obtained and shown in Table 3.

It can be found from Table 3 that when the Adam algorithm is used to train the CDLP model, the MAPE, MAE, and RMSE are less than those of the RMSProp algorithm. It indicates that the Adam algorithm is more effective than the RMSProp algorithm and is selected as the optimization algorithm of the CDLP model.

4.3.4. Other Parameters. In the 1DCNN layer, the convolution operation is implemented by convolution kernels, and 64 convolution kernels with a size of 2×1 are used, i.e., filters = 64, size = 2. In the dropout layer, the loss rate of the dropout function is set as 20%. In addition, the epoch is set as 500 iterations, and the batch size is set as 128.

4.4. Results and Analysis. The CDLP model is trained and tested with a designed training set and test set after the above model parameters are determined. At the same time, in order to verify the advantages of the CDLP model, the prediction results from the single CNN-LSTM-attention model and single CNN-GRU-attention model are extracted during the process of training and testing for the CDLP

TABLE 2: Evaluation indicator results under different neuron numbers of input and LSTM layer.

Neuron number in the network	MAPE (%)	MAE	RMSE
(6,16,16,1)	25.52	24.92	35.77
(6,32,32,1)	12.99	19.85	29.54
(6,64,64,1)	10.60	12.52	19.40
(6,128,128,1)	12.07	9.68	14.03
(12,16,16,1)	12.86	20.68	31.48
(12,32,32,1)	24.06	19.93	28.90
(12,64,64,1)	9.24	8.65	13.00
(12,128,128,1)	7.33	5.12	8.25
(18,16,16,1)	15.59	20.84	31.93
(18,32,32,1)	10.78	13.60	20.48
(18,64,64,1)	14.47	9.29	13.53
(18,128,128,1)	12.86	7.10	9.92
(24,16,16,1)	13.35	20.88	32.02
(24,32,32,1)	12.59	12.98	19.65
(24,64,64,1)	22.54	11.37	15.66
(24,128,128,1)	21.84	7.81	11.42

TABLE 3: Comparison of three evaluation indicators between RMSProp and Adam.

Optimization algorithm	MAPE (%)	MAE	RMSE
Adam	7.31	5.21	9.06
RMSprop	8.21	5.60	9.73

model. Moreover, the corresponding results are obtained. Figure 5 shows the loss function curve of the training set and test set of CNN-LSTM-attention and CNN-GRU-attention. Figure 6 shows the prediction results of the CDLP model for the test set.

From Figure 5(a), it can be seen that the loss function of the training set of the CNN-LSTM-attention decreases rapidly and steadily as the number of iterations increases and finally tends to a stable state. Then, the loss function of the test set goes through initial fluctuations as the iteration progresses, quickly tends to the loss function of the training set, and is in a stable state. It can be seen from Figure 5(b) that similar to the CNN-LSTM-attention, the loss function of the training set of the CNN-GRU-attention network decreases rapidly and steadily and finally tends to a stable state and the loss function of the test set also gradually tends to the training set after initial fluctuations. Finally, the loss function is in a stable state. The loss function curves of the training set and test set of CNN-LSTM-attention and CNN-GRU-attention show that the design of CNN-LSTM-attention and CNN-GRU-attention network in the CDLP model is reasonable.

Figure 6 contains a comparison of the traffic flow predicted results of the CDLP model with the actual value (top figure) and the error of predicted traffic flow (below figure). From the top figure, it can be found that the CDLP model fits the change trend of traffic flow very well, indicating that the model learns the time change characteristics of the traffic flow series and realizes the better prediction. Moreover, from Figure 6, it can be seen that the overall errors remain stable and most of them change in a certain range of -20 and 20. Moreover, based on the error of predicted traffic flow, the MAPE curve is obtained

and shown in Figure 7. From the figure, it can be found that the trend of the MAPE curve first quickly rises to the maximum value, then quickly decreases, and gradually becomes stable. Finally, the MAPE curve tends to be 5.12%. This shows that the CDLP model has excellent robustness and obtains small error, further showing that the CDLP model can better realize the prediction of traffic flow.

Furthermore, in order to further verify the prediction effect of the CDLP model, Figure 8 shows the absolute error comparison of the traffic flow predicted values of the CDLP model, CNN-LSTM-attention model, and CNN-GRU-attention models. Figures 8(a) and 8(b) show the traffic flow prediction errors of the first week and the second week in the test set under three models. As can be seen from the figure, the fluctuation range of the prediction error curve of the CDLP model is the smallest, followed by CNN-LSTM-attention and CNN-GRU-attention. This indicates that the prediction accuracy of the CDLP model is better than that of CNN-LSTM-attention or CNN-GRU-attention and also shows the advantages of the combination model compared to a single model.

Meanwhile, some baseline models published in recent years, which are LSTM, GRU, CNN, CNN-LSTM, CNN-GRU, CNN-LSTM-attention, and CNN-GRU-attention, are used to verify the accuracy of the CDLP model. The evaluation indicators of the CDLP and baseline models, which are MAPE, MAE, and RMSE, are obtained, as shown in Table 4. Moreover, the training times of CDLP and baseline models are shown in Table 5. It can be seen from Table 4 that the evaluation indicators of the CDLP model are the smallest, followed by baseline models. This shows that the prediction accuracy of the CDLP model is the best.

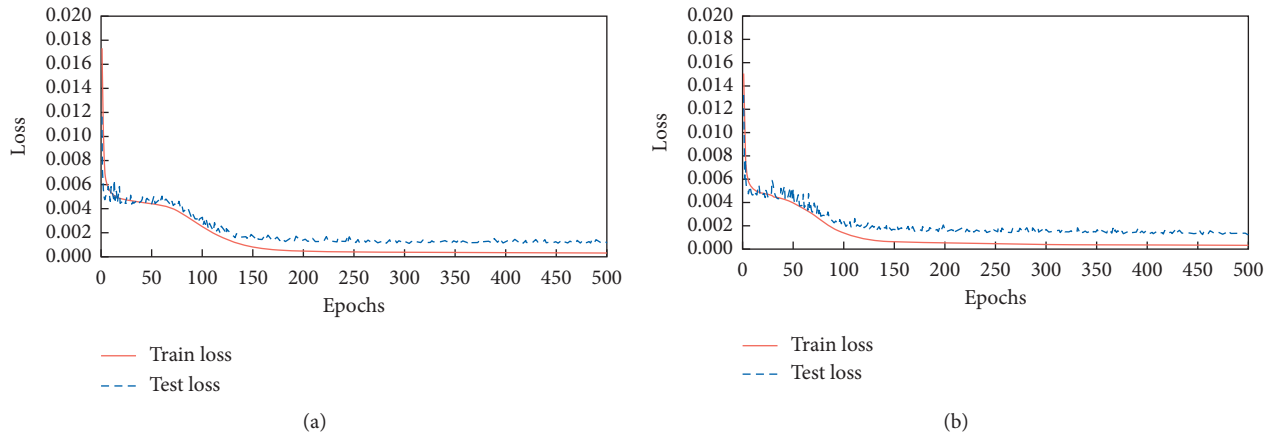


FIGURE 5: Loss function curves of the training set and test set: (a) CNN-LSTM-attention; (b) CNN-GRU-attention.

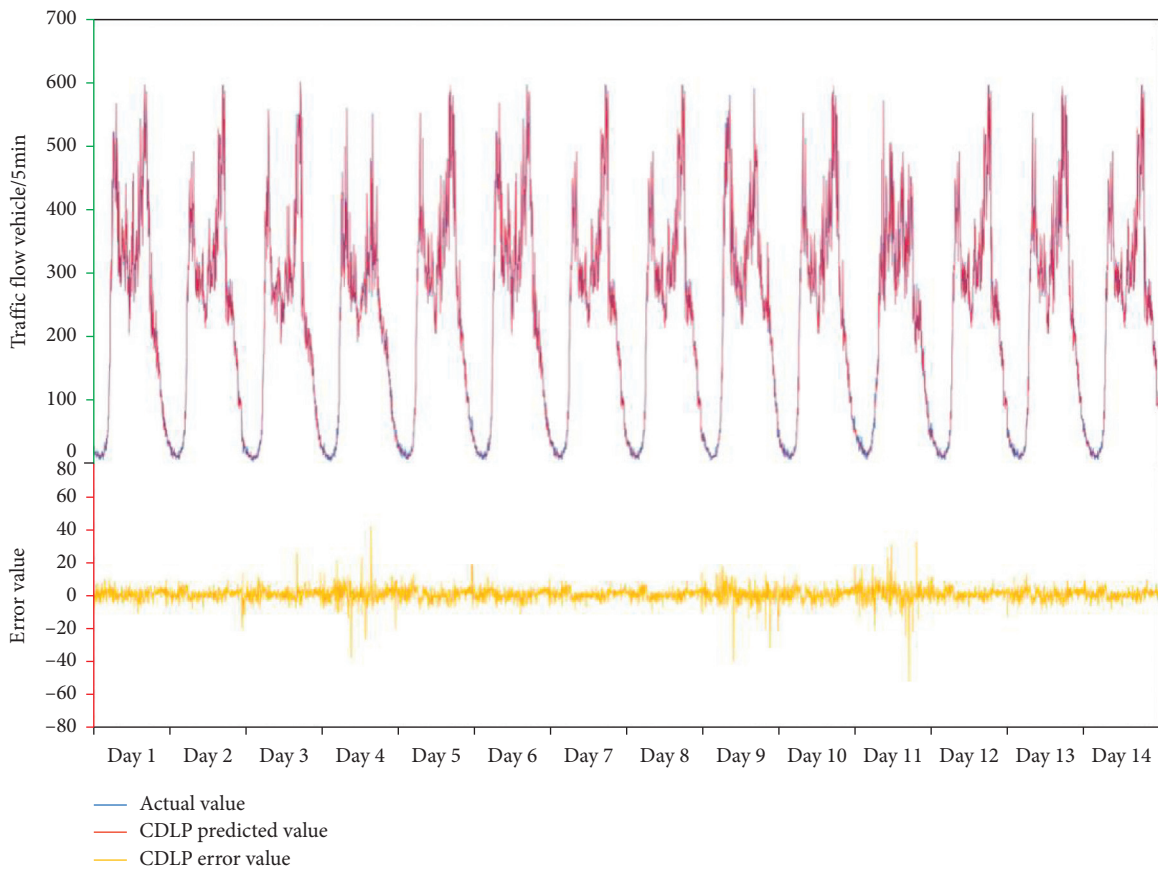


FIGURE 6: The prediction results of the CDLP model for the test set.

Moreover, it can be found from Table 5 that the training time of the CDLP model is as long as the time of the CNN-LSTM-attention model, but its prediction accuracy is higher than that of CNN-LSTM-attention model. The training time of the CNN model is the shortest, but the prediction accuracy is the lowest, so the robustness of the CDLP model is relatively high.

In addition, according to the DOWCA, the weights of CNN-LSTM-attention model and CNN-GRU-attention model in the CDLP model are calculated, as shown in Figure 9.

Figure 9 shows that the weights of CNN-LSTM-attention and CNN-GRU-attention are dynamic and constantly changing, which indicates the two methods have different

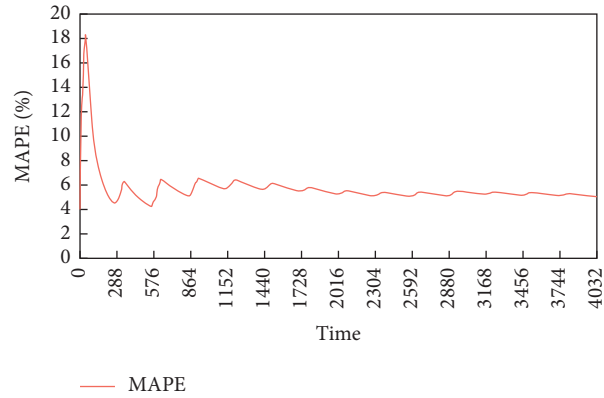


FIGURE 7: MAPE curve of CDLP model for the test set.

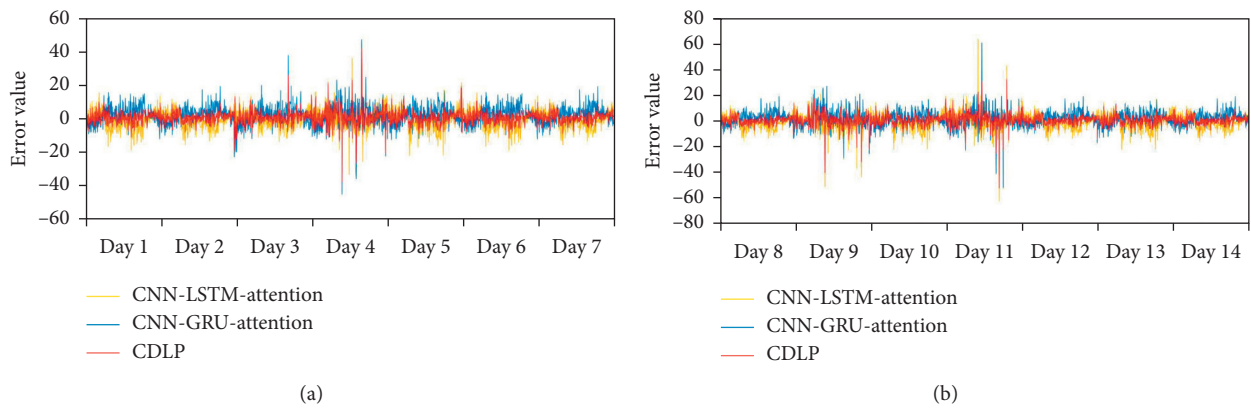


FIGURE 8: Error comparison of CDLP, CNN-LSTM-attention, and CNN-GRU-attention: (a) error of the first week in test set; (b) error of the second week in the test set.

TABLE 4: Comparison of three evaluation indexes of CDLP and baseline models.

Model	MAPE (%)	MAE	RMSE
LSTM	8.40	5.34	8.29
GRU	10.79	8.61	12.03
CNN	15.80	20.91	32.41
CNN-LSTM	7.67	4.64	6.66
CNN-GRU	7.48	4.21	6.43
CNN-LSTM-attention	6.64	4.81	7.69
CNN-GRU-attention	7.10	4.97	7.98
CDLP	5.12	3.26	6.52

TABLE 5: Comparison of model training time of CDLP and baseline models.

Model	Training time (min)
LSTM	48
GRU	42
CNN	35
CNN-LSTM	57
CNN-GRU	51
CNN-LSTM-attention	64
CNN-GRU-attention	60
CDLP	64

prediction results for the same traffic flow data. Moreover, it can be seen from Figure 8 that the weights of the two models gradually decrease from a large change at the beginning and eventually become stable, which reflects the systematic feasibility of the dynamic weighted coefficient algorithm, namely, the convergence. Furthermore, it shows that the weights of the CNN-LSTM-attention model are greater than those of the CNN-GRU-attention model, indicating that the prediction accuracy of the CNN-LSTM-attention model is higher than the CNN-GRU-attention model, which is consistent with the results in Table 4.

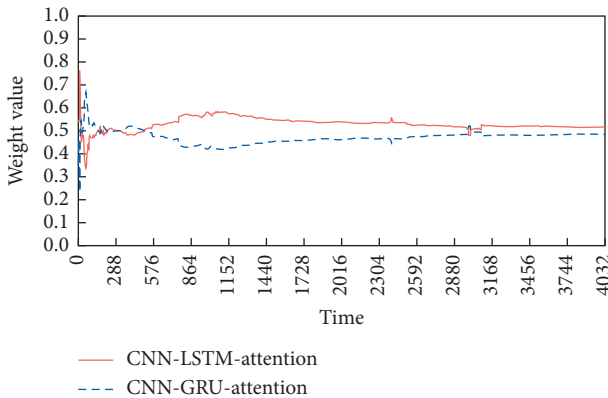


FIGURE 9: Dynamic weight value of CNN-LSTM-attention and CNN-GRU-attention model.

5. Conclusion and Future Work

Traffic flow prediction is an important part of the intelligent transportation system. In this article, a dynamic weighted coefficient algorithm for combinational prediction model is presented, namely, DOWCA. Furthermore, based on CNN, LSTM, GRU, and DOWCA, a combined deep learning model for short-term traffic flow prediction is proposed, namely, CDLP model. The structure of the CDLP model with an input layer, a hidden layer, and an output layer is designed. From the point of the combined model, the CDLP model includes two paralleled single models, i.e., CNN-LSTM-attention model and CNN-GRU-attention model. The parameters of CDLP model are determined by experiment, which includes loss function, the neuron number, and optimization algorithm.

The data from a field intersection are collected, and the dataset for the CDLP model is obtained through abnormal and missing data processing and normalization processing, which is divided into the training set and test set. The CDLP model is trained and tested. The results obtained show that the feasibility of the CDLP model can predict traffic flow with high accuracy. Moreover, in order to further verify the performance of the established model, based on the same dataset and the same parameter settings as the CDLP model, the baseline models are, respectively, used to predict the traffic flow. After analyzing the prediction results of these models, the results show that the accuracy of the CDLP model is higher than the baseline models. And DOWCA is validated to obtain the optimal weighted coefficients for CNN-LSTM-attention and CNN-GRU-attention in the CDLP model dynamically.

The structure of a CDLP model is designed and its parameters are set in this article. However, some parameters, for example, the number of nodes in the input layer and hidden layer in the model is obtained through experiments based on the selection of short-term traffic flow parameters in the past. How to optimize the parameters in a combined deep learning model needs to be further studied. Furthermore, traffic flow prediction involves several parameters; the deep learning structures based on the combinatorial algorithm can be expanded to multidimensional input variables, such as traffic speed and occupancy.

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 study was supported by China Postdoctoral Science Foundation Funded Project (2019M652437), the Scientific Research Foundation of Shandong University of Science and Technology for Recruited Talents (2019RCJJ014), Shandong Postdoctoral Innovation Project (201903030), and Key Research and Development Project of Shandong Province (2019GGX101008).

References

- [1] P. Ersoy and G. Brühman, "Intelligent transportation systems and their applications in road transportation industry in Turkey," in *Proceedings of the 12th International Conference on Logistics & Sustainable Transport*, pp. 11–13, Celje, Slovenia, June 2015.
- [2] N. Cui, B. Chen, K. Zhang, Y. Zhang, X. Liu, and J. Zhou, "Effects of route guidance strategies on traffic emissions in intelligent transportation systems," *Physica A: Statistical Mechanics and its Applications*, vol. 513, pp. 32–44, 2019.
- [3] I. O. Olayode, L. k. Tartibu, M. O. Okwu, and U. F. Uchechi, "Intelligent transportation systems, un-signalized road intersections and traffic congestion in Johannesburg: a systematic review," *Procedia CIRP*, vol. 91, pp. 844–850, 2020.
- [4] A. Richter, M.-O. Löwner, R. Ebendt, and M. Scholz, "Towards an integrated urban development considering novel intelligent transportation systems: urban development considering novel transport," *Technological Forecasting and Social Change*, vol. 155, Article ID 119970, 2020.
- [5] I. Okutani and Y. J. Stephanedes, "Dynamic prediction of traffic volume through Kalman filtering theory," *Transportation Research Part B: Methodological*, vol. 18, no. 1, pp. 1–11, 1984.
- [6] Y. Xie, Y. Zhang, and Z. Ye, "Short-term traffic volume forecasting using Kalman filter with discrete wavelet decomposition," *Computer-Aided Civil and Infrastructure Engineering*, vol. 22, no. 5, pp. 326–334, 2007.
- [7] J. Guo, W. Huang, and B. M. Williams, "Adaptive Kalman filter approach for stochastic short-term traffic flow rate prediction and uncertainty quantification," *Transportation Research Part C: Emerging Technologies*, vol. 43, pp. 50–64, 2014.
- [8] A. Emami, M. Sarvi, and S. A. Bagloee, "Short-term traffic flow prediction based on faded memory Kalman Filter fusing data from connected vehicles and bluetooth sensors," *Simulation Modelling Practice and Theory*, vol. 102, Article ID 102025, 2019.
- [9] M. S. Ahmed and A. R. Cook, "Analysis of freeway traffic time-series data by using Box-Jenkins techniques," *Transportation Research Board*, vol. 773, no. 722, pp. 1–9, 1979.
- [10] M. M. Hamed, H. R. Al-Masaeid, and Z. M. B. Said, "Short-term prediction of traffic volume in urban arterials," *Journal of Transportation Engineering*, vol. 121, no. 3, pp. 249–254, 1995.

- [11] B. M. Williams and L. A. Hoel, "Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: theoretical basis and empirical results," *Journal of Transportation Engineering*, vol. 129, no. 6, pp. 664–672, 2003.
- [12] B. M. Williams, "Multivariate vehicular traffic flow prediction: evaluation of ARIMAX modeling," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1776, no. 1, pp. 194–200, 2001.
- [13] L. Zhang, Q. Liu, W. Yang, N. Wei, and D. Dong, "An improved k-nearest neighbor model for short-term traffic flow prediction," *Procedia - Social and Behavioral Sciences*, vol. 96, pp. 653–662, 2013.
- [14] S. Cheng, F. Lu, P. Peng, and S. Wu, "Short-term traffic forecasting: an adaptive ST-KNN model that considers spatial heterogeneity," *Computers, Environment and Urban Systems*, vol. 71, pp. 186–198, 2018.
- [15] Z. Yang, Y. Wang, and Q. Guan, "Short-term traffic flow prediction method based on SVM," *Journal of Jilin University (Engineering and Technology Edition)*, vol. 36, pp. 881–884, 2006.
- [16] C. Luo, C. Huang, J. Cao et al., "Short-term traffic flow prediction based on least square support vector machine with hybrid optimization algorithm," *Neural Processing Letters*, vol. 50, no. 3, pp. 2305–2322, 2019.
- [17] J. Tang, X. Chen, Z. Hu, F. Zong, C. Han, and L. Li, "Traffic flow prediction based on combination of support vector machine and data denoising schemes," *Physica A: Statistical Mechanics and its Applications*, vol. 534, Article ID 120642, 2019.
- [18] W. Hong, "Traffic flow forecasting by seasonal SVR with chaotic simulated annealing algorithm," *Neurocomputing*, vol. 74, no. 12–13, pp. 2096–2107, 2011.
- [19] M. Castro-Neto, Y.-S. Jeong, M.-K. Jeong, and L. D. Han, "Online-SVR for short-term traffic flow prediction under typical and atypical traffic conditions," *Expert Systems with Applications*, vol. 36, no. 3, pp. 6164–6173, 2009.
- [20] H. Wang, W. Quan, and W. Y. Ochieng, "Smart road stud based two-lane traffic surveillance," *Journal of Intelligent Transportation Systems*, vol. 24, no. 5, pp. 480–493, 2020.
- [21] S. F. Wong, H. C. Mak, C. H. Ku, and W. I. Ho, "Developing advanced traffic violation detection system with RFID technology for smart city," in *Proceedings of the IEEE International Conference on Industrial Engineering and Engineering Management*, Singapore, February 2018.
- [22] A. Klein, M. Kelley, and M. Mills, "Evaluation of traffic detection technologies for IVHS," *Proceedings of SPIE*, vol. 2344, pp. 42–53, 1995.
- [23] J. Versavel and B. Boucke, "Video for traffic data and incident detection by traficon," in *Proceedings of the Fifth International Conference on Applications of Advanced Technologies in Transportation Engineering*, Newport Beach, CA, USA, April 1998.
- [24] D. Llorca, M. Sotelo, S. Sanchez, M. Ocaña, J. Rodriguez-Ascariz, and M. Garrido, "Traffic data collection for floating car data enhancement in V2I networks," *EURASIP Journal on Advances in Signal Processing*, vol. 2010, Article ID 719294, 13 pages, 2010.
- [25] L. Theis, W. Shi, A. Cunningham, and F. Huszar, "Lossy image compression with compressive autoencoders," in *Proceedings of the 5th International Conference on Learning Representations*, Toulon, France, March 2017.
- [26] L. Gao, Z. Guo, H. Zhang, X. Xu, and H. T. Shen, "Video captioning with attention-based LSTM and semantic consistency," *IEEE Transactions on Multimedia*, vol. 19, no. 9, pp. 2045–2055, 2017.
- [27] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *Proceedings of the 31st International Conference on Machine Learning*, vol. 5, pp. 3771–3779, Beijing, China, June 2014.
- [28] H.-F. Yang, T. S. Dillon, and Y.-P. P. Chen, "Optimized structure of the traffic flow forecasting model with a deep learning approach," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2371–2381, 2017.
- [29] Y. Wu, H. Tan, L. Qin, B. Ran, and Z. Jiang, "A hybrid deep learning based traffic flow prediction method and its understanding," *Transportation Research Part C: Emerging Technologies*, vol. 90, pp. 166–180, 2018.
- [30] J. Wang, R. Chen, and Z. He, "Traffic speed prediction for urban transportation network: a path based deep learning approach," *Transportation Research Part C: Emerging Technologies*, vol. 100, pp. 372–385, 2019.
- [31] W. Wang, H. Zhang, T. Li et al., "An interpretable model for short term traffic flow prediction," *Mathematics and Computers in Simulation*, vol. 171, pp. 264–278, 2020.
- [32] X. Guo and Q. Zhu, "A traffic flow forecasting model based on BP neural network," in *Proceedings of the International Conference on Power Electronics & Intelligent Transportation System*, pp. 311–314, IEEE, Shenzhen, China, January 2010.
- [33] H. Huang, Q. Tang, and Z. Liu, "Adaptive correction forecasting approach for urban traffic flow based on Fuzzyc-mean clustering and advanced neural network," *Journal of Applied Mathematics*, vol. 2013, Article ID 195824, 7 pages, 2013.
- [34] T. Li and L. Sheng, "Prediction for short-term traffic flow based on optimized wavelet neural network model," *International Journal of Computer Science & Information Technology*, vol. 7, no. 2, 2015.
- [35] W. Huang, G. Song, H. Hong, and K. Xie, "Deep architecture for traffic flow prediction: deep belief networks with multitask learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 5, pp. 2191–2201, 2014.
- [36] A. Koesdwiady, R. Soua, and F. Karray, "Improving traffic flow prediction with weather information in connected cars: a deep learning approach," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 12, pp. 9508–9517, 2016.
- [37] H. Xu and C. Jiang, "Deep belief network-based support vector regression method for traffic flow forecasting," *Neural Computing and Applications*, vol. 32, no. 7, pp. 2027–2036, 2020.
- [38] L. Han and Y. S. Huang, "Short-term traffic flow prediction of road network based on deep learning," *IET Intelligent Transport Systems*, vol. 14, no. 6, pp. 495–503, 2020.
- [39] X. Ma, Z. Dai, Z. He, J. Ma, Y. Wang, and Y. Wang, "Learning traffic as images: a deep convolutional neural network for large-scale transportation network speed prediction," *Sensors*, vol. 17, no. 4, 2017.
- [40] W. Zhang, Y. Yu, Y. Qi, F. Shu, and Y. Wang, "Short-term traffic flow prediction based on spatio-temporal analysis and CNN deep learning," *Transportmetrica A: Transport Science*, vol. 15, no. 2, pp. 1688–1711, 2019.
- [41] J. An, L. Fu, M. Hu, W. Chen, and J. Zhan, "A novel fuzzy-based convolutional neural network method to traffic flow prediction with uncertain traffic accident information," *IEEE Access*, vol. 7, pp. 20708–20722, 2019.
- [42] Q. Liu, B. Wang, and Y. Zhu, "Short-term traffic speed forecasting based on attention convolutional neural network for arterials," *Computer-aided Civil and Infrastructure Engineering*, vol. 33, no. 11, pp. 999–1016, 2018.

- [43] H. Peng, H. Wang, B. Du et al., "Spatial temporal incidence dynamic graph neural networks for traffic flow forecasting," *Information Sciences*, vol. 521, pp. 277–290, 2020.
- [44] S. Maher and P. Biswajee, "Severity prediction of traffic accidents with recurrent neural networks," *Applied Sciences*, vol. 7, no. 6, pp. 476–486, 2017.
- [45] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [46] X. Ma, Z. Tao, Y. Wang, H. Yu, and Y. Wang, "Long short-term memory neural network for traffic speed prediction using remote microwave sensor data," *Transportation Research Part C: Emerging Technologies*, vol. 54, pp. 187–197, 2015.
- [47] Z. Zhao, W. Chen, X. Wu, P. C. Y. Chen, and J. Liu, "LSTM network: a deep learning approach for short-term traffic forecast," *IET Intelligent Transport Systems*, vol. 11, no. 2, pp. 68–75, 2017.
- [48] Y. Tian, K. Zhang, J. Li, X. Lin, and B. Yang, "LSTM-based traffic flow prediction with missing data," *Neurocomputing*, vol. 318, pp. 297–305, 2018.
- [49] J. Zhao, Y. Gao, Z. Bai, H. Wang, and S. Lu, "Traffic speed prediction under non-recurrent congestion: based on LSTM method and BeiDou navigation satellite system data," *IEEE Intelligent Transportation Systems Magazine*, vol. 11, no. 2, pp. 70–81, 2019.
- [50] Z. Wang, L. Zhang, and Z. Ding, "Hybrid time-aligned and context attention for time series prediction," *Knowledge-Based Systems*, vol. 198, Article ID 105937, 2020.
- [51] J. Zhou, H. Chang, X. Cheng, and X. Zhao, "A multiscale and high-precision LSTM-GASVR short-term traffic flow prediction model," *Complexity*, vol. 2020, Article ID 1434080, 17 pages, 2020.
- [52] Y. Zhang, T. Cheng, Y. Ren, and K. Xie, "A novel residual graph convolution deep learning model for short-term network-based traffic forecasting," *International Journal of Geographical Information Science*, vol. 34, no. 5, pp. 969–995, 2020.
- [53] W. Li, X. J. Ban, J. Zheng, H. X. Liu, and Y. Li, "Real-time movement-based traffic volume prediction at signalized intersections," *Journal of Transportation Engineering Part A: Systems*, vol. 146, no. 8, Article ID 04020081, 2020.
- [54] D. Xia, M. Zhang, X. Yan, Y. Bai, and H. Li, "A distributed WND-LSTM model on map reduce for short-term traffic flow prediction," *Neural Computing and Applications*, vol. 33, no. 7, pp. 2393–2410, 2020.
- [55] S. Du, T. Li, X. Gong, and S. J. Horng, "A hybrid method for traffic flow forecasting using multimodal deep learning," *International Journal of Computational Intelligence Systems*, vol. 13, no. 1, p. 85, 2018.
- [56] G. Dai, C. Ma, and X. Xu, "Short-term traffic flow prediction method for urban road sections based on space-time analysis and GRU," *IEEE Access*, vol. 7, pp. 143025–143035, 2019.
- [57] Y. Gu, W. Lu, L. Qin, M. Li, and Z. Shao, "Short-term prediction of lane-level traffic speeds: a fusion deep learning model," *Transportation Research Part C: Emerging Technologies*, vol. 106, pp. 1–16, 2019.
- [58] C.-M. Own, F. Sha, and W. Tao, "Triplet decoders neural network ensemble system and t-conversion for traffic speed sequence prediction," *IEEE Access*, vol. 7, pp. 162070–162082, 2019.
- [59] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate, computer science," 2021, <http://arxiv.org/abs/1409.0473>.
- [60] C. Yin, X. Cheng, X. Liu, and M. Zhao, "Identification and classification of atmospheric particles based on SEM images using convolutional neural network with attention mechanism," *Complexity*, vol. 2020, Article ID 9673724, 13 pages, 2020.
- [61] L. Ye, Z. Liu, and Y. Wang, "Dual convolutional LSTM network for referring image segmentation," *IEEE Transactions on Multimedia*, vol. 22, no. 12, pp. 3224–3235, 2020.
- [62] Z. Zhao and W.-Q. Zhang, "End-to-end keyword search system based on attention mechanism and energy scorer for low resource languages," *Neural Networks*, vol. 139, pp. 326–334, 2021.
- [63] D. Li, J. Liu, Z. Yang, L. Sun, and Z. Wang, "Speech emotion recognition using recurrent neural networks with directional self-attention," *Expert Systems with Applications*, vol. 173, Article ID 114683, 2021.
- [64] J. Gu, Z. Wang, J. Kuen et al., "Recent advances in convolutional neural networks," *Pattern Recognition*, vol. 77, pp. 354–377, 2018.
- [65] G. Yang, S. Jiao, J.-P. Liu, T. Lei, and X. Yuan, "Error diffusion method with optimized weighting coefficients for binary hologram generation," *Applied Optics*, vol. 58, no. 20, pp. 5547–5555, 2019.
- [66] D. L. Guen, S. Pateux, C. Labit, G. Moury, and D. Lebedeff, "Optimal adapted weighting of wavelet coefficients for on-board coding of satellite images," in *Proceedings of the IEEE International Geoscience & Remote Sensing Symposium*, vol. 2, Honolulu, HI, USA, February 2000.
- [67] H. Wang and J. Shen, "Research on the voltage prediction of unmanned aerial vehicle photovoltaic modules based on new combination optimization algorithm," *International Transactions on Electrical Energy Systems*, vol. 29, no. 7, 2019.
- [68] Z. Qu, S. Xie, R. Feng, N. Qu, H. Lv, and Z. Chi, "Electricity sales prediction model of electricity market of the non-linear regression combination with the optimal weight," *Journal of Computers*, vol. 30, no. 5, pp. 75–87, 2019.
- [69] L. M. Castro, N. González-Cabrera, D. Guillen, J. Tovar-Hernández, and G. Gutiérrez-Alcaraz, "Efficient method for the optimal economic operation problem in point-to-point VSC-HVDC connected AC grids based on Lagrange multipliers," *Electric Power Systems Research*, vol. 187, Article ID 106493, 2020.
- [70] Q. Wang, S. Bu, and Z. He, "Achieving predictive and proactive maintenance for high-speed railway power equipment with LSTM-RNN," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 10, pp. 6509–6517, 2020.

Research Article

Linear Programming Model and Online Algorithm for Customer-Centric Train Calendar Generation

Tommaso Bosi  and Andrea D'Ariano 

Dipartimento Di Ingegneria, Sezione Di Computer Science and Automation, Università Degli Studi Roma Tre, Via Della Vasca Navale 79, Rome 00146, Italy

Correspondence should be addressed to Tommaso Bosi; tommaso.bosi@uniroma3.it

Received 26 April 2021; Accepted 31 May 2021; Published 21 June 2021

Academic Editor: Erfan Hassannayebi

Copyright © 2021 Tommaso Bosi and Andrea D'Ariano. 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.

An important objective for train operating companies is to let users, especially commuters, directly query the ICT system about trains' availability calendar, based on an online approach, and give them clear and brief information, expressed through "intelligent" phrases instead of bit maps. This paper provides a linear programming model of this problem and a fast and flexible heuristic algorithm to create descriptive sentences from train calendars. The algorithmic method, based on the "Divide and Conquer" approach, takes the calendar period queried in its whole and divides it into subsets, which are successively processed one by one. The dominant limitation of previous methods is their strong dependence on the size and complexity of instances. On the contrary, our computational findings show that the proposed online algorithm has a very limited and constant computation time, even when increasing the problem complexity, keeping its processing time between 0 and 16 ms, while producing good quality solutions that differ by an average surplus of 0.13 subsentences compared to benchmark state-of-art solutions.

1. Introduction

The European rail sector faces a number of important challenges that constitute together serious barriers for the enhancement of its attractiveness and competitiveness on the global market. This can be done through a comprehensive and coordinated approach to research and innovation and focusing not only on the needs of the rail system providers but also on the needs of the users.

Therefore, as reported in the Shift2Rail master plan [1], one of these is a quality of service challenge: rail still does not come across as a user-friendly transport mode, with 19% of Europeans simply avoiding taking trains because of accessibility issues. In today's hyper-connected society, railway customer service needs a radical rethinking to be adapted to the constant and rapid evolution of quality expectations of travellers.

Into the Shift2Rail framework, we can identify five main asset-specific Innovation Programmes (IPs), covering all the different structural and functional subsystems of the

rail system, as illustrated in Figure 1. These five IPs are not independent of one another and, into each of them, customer satisfaction represents one of the major keys. By means of this interdependence, evolutions in the technology in one part of the system can lead to changes in performance in another part. Starting from this new viewpoint, optimization covers each phase of the railway organization process moving toward perceived service quality by rail customers.

Many examples of this new research approach can be recognized in the literature: to improve travel safety, Yin et al. [2] propose a mathematical formulation to minimize the crowdedness of stations during peak hours to synchronously generate the optimal coordinated train timetables; "To Wait or Not to Wait?" is the question submitted by Schanchtebeck and Schöbel [3] for the delay management problem, in order to satisfy two different customer categories; analyzing and seeking an equilibrium point between the optimization of reordering choices of train dispatchers and the route choice of passengers in the

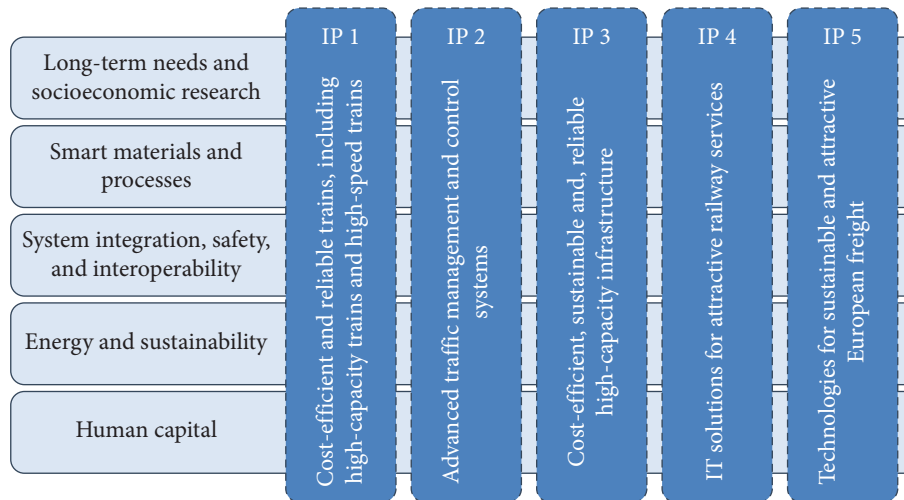


FIGURE 1: The five IPs of the Shift2Rail framework (source: Shift2Rail master plan).

available services of the railway transport network is rather the aim of Corman [4]; Hong et al. [5] define a method to solve train rescheduling and passenger reassignment simultaneously to facilitate real-time re-ticketing; some more research papers that aim to improve quality service experienced by users were selected by the special issue by D'Ariano et al. [6]; like Li et al. [7] where smart card data transactions are converted to a model of route choice, throughout the network, to improve the amount of insight into the travellers' behavior; D'Acerno et al. [8] aim to provide operations' parameters for metro systems so as to support the planning and implementation of energy-saving strategies while maintaining a targeted service level; a scalable method for dynamic profiling is introduced by Toader et al. [9] that aims to discover knowledge, like travel habits, from data in motion, and provide faster sharing mobility services in dynamic contexts; Botte et al. [10] focus on a Neighborhood Search Algorithm for optimal intervention strategies in the case of a metro system failure with the purpose to keep a certain service quality level; European journals themselves began to actively promote an ongoing special issue in order to "provide the highest quality with minimal travel cost and time to satisfy the ultimate needs of customers" (as mentioned in the call for papers of *Journal of Advanced Transportation: Advances in Modelling and Data-Driven Optimization of Urban Transport and Logistics*).

The problem studied in this paper belongs to IP2 and IP4 programmes of Shift2Rail. These two IPs aim to increase punctuality and the use of accurate and real-time data for improved passenger information, to minimize travellers' inconvenience. Based on the user's preferences, personal and secure mobile *Travel Companion* will store and share their personal preferences in a wallet, providing personalized journey and messaging help.

One of the most important and basic information for users about railways services' availability is the train calendar, which is seen by travellers as the final result of several different and complex organization processes, from

train routing to scheduling, timetable generation, and so on [11–14]. Along with basic information, such as arrival and departure times, the first data that users need to know is on which days the service is offered. Due to the present-day pace of life, it could be too much effort for a person to check a train calendar if it is expressed in the numerical form. This could bring in mistakes and forgetfulness, leading to travellers' inconvenience. Therefore, we desire to generate a more readable way to communicate trains' availability in order to ease its comprehension and memorization by users. Actually, railway companies, such as Trenitalia S.p.A., already use web services to provide trains availability information related to predetermined time-lapse through bit maps accompanied by descriptive sentences. Nevertheless, this time-lapse is defined by the company, so the customer cannot directly query the ICT system about a specific period of interest. Our purpose is to develop an online train calendar generation tool that permits, especially commuters, to ask when service is offered into an arbitrary period. Even if the problem is a niche one, the little previous literature has approached it mostly through linear programming, developing mathematical models that give optimal solutions based on assumptions made, but with temporal complexity strongly depending on the sizes and characteristics of the instances. To avoid this negative scalability feature, we propose to solve the problem with a new fast algorithmic method, developed through C programming language, which has shown constant complexity and produces good quality solutions. To test the effectiveness of this method, we have compared its performance on 264 instances, with a mathematical model strongly inspired by the third and most efficient one proposed by Amorosi et al. [14], noting how the algorithm returns similar solutions while reducing the average processing time from 381 ms to 2.32 ms.

In addition to this improvement in temporal performances, the flexibility provided by this new approach allows it to print out more detailed and suitable sentences, based on the specific transport field considered, by modulating the

core body of the algorithm code but without altering the computational complexity. A natural extension in the application of this approach can be the literal description of urban rail and bus systems' calendars, where a periodicity on the service availability also exists. In this case, given the smaller scale of instances, one could enrich the sentences' descriptions by considering, besides days' timetables, hours' ones as well, in order to share more compact availability information with passengers and to better satisfy the customer needs, which is the ultimate goal of train operating companies.

The major contribution of this paper lies in a new solving method applicable to quickly generate descriptive sentences from event calendars expressed through bit maps, in a quicker and more adaptable way than the earlier ones, in order to be applied as an online tool. Therefore, the technique presented can be extended to any application for online textual description service availability, also in other transport fields such as airplanes, ferry, and long trip buses. But this technique can be also adopted to describe the frequency of any event having a certain periodicity like sports events, opening days of commercial activities, and so forth.

To summarize, this paper considers the problem of generating descriptive sentences from train calendars, especially for commuters. The state of the art proposed a linear programming model which inspired us to create an improved model that can be easily transformed into an online tool. The main challenge of this modeling approach is the strong dependence of processing times on the complexity of the instances. To fill this gap, we propose a fast heuristic algorithm based on the "Divide and Conquer" approach, which is implemented in C language. This has been tested on 264 instances. The results report processing time much shorter than the ones required by CPLEX to solve the model, thus proving a better propensity to be used as a practical online tool.

We next introduce a brief overview of the paper structure: Section 2 reviews the most related literature, specifically the third model proposed by Amorosi et al. [15], as this is the only available research paper (to the best of our knowledge) that considers the specific train calendar generation problem in the context of the Italian railways; Section 3 describes how the ICT system works and at what point of the process of sharing our algorithm is inserted, and provides the Integer Linear Programming (ILP) model introduced to model the studied problem, which will be compared in Section 4 with the developed online algorithmic method (i.e., the heuristic algorithm) and outlines the proposed method in all its key phases through a high-level flowchart; Section 4 shows numerical simulations performed, arguing some observations about the performance values achieved by the heuristic algorithm and the ILP model and defining strengths and weaknesses of each approach; Section 5 concludes the paper by summarizing the main findings, by suggesting directions to improve the heuristic algorithm, and by summing up other possible applications of the proposed tool.

2. Literature Review

This section is mainly focused on the third model presented by Amorosi et al. [15]. The reason is related to three different considerations: first, the problem is a niche one, so there is very little material about possible solving methods [16]); second, this paper is the only one that considers the Italian railway context, with its particular passenger timetables; third, the third model is the fastest of the three models presented in their paper. Furthermore, the simulations studied in Amorosi et al. [15] are created with the active participation of Trenitalia, so they can be considered as practical cases, in which real-world characteristics have been considered for the 264 instances used in our numerical experiments.

The idea underlying the method proposed by Amorosi et al. [15] consists of several phases. First of all, the method takes as input a *periodicity*, which is a binary vector that represents the availability of the train in the time-window queried, and associates with each of its entry a progressive index (we will discuss these input data more in detail in section 3). Then, from the periodicity, only some days, in which the service is available, are extracted. The periodicity is decomposed based on 46 typologies of binary vectors, called *clusters*, which refer to a particular availability frequency, such as "all Mondays," "all Tuesdays," . . . , "Holidays" and so on, currently adopted by the main Trenitalia's ICT systems. An example of this relevant passage for the train calendar can be seen in Figure 2, considering the only "Weekends" and "Wednesdays-Thursdays" clusters.

Once all the clusters are created for the periodicity in the input, some copies are replicated for each cluster. These copies are eventually used in the case of a particular cluster that is used to represent more than one subperiod of the periodicity. The simulations show that a number of 5 copies should be enough for any kind of instance. The input data are given to a mathematical model implemented in the solver *IBM ILOG CPLEX*, which returns the minimum number of sub-vectors extracted from the clusters-copies chosen.

To choose from which cluster-copy the sub-vectors must be extracted, the following information is defined:

- (i) A quality threshold α , forcing the minimum percentage of ones that a sub-vector must contain to be feasible
- (ii) The minimum length l of the sub-vector extracted
- (iii) The start date $Y_{c,k}^I$ of the sub-vector extracted from the cluster c copy k , which is an integer variable
- (iv) The final date $Y_{c,k}^F$ of the sub-vector extracted from the cluster c copy k , which is an integer variable too

The adopted solver generates from each cluster c copy k all feasible subvectors based on the first two parameters α and l . Then, defining the start date $Y_{c,k}^I$ and the final date $Y_{c,k}^F$, the solver decides from which cluster c copy k to extract the most suitable sub-vectors for each sub-period of the periodicity. If the sub-vector chosen is populated by some zeros, those are described in the corresponding sentence as exceptions.

3	4	10	11	17	18	24	25	31	32	38	39
1	1	1	1	0	1	0	0	0	0	0	0

(a)

1	7	8	14	15	21	22	28	29	35	36	42	43	49	50
0	0	0	0	0	1	1	1	1	1	1	1	1	1	1

(b)

FIGURE 2: Examples of “Weekends” (a) and “Wednesdays-Thursdays” (b) clusters extracted from the practical case.

In the absence of the original implementation, to build up a comparison as accurate as possible and directly check the strengths and weaknesses of this approach, we have personally developed a mathematical model strongly inspired by the Amorosi et al.’s [15] model (a detailed description of this model will be given in section 4). This model has three fundamental differences with the work done in Amorosi et al. [15]:

- (1) Our model is implemented through Python, importing and using the *docplex* module, and it can thus be thought of as a stand-alone tool, implemented into a single environment. On the contrary, Amorosi et al.’s [15] model was being fed through data generated in *Microsoft .NET*, solved in *IBM ILOG CPLEX*, and the numerical solutions were then translated into train calendars by an external script. Using the *docplex* module, we have linked our optimization model with the preprocessing and post-processing phases, effectively making it an online tool, which receives input data about periodicity queried by the users and returns the corresponding descriptive sentence (Figure 3).
- (2) Before giving any input data to Amorosi et al.’s [15] model, the previous approach applied preprocessing functions that allow their model to know when a specific day was associated with a cluster and when it was not associated. On the contrary, in the new model presented in this paper, this information is expressed by a binary data $c_{d,c,k}^*$. When $c_{d,c,k}^*$ for the day d , cluster c , copy k is equal to 1, this means that day d is covered by that cluster-copy, 0 otherwise. This additional input data allow our model to avoid using a preprocessing phase and directly transfer the information on whether or not a day belongs to a cluster.
- (3) In our model, the parameter l considered by Amorosi et al. [15] to extract sub-vectors is removed, along with constraints (1). The removal of this parameter allows to lighten the model of $|C| * |K|$ constraints, where C is the set of clusters and K the set of the clusters-copies used by the model.

$$Y_{c,k}^F - Y_{c,k}^I \geq (l - 1) * x_{c,k} \quad \forall c \in C \quad \forall k \in K. \quad (1)$$

In constraints (1), to consider single scattered days also, we need to set the parameter l equal to 1, resulting in the argument to the right to be set equal to zero. With this setting, the only case in which any of constraints (1) can be violated is when the difference between $Y_{c,k}^F$ and $Y_{c,k}^I$ is

negative, that is, when $Y_{c,k}^I$ is associated with a date preceding the one associated with $Y_{c,k}^F$. However, this case is not allowed by constraints (4) and (5) in our model. As a result, there is no reason to enforce constraints (1), and we thus remove them from our model.

The simulations in Amorosi et al. [15] showed four main limitations of their approach:

- (i) The processing times are strongly dependent on the size and complexity of instances in the input, and this is not so good for an online tool. In Figure 4, we can see a scatter plot of the processing time for each of the 264 instances, expressed in milliseconds. The chart shows significant variability in the distribution of the values, with peaks up to 4.5 sec. These peaks correspond to instances populated by single scattered days. This fact is one of the weaknesses of their approach along with the low-quality solutions associated with some instances.
- (ii) The use of a percentage threshold α (based on the size of the sub-vector considered) could be a double-edged sword as, if the periodicity is quite long, the solver could not extract the most properly cluster, and add many exceptions to the sentence printed in output. During the simulations, we tried out α values equal to 80% and 90%, noting that: in the first case, the solver does not tend to extract the most suitable descriptive clusters and/or add several exceptions, when the periodicity size is particularly extended. This tendency is illustrated by Figure 5; in the second case, decreasing the number of exceptions permitted, their model has difficulties in the processing instances, which are characterized by single scattered days, both in terms of quality and computation time. For example, if the briefest description for the periodicity is “*The service is provided from Monday to Saturday from x/y to z/w*” but the time window considered is wide enough, their model solution could take a sub-vector of the cluster-copy “*Monday-Friday*”, while the corresponding solver prints out “*The service is provided from Monday to Friday from x/y to z/w except for Saturday t/y, Saturday h/y. . .*”. However, for the comparison between the different methods, we fixed the threshold to 90%, in order to prevent the solver from using too many exceptions, lowering the quality-related performance.
- (iii) The parameter l , being a fixed value, must be equal to 1, in order to cover the periodicity populated by single days as well. Due to this necessity, their model has several difficulties both in the timely processing and in

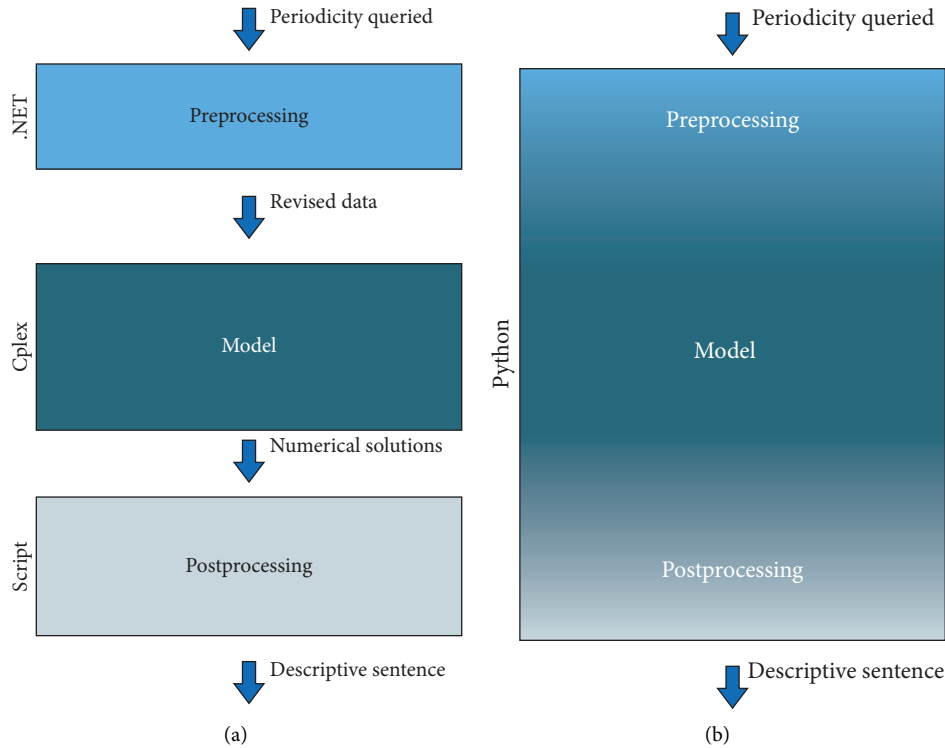


FIGURE 3: Structural difference between the previous model (a) and the newly developed model (b).

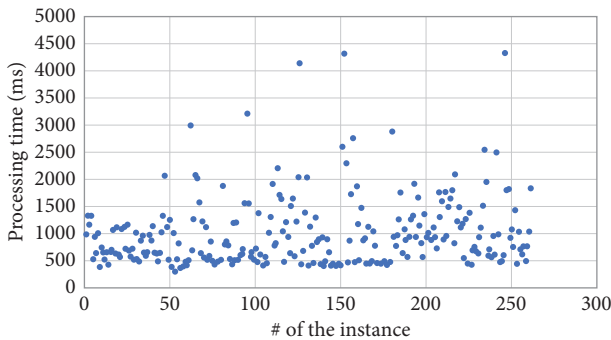


FIGURE 4: Scatter plot on the running times for the 264 instances of Amorosi et al. [15].

the association of sentences with instances with single scattered days. When their model has to process these types of instances, we observe both a considerable increase in processing times and the following solver’s decision has to extract whatever cluster-copy that covers those days, simply by modulating the start and final dates of the sub-vector. For example, if in a week the service is available only on Wednesday, their model could take a copy of the cluster “Working days,” while the corresponding solver prints out “The service is provided on working days from 13th May to 13th May.” Even though the last sentence is correct, the sentence is clearly not so effective and might confuse the users.

- (iv) Their model does not consider the possibility of inserting new clusters without involving a significant increase in processing times or storing and

adding extra days to the subsentences, instead of using one or more additional clusters. During a subperiod of the periodicity, for example, if the service is provided on Monday, Tuesday and Saturday, the sentence in output will be “The service is provided on Monday and Tuesday from x/y to z/w; on Saturday from x/y to z/w,” that is, the solver will consider two different clusters-copies. On the contrary, we could insert new clusters such as “Monday-Tuesday and Saturday” to use one cluster only and make the sentence more readable and effective. Why was this not performed in the approach proposed in the literature? Probably because considering all possible combinations among the week days would significantly increase the computational complexity of their approach. Otherwise, if the extra day concerning the cluster associated with that week is exactly one, we could print “The service is provided on Monday and Tuesday from x/y to z/y including Saturday t/y” in output.

These limitations and new features can be, respectively, eliminated and realized through the speed and flexibility of our algorithmic method, which is based on C programming language, as described in Section 3.

3. Materials and Methods

As we anticipated in the previous section, starting from a train calendar expressed through a bit map (Figure 6), we want to query a defined time window and generate the

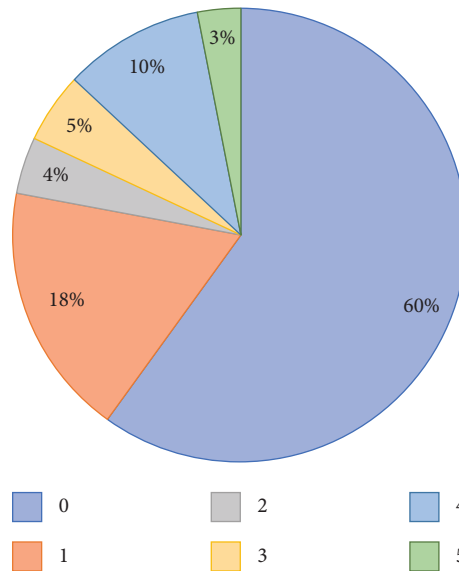


FIGURE 5: Percentages of the number of exceptions printed out by the model when the solutions differ.

clearest and briefest sentence describing the train availability. Within the ICT system, the calendars are represented by binary vectors, in which the zeros correspond to the days on which the service is not provided and the ones on which it is provided.

The binary vector based on the time window of interest and extracted from this calendar form is called *periodicity*. Figure 7 is an example of periodicity referred to the time window introduced in Figure 6. Due to the intersection between mathematical and natural language domains, there are many different ways to express the same periodicity.

In this case, for example, to represent the periodicity through a sentence, we can use the positive way,

“The service is provided on working days from 1st May to 30th June.”

Or the negative one,

“The service is not provided on weekends from 1st May to 30th June.”

However, throughout this paper, the positive way is used. For the positive way itself, different representations of the same information could be implemented for the same instance. We could say that “*The service is provided from Monday to Thursday from 1st May to 30th June except for 1st May, 8th May, . . .*” and so on, or, as we reported above, “*The service is provided on working days from 1st May to 30th June.*”

The most readable among them is for sure the second one. For this reason, we want to develop a fast tool that can automatically recognize which representation is better and prints it out. According to our study as well as previous studies, the clearest descriptive phrase is the briefest one, which is more readable and storable.

Obviously, if a service is available always or on Monday all year long, there is no need at all for optimization.

However, the use of a heuristic algorithm or mathematical programming approach is motivated by complex cases of service availability, where a nonoptimized sentence can be very long and not easy to understand. This can thus be even useless in some cases for the customers. The practice case considers the distribution of service availability during the year depending on many variables, such as customer demand, holidays, the number of trains, limited resources, and operational constraints [5, 17].

Following the practical case (as shown in Figure 8), we can see a more likely train calendar queried, along with its periodicity, populated by different subperiods of train availability and by exceptions and days in surplus.

The most concise descriptive phrase for the periodicity below could be:

“*The service is provided on the weekends from 3rd July to 18th July except for Saturday 17th July; on Wednesdays and Thursdays from 21st July to 5th August including Friday 6th August; on working days from 9th August to 20th August.*”

We generate this type of sentence by inserting the train calendar’s periodicity as input, which is a binary vector. In Section 2, we mentioned the ILP model implemented to solve the studied problem, in order to compare the literature approach with the algorithmic approach presented in this paper. In the following, we provide its description.

3.1. Notations and Mathematical Formulation. We consider the following input data:

O = set of operating days associated with the periodicity (in which the service is provided).

C = set of clusters.

$C_{c,k}$ = k -th copy of cluster $c \in C$.

$d_{d,c,k}^*$ = (position of the date d in $C_{c,k}$) + 1.



FIGURE 6: Example of a train calendar.

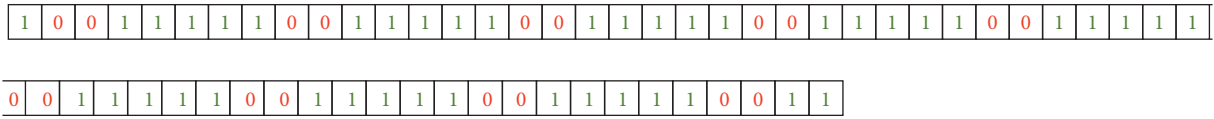


FIGURE 7: Example of periodicity extracted from the train calendar.

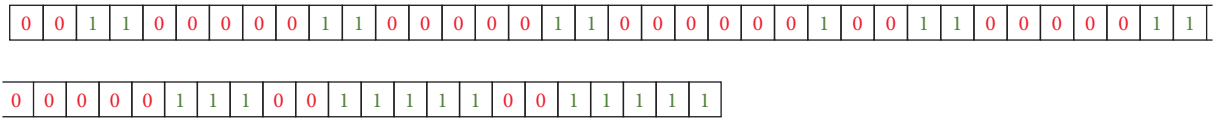
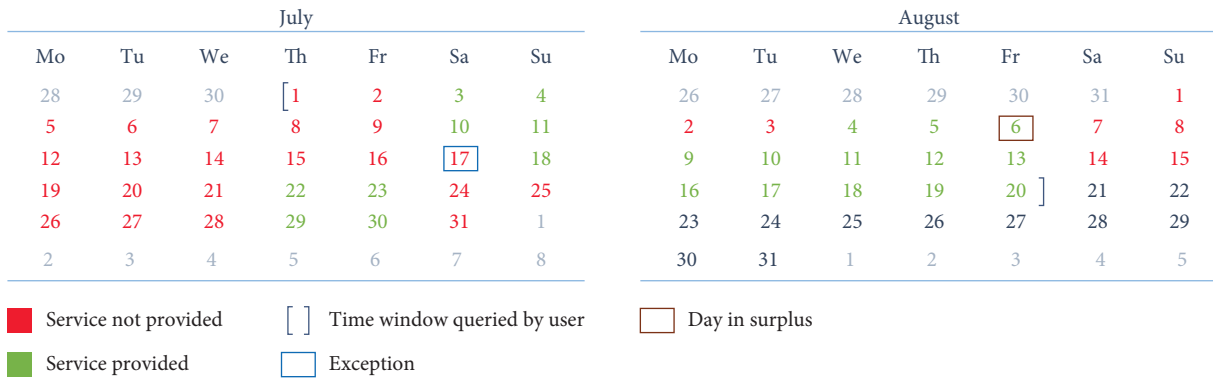


FIGURE 8: A practical case.

$c_{d,c,k}^* = \{ 1, \text{ if the date } d \text{ is covered by the copy } C_{c,k} \text{ of cluster } c, 0, \text{ otherwise.}$

$\alpha =$ minimum percentage of ones that the sub-vectors must have to be feasible.

Tot = cardinality of the periodicity.

$M =$ big integer.

We define the following decision variables.

(i) Integer variables:

$Y_{c,k}^I =$ integer representing the start position of the sub-vector extracted from the copy $C_{c,k}$ of cluster c

$Y_{c,k}^F =$ integer representing the final position of the sub-vector extracted from the copy $C_{c,k}$ of cluster c

(ii) Binary variables:

$K_{d,c,k} = \{ 1, \text{ if the date } d \text{ is covered in the solution by the copy } C_{c,k} \text{ of cluster } c, 0, \text{ otherwise.}$
 $x_{c,k} = \{ 1, \text{ if the the copy } C_{c,k} \text{ of cluster } c \text{ is chosen in the solution, } 0, \text{ otherwise.}$

Through these input data and decision variables, we present the following ILP model:

$$\min \sum_{c \in C,k} x_{c,k}. \tag{2}$$

subject to

$$\sum_{d \in O} K_{d,c,k} * c_{d,c,k}^* \geq \alpha * (x_{c,k} + Y_{c,k}^F - Y_{c,k}^I) \quad \forall c \in C \quad \forall k \quad (3)$$

$$Y_{c,k}^F \geq d_{d,c,k}^* * K_{d,c,k} \quad \forall d \in O \quad \forall c \in C \quad \forall k \quad (4)$$

$$Y_{c,k}^I \leq (1 - M) * d_{d,c,k}^* * K_{d,c,k} + d_{d,c,k}^* * M \quad \forall d \in O \quad c \in C \quad \forall k \quad (5)$$

$$\sum_{c \in C, k} K_{d,c,k} * c_{d,c,k}^* \geq 1 \quad \forall d \in O \quad (6)$$

$$x_{c,k} \geq \frac{\sum_{d \in O} K_{d,c,k}}{Tot} \quad \forall c \in C \quad \forall k \quad (7)$$

As we can see in the mathematical models (2)–(7), the objective function minimizes the number of sub-vectors extracted, represented by $x_{c,k}$, the binary variable associated with the cluster c and copy k (2). The following constraints explain, respectively, that the percentage of ones into a feasible sub-vector must be greater than or equal to the threshold α (3); constraints (4) and (5) impose that the two integer variables, $Y_{c,k}^I$ and $Y_{c,k}^F$, of a feasible sub-vector take progressive indexes, respectively, before the first date and after the last date covered by the sub-vector that they are associated with; every date in which the service is available must be covered at least one time (6); if a cluster c and copy k are chosen in the solution to cover at least one date of the periodicity, the variable $x_{c,k}$ associated with this must be activated (7). Based on preliminar numerical simulations, we have set the threshold α equal to 0.9. The reason is that if we set this threshold to a lower value, the solver would generate a less fitting sub-vector along with many exceptions; on the contrary, if we set it to a higher value, the solver would take too much time in processing the tested instances.

We will next refer to this mathematical model as “the model,” in order to get the reading smoother.

At this point, we can describe the proposed heuristic algorithm. First of all, as an online tool, it interacts with users by accepting the input data queried: start and final dates of the time window of interest. To avoid the use of external one-time filled calendars, as happened in previous literature, the second step plans to generate an internal calendar through mathematical formulas. Then, the periodicity generated by the ICT systems is taken and divided into the weeks that make it up. We can see how the algorithm approach is quite different from the previous one, indeed; while the model considers the periodicity as a whole, the algorithm follows the “Divide and Conquer” approach [18]: taking the problem in its entirety and dividing it into less complex subproblems. The idea here comes from the *Work Breakdown structure* [19] activity, used to divide large projects into project segments, called *leaves*, to be assigned to each operating unit.

Our algorithmic method is implemented in C language. The decision to utilize a programming language, like C, is related to two different reasons: this is a specific request of

the train operating company; it is a compiled one, therefore it is faster than a programming language interpreted [20]. Our first aim was to mitigate the variability and the duration of processing time employed by past literature, in order to develop a more practical tool. Furthermore, we wanted the possibility to insert more clusters based on the particular railway context to improve the briefness of sentences printed and the chance to create useful functions, such as the one related to extra days. But how does the heuristic algorithm work? The code is divided into two macro-phases: a preparatory one that manipulates the periodicity, and the next one that processes the periodicity itself. The main phases of the proposed heuristic are illustrated by the flowchart in Figure 9.

After we have carefully manipulated the input data, we need a tool that transforms the subperiods of initial periodicity into sentences. This one is a classical C data structure called the *Box*. The Box is like a single machine that receives the WIPs, that is the weeks, through a conveyor belt and does something and pulls out the final products, which are the sentences. The conveyor belt is represented by a while loop, which is the core of the second phase. The exit condition for the while loop is connected with the number of weeks of the input periodicity: the while loop goes on until the last week has not been processed.

What does the machine do exactly to generate a solution to the studied problem? Into the Box, the weeks are stored three at a time. A descriptive cluster of the days, on which the service is offered, is assigned with the week in the first position. Later, the weeks stored in the second and third positions are compared with the one in the first position, depending on the case, to check if they are different, similar, or equal.

This comparison is done based on the days of the week in which the service is available:

- (i) If two weeks have the same days on, this means that the service is offered on the same days and so the compared weeks can be considered as equal. The second week is thus “incorporated” into the first one. This means that the cluster associated with the first week is also associated with the second one, and when the sentence, which describes the service availability of these two weeks (based on the cluster associated) is printed out, a specific time window will be considered, which starts from the first day of availability of the first week to the last day of availability of the second week.
- (ii) If two weeks differ from each other by only one day of unavailability or availability, this means that this day could be, respectively, an exception or an extra day. If this is the case, these weeks can be considered as *similar*. To check if it is true, we compare also the first week, with the third one, and if they are the same, the similarity is confirmed. The second week is thus incorporated into the first one, and the day apart is labeled as an extra one or an exception, based on the case.

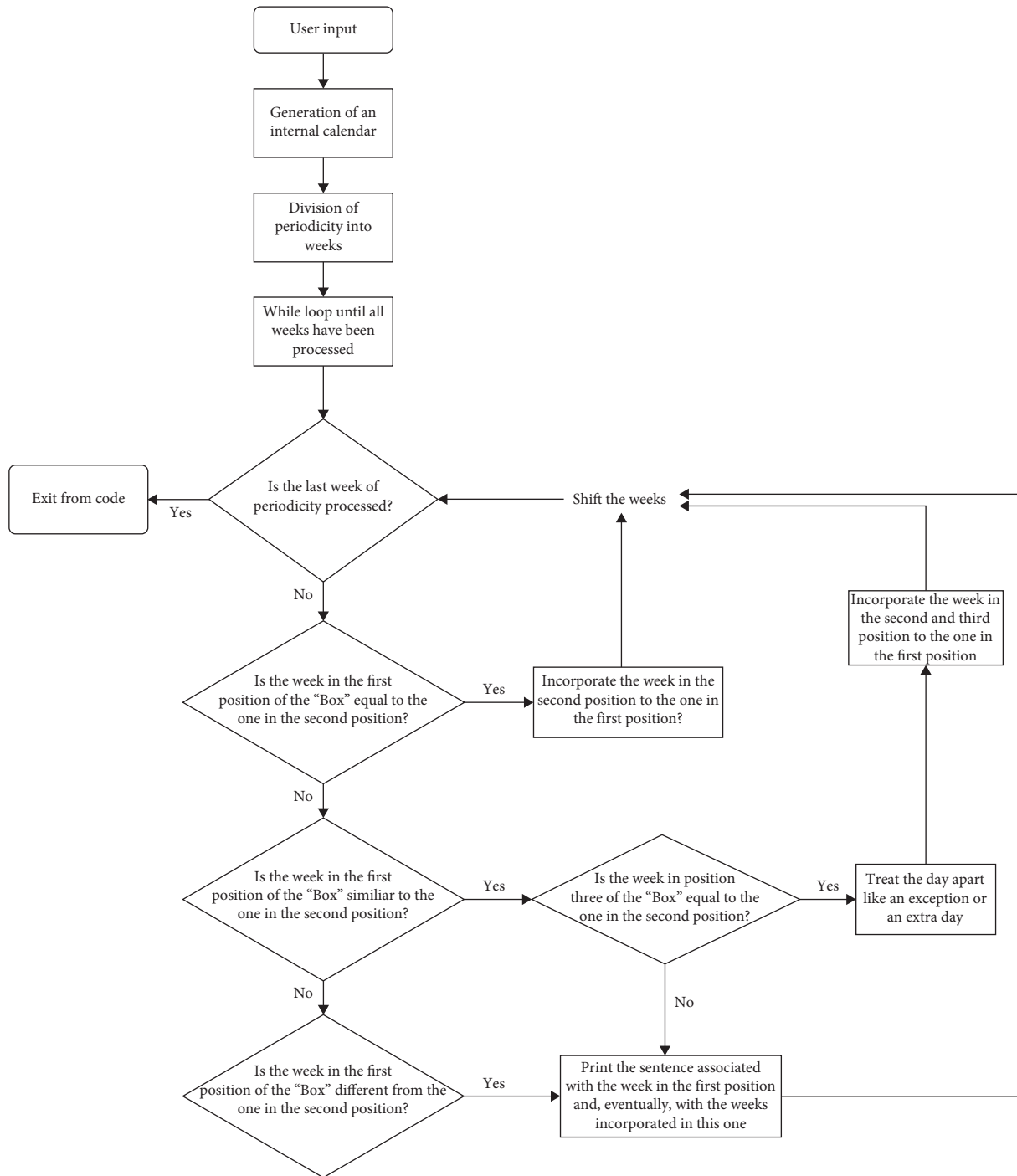


FIGURE 9: Flowchart of the proposed heuristic algorithm.

(iii) If two weeks differ from each other by two or more days of unavailability or availability, we consider these as different. In this case, we do not incorporate the second week with the first one, and we represent the two weeks, and the ones incorporated with them, with different clusters. Therefore, the algorithm will print out different sentences from these two subperiods.

An example for each case is shown in Figure 10. Once we know if the week in the second position of the Box is different (c), similar (b), or equal (a) to the one in the first position, a function scrolls the weeks in order to sequentially process all the weeks of the periodicity.

Once all the weeks have been processed and the sentences associated with the subperiods have been printed out, the code exits from the while loop. However, for the practical use of this

tool, we will consider the implementation of an infinite loop, such as a *while True*, containing the whole code along with a break condition. Another important feature (that solved an important limitation of the previous method) is that, thanks to the flexibility of the C programming language, we inserted other clusters in addition to the starting 46, such as the cluster “Monday-Tuesday and Saturday” and all possible combinations between different days of the week, without altering the computational complexity, as for the code they are only strings to search. This added feature allows us to reduce the number of clusters used to print out a solution. As we will discuss in the next section, this approach ensures that a good constructive solution is generated with constant complexity and the clusters are adapted to the specific rail context in which the train operating company is part.

To better explain how the workflow of our algorithm operates, a step-by-step description of the activities performed on a trial instance is presented. We assume that the period queried by a user is composed of only four weeks, from 1st March to 28th March (Figure 11).

First of all, the algorithm takes the input data, start and final dates of the time window, along with the periodicity associated with the specific train requested, and creates the internal calendar, which is a matrix. Then, the periodicity is divided into its weeks based on the weekdays assigned to each progressive index through the following mathematical formula. The variables in this formula are described in Table 1.

$$d_w = \text{mod}\left(\frac{y + (y - 1/4) - (y - 1/100) + (y - 1/400) + d_c}{7}\right). \quad (8)$$

For example, to figure out which day of the week corresponds to 10th January 2020, we will consider $y = 2020$ and $d_c = 10$, obtaining a d_w equal to 6, that is, Friday.

The data structure “Box” takes the first three weeks, associates the cluster “Monday-Wednesday” with the week in the first position, and starts the comparison between the latter and the one in the second position (Figure 12).

Since they are not equal but differ by one day only, Wednesday 10th March, the algorithm checks if they could be similar or different by comparing the week in the first position with that in the third position. The latter is equal to the week in the first position, so the similarity is confirmed. Consequently, the weeks in the second and third positions are incorporated with the one in the first position. Wednesday 10th March is stored as an exception and the time-lapse of cluster “Monday-Wednesday” is updated, starting from Monday 1st March to the last day of the week in the third position, which is Wednesday 17th March. The weeks in the second and third positions are scrolled, and the fourth week enters the “Box.” The latter is compared with the week in the first position and, as they have only two days in common, they are classified as “different.” The cluster associated with the week in the first position is then printed out as a subsentence of the periodicity, the fourth week is inserted in the first position of the “Box,” and a new cluster is attached with it, that is, the cluster “on Weekends” on Saturday 27th and Sunday 28th March.

Since all the weeks are processed, the counter meets the exit condition and the while loop ends. The sentence connected to the periodicity is thus displayed, as we can see in Figure 13.

Once the algorithm has been described, some differences with the literature approach can be identified:

- (i) The modeling approach looks at the instance in its entirety and creates the clusters-copies over the overall time window queried. This leads to an increase in the computational effort to compute the best possible solution. On the contrary, the heuristic algorithm looks at the instance week by week, generating and associating, when required, a cluster of seven days with the week in the first position of the box (Figure 14). This cluster is compared with the weeks ahead, and no new cluster is generated until there exists a significant difference between two weeks, in terms of days of service availability. This trade-off in the view and segmentation of the periodicity allows to sharply reduce the processing times.
- (ii) Through the modeling approach, introducing a new cluster c_1 to improve the solution quality would lead to the computation of that cluster over the whole instance along with the replication of all its copies. Differently, the proposed heuristic algorithm implements clusters through strings and, therefore, the most considerable strain lies in the association of each week with a cluster, performed through the sorting of arrays of a length of seven. This allows to easily insert new clusters with a considerably reduced computational effort compared to the model, in order to improve the quality of descriptive sentences.
- (iii) Another relevant difference between the model and the algorithm is the introduction of a function that considers extra days between weeks. Let us assume that, in the period of interest, the service is offered with the same frequency, except for some scattered weeks, in which there is an extra day, as in the case of festivities. In this case, the modeling approach will associate at least two clusters with the periodicity, and therefore, two subsentences. Through the extra days’ function implemented in the algorithm, these days will be stored as extra days and added to the single cluster choices to describe the periodicity. This sample function is an example of the flexibility of this type of tool and its potential to be adapted to different public transport contexts, to increase the service quality perceived by the users.

4. Results and Discussion

In order to compare how the two methods are presented in this work, we tested them on 264 instances imported from a .csv file. While the model was developed and executed on the IDE Pycharm 2020.3.3, for the heuristic algorithm we used Code:Blocks 20.03. All tests were performed on a Windows

	Week 1		Week 2
Monday	1	=	1
Tuesday	1	=	1
Wednesday	1	=	1
Thursday	0	=	0
Friday	1	=	1
Saturday	0	=	0
Sunday	0	=	0

(a)

	Week 1		Week 2
Monday	1	=	1
Tuesday	1	=	1
Wednesday	1	=	1
Thursday	0	=	0
Friday	0	!-	1
Saturday	0	=	0
Sunday	0	-	0

(b)

	Week 1		Week 2
Monday	1	=	1
Tuesday	1	=	1
Wednesday	1	=	1
Thursday	0	!=	1
Friday	0	!=	1
Saturday	0	!=	1
Sunday	0	=	0

(c)

FIGURE 10: Examples of the three possible comparison results.

operative system Intel i7 processor with 2.6 GHz and 16 GB of RAM. The whole 264 instances are created based on the ones presented in the study of Amorosi et al. [15] and on the potential weakness of each method, that are, respectively, the less effective processing of the descriptive cluster “Every day,” when it is used as a solution by the heuristic algorithm; the poor performances, in terms of efficiency and effectiveness, generated by the model approach in processing periodicities populated by single scattered days.

We test the ILP model and the heuristic algorithm for various possible situations in order to quantitatively evaluate their potential. We also investigate the exponential time and the high number of clusters used by the model of Section 3 to process periodicity populated by single days. This creates “rare events” that significantly alter the performance indexes employed. Therefore, the periodicities described in whole or in part by the cluster “Every day” represent 24%, while the ones populated by single days are 4% of all the tested cases.

The experiments consist of three different sets of periodicities, updated to 2020: the first one is composed of 62 instances ranging from 13th September to 24th November 2020; the second one contains 88 instances ranging from 3rd May to 2nd August 2020; the last one is made up of 113 instances from 2nd February to 20th May 2020. So, the sets cover, respectively, segments with a length equal to two, three, and almost four months.

We considered both temporal and quality indexes, precisely: average, maximum, and minimum processing times; average, maximum, and minimum numbers of descriptive clusters used. The values obtained for each index are indicated in Table 2.

Two other important indexes that we can consider are: the number of different clusters used by each method when two different solutions are printed out; the number of exceptions used by each method when the solutions differ by the number of clusters used. This last index is important due to the

March						
Mo	Tu	We	Th	Fr	Sa	Su
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4
5	6	7	8	9	10	11

■ Service not provided Time window queried by user
■ Service provided Exception

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28
1	1	1	0	0	0	0	1	1	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	1	1

FIGURE 11: Train calendar and periodicity of the trial instance.

TABLE 1: List of symbols used in mathematical formula (8).

Symbol	Description
d_w	Values ranging from 0 to 6, where 0 is associated with the day of week “Saturday” and 6 with “Friday”. Therefore, for “Monday” this value will be equal to 2.
y	Current year. For the simulations, we consider this value equal to 2020.
x	The integer portion quotient x inside the brackets.
d_c	The number of the year-to-date days, starting from 1 st January to which we associate the value 1.
mod	The modulo operator, which returns the remainder of the division inside the round brackets.

The “Box”				
	Week 1		Week 2	Week 3
Monday	1	=	1	1
Tuesday	1	=	1	1
Wednesday	1	!=	0	1
Thursday	0	=	0	0
Friday	0	=	0	0
Saturday	0	=	0	0
Sunday	0	=	0	0

Week 4
0
0
0
0
0
1
1

FIGURE 12: Illustration of the data structure “Box.”

Week 1	Week 2	Week 3	Week 4
1	1	1	0
1	1	1	0
1	0	1	0
0	0	0	0
0	0	0	0
0	0	0	1
0	0	0	1

The service is provided: from Monday to Wednesday from 1st March to 17th March except for 10th March; on the weekend of 27th and 28th March.

FIGURE 13: The sentence printed out for the considered trial instance.

TABLE 2: Values measured for each method based on the performance indexes considered.

Time	ILP model	Heuristic algorithm
Average	1000	2.32
Max	4334	16
Min	300	0
Number of clusters		
Average	1.19	1.32
Max	6	5

Note: time is expressed in ms.

limitation identified with the use of a percentage threshold by the model. In fact, what can happen is that the model may use fewer clusters, but along with many more exceptions.

As we can see in Figure 15, the percentage of periodicities to which the two methods associate different number of clusters is 30% of the whole tests. Eighty-two percent of the solutions of this 30% differ by one cluster, 9% differ by two clusters, and the other 9% differ by three or more clusters.

However, the latter 9% can be traced back to the difficulties of the model to process instances populated by single days. The following figure (Figure 5), rather, represents the percentages of the number of exceptions inserted in the sentences printed out by the model when the two methods used a different number of clusters to describe each periodicity.

The following plots (Figure 16) consider, respectively, the processing time of each periodicity tested, and the number of clusters used to print out the solution by each method.

The results in Figure 16 show how the proposed heuristic algorithm solves our initial research questions and the limitations identified from the past literature. First, we were looking for an online tool that could interact with rail users. Due to this specific feature, the tool has to be very fast. To achieve this goal, the past literature made use of external tools, such as.NET and scripts in addition to their model, thus increasing the resulting processing times. The model

and the heuristic algorithm proposed in this paper are both stand-alone tools. Figure 17 illustrates the distribution of processing times into specific time frames and, as we can see, 40% of them exceeds the seconds up to 4.3. This can be a problem in practice, to maintain fast response times, while considering the time required for transfer of information to and from the server. Unlike the model, the heuristic algorithm maintains a constant complexity, as proved by Figure 16 and Table 2, which is independent from the length or complexity of the considered instance, with processing times ranging from 0 to 16 ms and an average processing time of 2.32 ms. A constant computational complexity means that developing an online tool based on our algorithm would not be subject to significant variability of the response times to the users' queries.

Second, the number of clusters used by the model is up to 6 per instance, while the one used by our heuristic algorithm is up to 5 per instance. Even though this is a slight difference, to properly assess the quality of the provided solutions, we should consider which clusters are used to print out the descriptive sentences as well. Indeed, looking at Figure 18 on the distribution of the number of clusters, in correspondence with the solutions that exceed the three clusters used by the model, there are instances populated by single scattered days, for which the solver tends to extract whatever cluster covers the corresponding single day, regardless if this is the most proper one, by

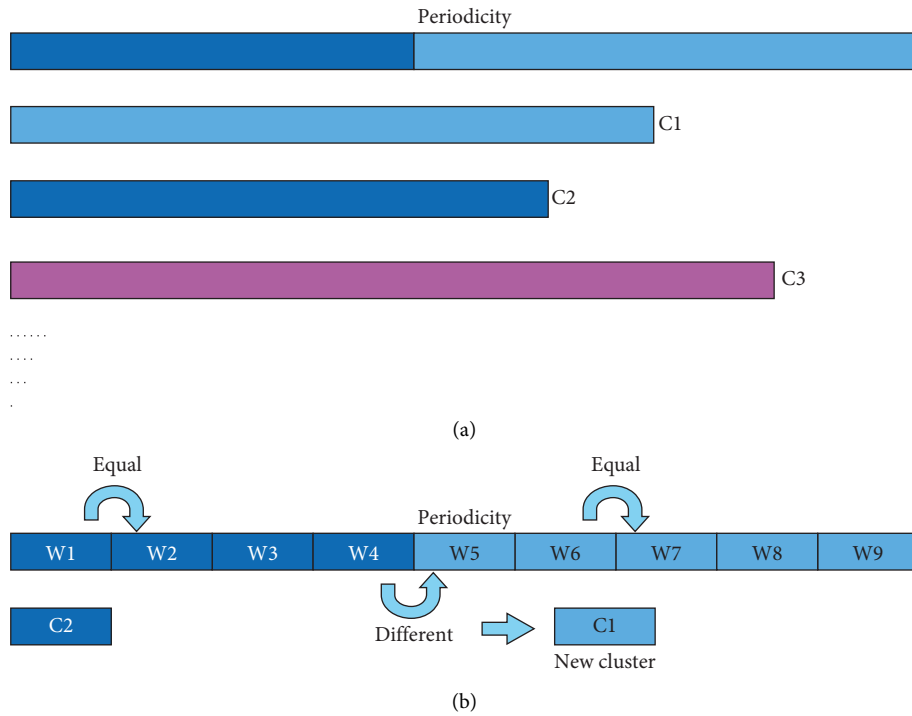


FIGURE 14: Methodological differences between the model (a) and the heuristic algorithm (b).

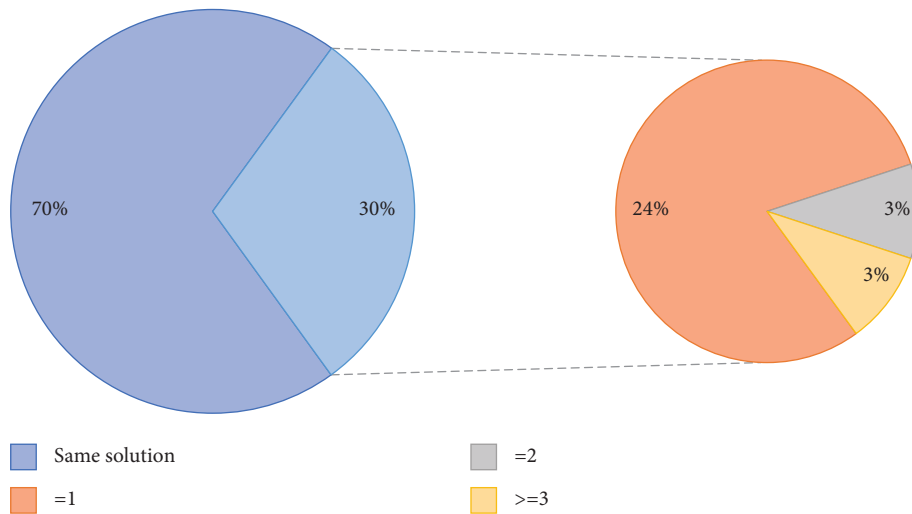


FIGURE 15: Percentage of different solutions used by each method, along with the number of different clusters used.

modulating the endpoints $Y_{c,k}^I$ and $Y_{c,k}^F$. This leads to a lowering of solution-quality and an increase in the likelihood of confusing the users. Differently, the algorithm works very well with single days, since this does not require the use of any threshold value, which could insert many exceptions, to minimize the studied objective function. Furthermore, our algorithm returns the most effective descriptive cluster that could be associated with these subperiods, keeping away from the modulation of the start and final dates of whatever cluster covers them. For example, if the single day on which the service is offered is Wednesday, the algorithm will associate the cluster

“Wednesday” with that week instead of using the cluster “Working days” and thus choosing the $Y_{c,k}^I$ and $Y_{c,k}^F$ values which correspond to that particular Wednesday.

When looking at the quality performance measurements in Table 2 and Figure 18, the heuristic algorithm makes use of 0.13 extra clusters compared to the model, which computes the optimal solution. Furthermore, even though the distribution of the instances with two clusters is higher in the solutions provided by the heuristic algorithm compared to the ones computed by the model, the algorithm avoids exceeding three clusters to solve these benchmark instances. This means that the algorithm behaves more consistently, avoiding rare events that lead to

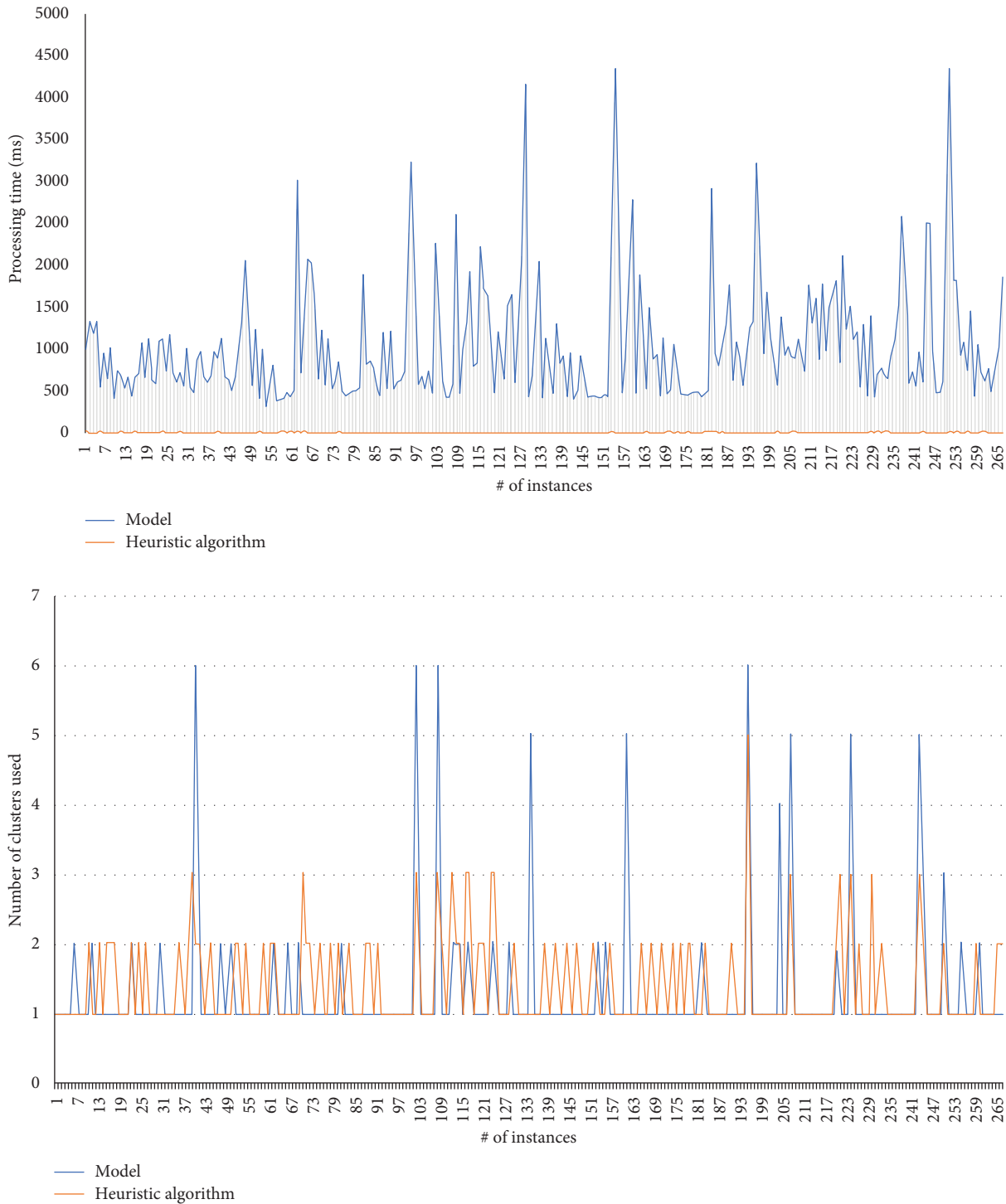


FIGURE 16: Running times and clusters used for the 264 instances.

important disservices, while sharing information with the users. Furthermore, the algorithm identifies more descriptive clusters and introduces new and more flexible functions (e.g., information on the extra days) without increasing the overall processing time. This additional flexibility reduces the number of clusters associated with each solution and improves the resulting solution quality. Differently, in the modeling approach, these

additional functions might lead to new constraints and variables, thus potentially increasing the computational complexity.

However, there are two main limitations that we have met: the “*Every day*” cluster is worked more effectively by the model, while the algorithm tends to unpack the beginning and end times of each period. For example, if the solution generated by the model is “*The service is provided*

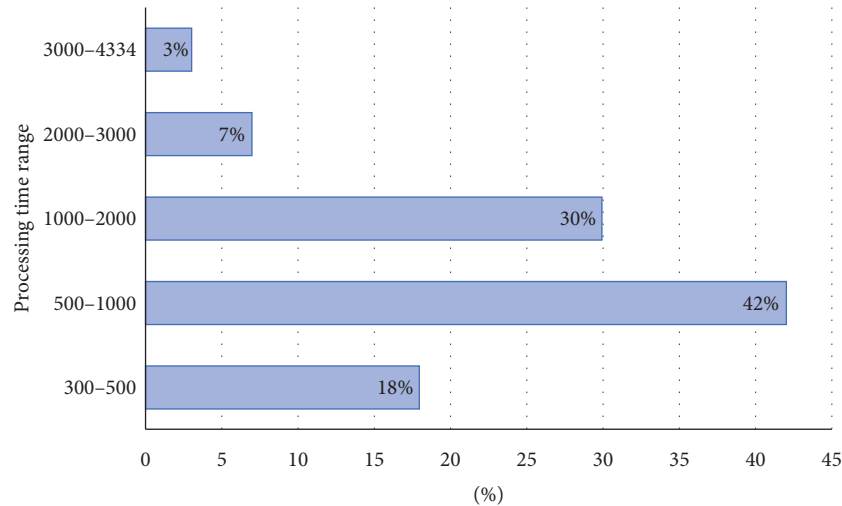


FIGURE 17: Distribution of processing times for the model.

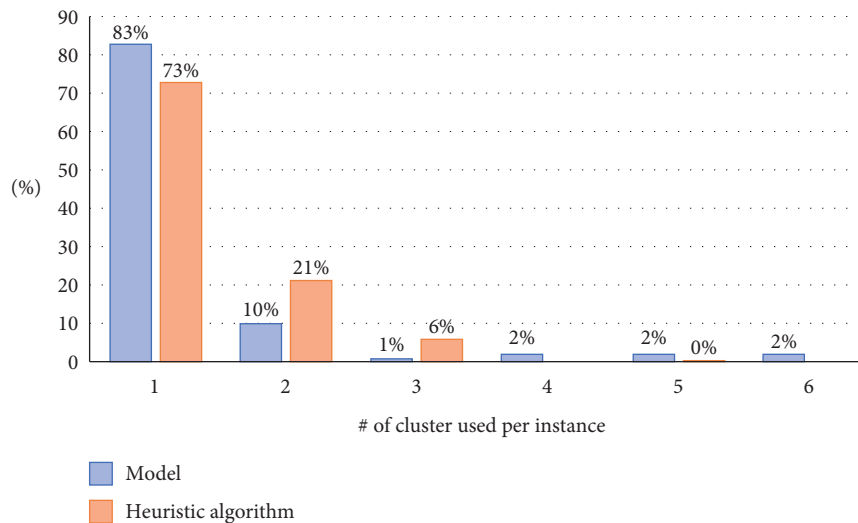


FIGURE 18: Percentage distribution of the number of clusters used by the model.

every day from x/y to z/w,” the heuristic algorithm, which works week by week, could break it down into two different clusters; the second limitation is actually linked to the characteristics requested by the train operating company itself. In fact, while the model can work on a periodicity over the years, the algorithm can only consider one year at a time. This is because the train operating company requests a tool that would work from six months to six months.

5. Conclusions

In this paper, we developed a fast and flexible alternative method to the approach of state-of-the-art for train calendars’ textual generation that mainly allows us to maintain a constant computation time, return good quality solutions, and introduce new functions to enhance the effectiveness of the sentences to be printed out. Theoretical and practical contributions lie in a new ILP model along with a fast heuristic

algorithm for solving the online train calendar generation problem. The ILP model has three main differences with Amorosi et al.’s [15] model: our model considers a new data $c_{d,c,k}^*$ with the aim of avoiding preprocessing functions which were used by Amorosi et al. [15] to filter input data; the parameter l in their model, employed to define a bond to the length on the subtences extracted, is not considered in our model along with the $|C| * |K|$ constraints involving this parameter; our model is embedded in Python, which allows the integration both of a preprocessing phase of the input data and a postprocessing phase of output data into a single environment, thus improving the online interaction with the users and the speed of generating train calendar descriptive sentences that were performed by an external script in Amorosi et al. [15].

Regarding the algorithmic contribution, the heuristic proposed in our paper exploits a “Divide and Conquer” logic to tame the whole periodicity through the generation of week-by-week solutions. From the computational

experiments, our algorithm presents, on average, a strong (equal to 99.8%) processing time reduction compared to the modeling approach, while remaining below 1.32 sub-sentences per periodicity. Due to its fast response time and its ability to compute good quality solutions, the proposed algorithm can be considered as the best choice to develop a valid online tool for train calendar generation.

Is it possible to go further in this direction? Of course, there are some open issues. The first thing that we could improve is the processing of the “Every day” cluster, giving the algorithm the possibility to predict that. What we mean is to allow the algorithm to intelligently understand if a subperiod that can be effectively described by the “Every day” cluster has begun. This should further reduce the average number of clusters used.

Another important way to improve the current results could be to reconsider the initial assumptions on what we consider a readable and intelligent sentence based on a more realistic perceived quality by the user, for example, through the perceived service quality model proposed by [21]. What if the minimum number of clusters is not a quality parameter for the sentences? Before we began developing the current algorithm, we intended to start a survey campaign among university commuter students, but this was not fully possible due to the restrictions imposed by the covid-19 situation. This could have allowed us to confirm our logical assumptions or reshape them based on new considerations from the perceived quality of service.

Moreover, the flexibility demonstrated by the proposed algorithm enables it to be adopted to develop online tools for event calendars’ description in other public transport contexts with different problem specifications and descriptive sentences to be printed out. From the ground to air transport, the procedural method expressed by the main body of the algorithm could be populated by additional features, and be addressed not only to external users but also to the transfer of information for internal staff, as in the case of freight transport sector.

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 there are no conflicts of interest regarding the publication of this article.

Acknowledgments

The authors would like to thank Dr. Gianluca Giacco (Trenitalia senior manager) and Dr. Lavinia Amorosi (Sapienza University researcher) for their feedback on the problem and for generously sharing railway data.

References

- [1] Strategic Master Plan, *Shift2Rail Joint Undertaking*, Strategic Master Plan, Brussels, Belgium, 2015.
- [2] J. Yin, A. D’Ariano, Y. Wang, L. Yang, and T. Tang, “Timetable coordination in a rail transit network with time-dependent passenger demand,” *European Journal of Operational Research*, 2021.
- [3] M. Schachtebeck and A. Schöbel, “To wait or not to wait—who goes first? delay management with priority decisions,” *Transportation Science*, vol. 44, no. 3, pp. 291–428, 2010.
- [4] F. Corman, “Interactions and equilibrium between rescheduling train traffic and routing passengers in microscopic delay management: a game theoretical study,” *Transportation Science*, vol. 54, no. 3, pp. 565–853, 2020.
- [5] X. Hong, L. Meng, A. D’Ariano et al., “Integrated optimization of capacitated train rescheduling and passenger reassignment under disruptions,” *Transportation Research Part C: Emerging Technologies*, vol. 125, p. 103025, 2021.
- [6] A. D’Ariano, F. Corman, T. Fujiyama, L. Meng, and P. Pellegrini, “Simulation and optimization for railway operations management,” *Journal of Advanced Transportation*, vol. 2018, Article ID 4896748, 3 pages, 2018.
- [7] W. Li, Q. Luo, Q. Cai, and X. Zhang, “Using smart card data trimmed by train schedule to analyze metro passenger route choice with synchronous clustering,” *Journal of Advanced Transportation*, vol. 201813 pages, 2018.
- [8] L. D’Acierno, M. Botte, M. Gallo, and B. Montella, “Defining reserve times for metro systems: an analytical approach,” *Journal of Advanced Transportation*, vol. 201815 pages, 2018.
- [9] B. Toader, A. Moawad, T. Hartmann, and F. Viti, “A data-driven scalable method for profiling and dynamic analysis of shared mobility solutions,” *Journal of Advanced Transportation*, vol. 202115 pages, 2021.
- [10] M. Botte, C. Di Salvo, A. Placido, B. Montella, and L. D’Acierno, “A neighbourhood search algorithm for determining optimal intervention strategies in the case of metro system failures,” *International Journal of Transport Development and Integration*, vol. 1, no. 1, pp. 63–73, 2017.
- [11] M. Samà, A. D’Ariano, F. Corman, and D. Pacciarelli, “A variable neighbourhood search for fast train scheduling and routing during disturbed railway traffic situations,” *Computers and Operations Research*, vol. 78, pp. 480–499, 2017.
- [12] V. Cacchiani and P. Toth, “Nominal and robust train timetabling problems,” *European Journal of Operational Research*, vol. 219, no. 3, pp. 727–737, 2012.
- [13] Y. Wang, Y. Gao, X. Yu, I. A. Hansen, and J. Miao, “Optimization models for high-speed train unit routing problems,” *Computers and Industrial Engineering*, vol. 127, pp. 1273–1281, 2019.
- [14] F. Corman, A. D’Ariano, A. D. Marra, D. Pacciarelli, and M. Samà, “Integrating train scheduling and delay management in real-time railway traffic control,” *Transportation Research Part E: Logistics and Transportation Review*, vol. 105, pp. 213–239, 2017.
- [15] L. Amorosi, P. Dell’Olmo, and G. L. Giacco, “Mathematical models for on-line train calendars generation,” *Computers and Operations Research*, vol. 102, pp. 1–9, 2019.
- [16] H. Bachraty, E. Krsak, and M. Tavec, “Algorithm for generating text descriptions of bit calendars,” *Communications*, vol. 11, no. 3, pp. 54–62, 2009.
- [17] N. Kumar and A. Mishra, “A multi-objective and dictionary-based checking for efficient rescheduling trains,” *Alexandria Engineering Journal*, vol. 60, no. 3, pp. 3233–3241, 2021.
- [18] N. Dethlefs, A. Schoene, and H. Cuayáhuil, “A divide-and-conquer approach to neural natural language generation from structured data,” *Neurocomputing*, vol. 433, pp. 300–309, 2021.
- [19] E. I. A. E. Lester, “Work breakdown structures,” *Project Management, Planning and Control*, pp. 53–59, Butterworth-Heinemann, Oxford, UK, 2017.

- [20] C. Wootton, "Compiled and interpreted languages," in *Developing Quality Metadata: Building Innovative Tools and Workflow Solutions*, pp. 299–308, CRC Press, Boca Raton, FL, USA, 2007.
- [21] I. G. M. Y. Bakti, T. Rakhmawati, S. Sumaedi, and S. Damayanti, "Railway commuter line passengers' perceived service quality: hedonic and utilitarian framework," *Transportation Research Procedia*, vol. 48, pp. 207–217, 2020.

Research Article

A Balanced Strategy for the FFBS Operator Integrating Dispatch Area, Route, and Depot Based on Multimodel Technologies

Qingfeng Zhou ^{1,2}, Jun Zhou ¹, and Chun Janice Wong ³

¹Shenzhen Urban Planning & Land Resource Research Center, Shenzhen 518055, Guangdong, China

²Shenzhen Key Laboratory of Urban Planning and Decision Making, Shenzhen 518055, Guangdong, China

³Harbin Institute of Technology, Shenzhen 518055, Guangdong, China

Correspondence should be addressed to Chun Janice Wong; janicewong@hit.edu.cn

Received 21 December 2020; Revised 26 April 2021; Accepted 10 May 2021; Published 20 May 2021

Academic Editor: Erfan Hassannayebi

Copyright © 2021 Qingfeng 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.

Bicycle scheduling is the essential strategy for balancing the demand for the public bicycle system (PBS). Existing literature pays more attention to bike scheduling models and their solutions, but seldom discusses the dispatch area and depot center. Reasonable dockless public bicycle dispatch area and optimal dockless bike dispatch depot location in the service area were discussed from the existing shared bicycle operation data in this paper. We proposed a feasible framework including bike trip network segmentation, mean-shift clustering based on the point position, VRP model, genetic algorithm, and TOPSIS evaluation method. The effectiveness and superiority of the division of the dispatch area are verified. The main evidence for this article is (1) although the cycling networks of bicycles are different at different times of the day, the results of community division are relatively stable and have great similarities. (2) The plan of the dispatch area has impacted on the operation efficiency of the PBS. For a scheduling area, the target value of the optimal scheduling strategy corresponding to different dispatch centers is obviously different. Therefore, the location of the dispatch center has a great impact on the quality of the scheduling strategy. The dispatch area determined by bike trip OD community detection has stable characteristics of scheduling costs. (3) This work is an attempt to combine big data and model technology to assist city management. We build a feasible framework to serve a balanced strategy for FFBS which can provide reasonable dispatch area, optimal dispatch depot location, dispatch truck's route length, load action, and time window. Our proposed framework provides new ideas for regional traffic dispatching for the traffic management department and FFBS operator, which has certain practical reference significance.

1. Introduction

Public bike system (PBS), also called a bicycle-sharing system (BSS), which was born in 1965 in Europe, has been developed for three generations [1]. The concept of the PBS/BSS is simple: a user arrives at a station, takes a bike, uses it for a while, and then returns it to another station. It is economical, eco-friendly, and healthy, has ultralow carbon emissions, is more equitable, has increasingly received attention in the last decade, and has rapidly emerged in many cities all over the world [2]. Since 2016, a relatively new model of the PBS, known as the free-float bike-sharing (FFBS) system, has increasingly gained its popularity. The FFBS is based on the mobile app and GPS which eliminates stations and docks (also called dockless bike). Passengers can

easily pick up and drop off the bike anywhere using their cell phone. This system is quite spread nowadays through enterprises as OFO and Mobike since early 2016 in China.

The FFBS is an innovative bike-sharing model. FFBS saves on start-up cost, in comparison to station-based bike sharing (SBBS), by avoiding the construction of expensive docking stations and kiosk machines. FFBS prevents bike theft and offers significant opportunities for smart management by tracking bikes in real time with built-in GPS. Despite the convenience and flexibility provided to users and their contribution to the sustainability of urban transportation, FFBS systems also face numerous challenges. For SBBS, the lack of resources is the major issue: a user can arrive at a station that has no bike available or wants to return her bike at a station with no empty spot. Like SBBS,

the success of FFBS depends on the efficiency of its rebalancing operations to serve the maximal demand as possible. It is not easy to maintain an effective balance in the distribution of bikes. The relationship between the supply and demand of free-float bikes is more complicated because of no restrictions on the use time and location. Owing to the fluctuating and asymmetric demand for rides throughout the day, the spatial distribution of bikes is highly imbalanced. Some scholars have studied the mobility patterns and imbalance characteristics of FFBS by analyzing its historical trip [3–5]. Their studies provide insights to assist the system operator to make more informed decisions. There are many research studies on public bicycle balance strategies. A widely adopted rebalancing tactic is the operator-based approach, characterized by a fleet of trucks and staff dedicated to manually transferring bikes across different regions. The strategy generally includes three contents: (1) determining the scope of the rebalance area; (2) determining the location of the dispatch center; and (3) finding the optimal dispatch strategy. Among them, the research on optimal dispatching strategies is the most concerned issue for scholars, and the scheduling objectives and solving algorithms have obtained rich results. However, a reasonable dispatching area is necessary for effective dispatching work. For those bicycle stations included in a dispatch area, if most stations in the area need to drop off bicycles, the bikes on the dispatch vehicle may be insufficient and need to return to the dispatch center for loading; if most stations in the area need to pick up bicycles, then the dispatching vehicle cannot quickly and effectively load these bicycles away. The reasonable dispatch area should maintain a balanced relationship between pickup and drop-off bikes.

In this study, we mainly paid more attention to the dispatching range and the location of the dispatching center for the FFBS. This paper is the first to comprehensively consider the scope of the rebalance area, dispatching route, and dispatching centers to provide a balanced strategy for FFBS system operators. The framework involves the network community structure, vehicle balancing problem, and multiobjective decision-making as shown in Figure 1. Two issues are discussed from the existing shared bicycle operation data: (1) how to determine a reasonable dockless public bicycle dispatch area? (2) How to find an optimal dockless bike dispatch depot location in the service area? In the framework, first, a network was established based on real bike trip OD data. Second, the social network theory was used to analyze the trip network structure by the community detection algorithm. The subgroups of the trip network can be obtained. The number of bicycles flowing between subgroup grids is relatively stable at a certain time. Therefore, the results of subgroup grids can be considered as a scope of the rebalance area. Third, for the rebalance area (grids of a subgroup), each grid is considered a potential dispatch center. We built a vehicle route problem model with two optimal objectives: the minimal vehicle cost and the maximum grid rebalance rate. The genetic algorithm mixed with Technique for Order Preference by Similarity to an Ideal Solution (TOPSIS) was used to find the optimal rebalanced strategy, and each potential dispatch depot's best rebalanced

strategy records the vehicle path, vehicle cost, and maximum grid rebalanced rate. Last, bringing together the best strategies of all potential dispatch centers in a rebalance area, TOPSIS was applied again to select the best dispatch center by considering vehicle costs and grid rebalancing rates. Our proposed framework provides new ideas for regional traffic dispatching for the traffic management department and FFBS operator, which has certain practical reference significance.

The rest part of this paper is organized as follows: Section 2 presents a literature review on community detection and rebalancing operations in bike-sharing systems. In Section 3, we described the community detection algorithm, the vehicle balance problem model, and the genetic algorithm used. In Section 4, we described the data of the OFO bicycle trip and the establishment of the OD network in Shenzhen. In Section 5, we analyzed community identification of the OFO bike trip network subgroup in Shenzhen and demonstrated the process of the framework to select the optimal dispatch center and its corresponding rebalanced strategy. Finally, Section 5 summarizes the results of this study and provides direction in future studies.

2. Literature Review

Sharing bike involved in many areas of research, and it is broadly based on two perspectives: user perspective and system perspective. In the remainder of this section, we mainly review the literature on community detection and rebalancing operations in the PBS.

2.1. Community in Networks. Newman and Girvan [6] gave widely accepted and used definitions: a community is a subgraph containing nodes which are more densely linked to each other than to the rest of the graph. A graph has a community structure if the number of links into any subgraph is higher than the number of links between those subgraphs. Very promising research on complex network theory attains the detection of communities [7]. A graph consists of edges and vertices such as $G = G(V, E)$. Community detection is the identification of $nc \geq 1$ communities in G such that the vertices of a community form an overlay of V . $C = \{C_1, C_2, \dots, G_{nc}\}$. If the intersection of the vertices of any two communities is empty, C is called a disjoint community; otherwise, it is called an overlapping community. Traditionally, community detection in graphs aims at identifying the modules only based on the topology. The problem has a long tradition, and it has appeared in various forms in several disciplines. New advances also propose the study of detection of communities in weighted networks where not only the topology influences the shaping of clusters but also the weight of each link. Many authors have proposed methods and algorithms to detect communities in networks. The literature indicates three main methods: divisive algorithms, optimization methods, and spectral methods.

Community detection in transportation network research is mostly used to discover urban activity structures.

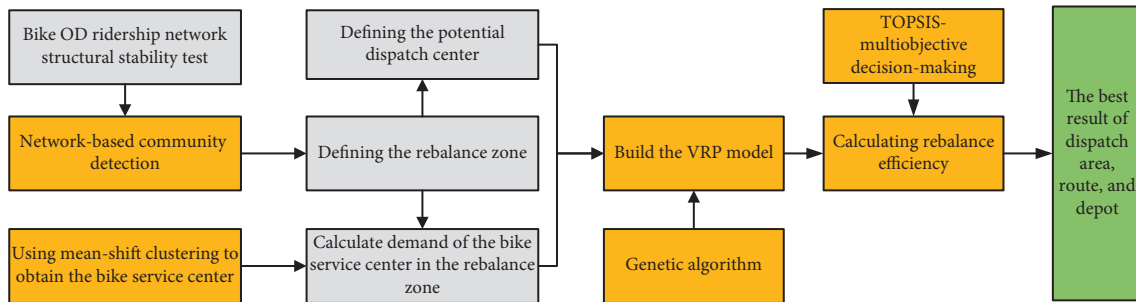


FIGURE 1: The workflow of this study.

De Montis et al. [8] adopted an algorithm based on the maximization of the weighted modularity of the commuter network to detect productive basins composed of municipalities. Du et al. [9] observed the dynamic mobility flows using community snapshots of different spatial stations over time using Shanghai as a case study. Zhang et al. [10] used community detection to evaluate the bus network topological structure. Austwick and Zaltz [11] employed community detection in the bicycle-sharing network to explore usage in cities. Borgnat et al. [12] discussed how Lyon's shared bicycle system, called Velo'v, is a dynamical complex network and how using community detection methods gives interesting results. Yao [13] constructed the public bicycle networks of different urban areas based on the real-time data of the Nanjing public bicycle system. Secondly, we analyzed and compared degree, strength, radiation distance, and community structure of the networks to understand the internal relations of the public bicycle system. In the next section, we illustrate the Louvain algorithm as proposed by Blondel et al. [14]. Therefore, we apply it to detect territorial clusters shaped by commuting in the center area of Shenzhen.

2.2. Bike-Sharing Rebalancing Strategies. Bike rebalancing strategies can be classified into user-based and operator-based. User-based approaches promote customers to select more appropriate origins/destinations to realize a more balanced system. Operator-based approach emphasizes the optimal dispatching of trucks to manually transferring bikes across different areas which is a widely adopted rebalancing tactic.

2.2.1. The User-Based Strategies. Fare discounts or pricing schemes are the most used means to motivate users to change destinations for improving the operational performance of the PBS [15, 16]. The optimized price vector determines the level of incentives that can persuade users to ride a bike from or stop at neighboring stations, thereby strategically reducing the number of unbalanced stations. Patel et al. [17] developed a discrete event simulation model of a real-world PBS to evaluate the effectiveness of incentives in rebalancing the system. Their job can consider profit and service levels to choose the best incentive plan. Reiss and Bosenberger [18] studied a user-based approach to a FFBS system and discussed the advantages and applicable

scenarios for both operator-based and user-based relocation strategies. Wu et al. [19] employed user-based tactics by incentivizing users to perform repositioning activities and constructed a more detailed quantitative model which can derive the optimal incentive scheme for the FFBS system. Some studies considered both user-based and operator-based strategies. Pfrommer et al. [20] used the model-based predictive control principle to examine the combination of dynamic truck routing and incentive scheme design in bicycle redistribution. Li and Shan [21] proposed two types of users in the BSS: leisure travelers and commuters. Operators and governments can adopt a bidirectional incentive model to improve their redistribution service levels. Ghosh and Varakantham [22] proposed a potentially self-sustaining and environment-friendly system of dynamic repositioning, which moves idle bikes during the day with the help of bike trailers. Their work can provide an optimization formulation that generates "repositioning" tasks.

2.2.2. The Operator-Based Strategies. The goal of the operator-based bike rebalancing problem is usually to find a minimum cost route for a vehicle and restore the inventory level at every bike station to its target value by picking up and delivering bicycles as necessary. The problems are classified into static bike rebalancing problem (SBRP) and dynamic bike rebalancing problem. The SBRP concerns the task of repositioning bikes among stations when traffic is low, and the PBS is idle [23]. The term static refers to the assumption that the number of bicycles at each station is known in advance and will not change during the pickup and transfer operations. The bike numbers cannot be adjusted in real time. Contrary to the static problem, for dynamic repositioning, the number of bicycles may change during the operations due to users renting and returning bicycles. Operator transfers and user trips occur simultaneously, so more real-time information must be considered in dynamic repositioning [24, 25]. There are various models and solution methods proposed to address the repositioning problem. Cruz [26] dealt with the SBRP of one single vehicle available, and the objective is to find a least-cost route that meets the demand of all stations and does not violate the vehicle capacity limits in the tour. An iterated local search-based heuristic was used to solve the problem. Based on the investigation of the net flow of each bike-sharing station in Jersey City, Zhang [27] proposed an integer linear

programming formulation to model bike-sharing static rebalancing. The formulation considers the problem introduced by the need to collect bicycles in need of repair, and a hybrid discrete particle swarm optimization algorithm is used to solve the model. Tang [28] proposed a bilevel programming model to formulate the SBRP, which can be used to determine the number of bikes loaded and unloaded at each station and the optimal truck routes in bike-sharing systems. Chemla et al. [29] presented efficient algorithms to solve the SBRP. Similar studies are available in the literature [30–32]. The multiple-vehicle balancing problem (MVBP) has the same objective that requires to design a set of routes and pickup and delivery operations along each route with multiple vehicles available. Casazza et al. [33] dealt with the MVBP and proposed an integer linear programming formulation obtaining proven optimal solutions for MVBP instances with up to 25 stations and an unbounded number of vehicles. Ho and Wai [34] proposed a hybrid large neighborhood search for solving the MVBP. Their heuristic is evaluated on three sets of instances with up to 518 stations and five vehicles. Júnior [35] presented an integer programming formulation, implemented under a branch-and-cut scheme, in addition to an iterated local search metaheuristic that employs efficient move evaluation procedures. Ghosh et al. [36] proposed an optimization formulation to reposition bikes using vehicles while also considering the routes for vehicles and future expected demand. They decomposed the problem (bike repositioning and vehicle routing) and aggregated bike stations to reduce the computation time significantly. Szeto and Shui [37] investigated the routes of the repositioning vehicles and the loading and unloading quantities at each bike station to firstly minimize the positive deviation from the tolerance of total demand dissatisfaction and then service time. This set of strategies is then embedded into an enhanced artificial bee colony algorithm to solve the BRP.

For the FFBS, the research on the bike balance problem is growing. Pal and Yu [38] presented a novel mixed-integer linear program for solving the SBRP in a series of studies of FFBS planning and management. The proposed formulation can not only handle single as well as multiple vehicles but also allows for multiple visits to a node by the same vehicle. They used a hybrid nested large neighborhood search with variable neighborhood descent algorithm, which is both effective and efficient in solving static complete rebalancing problems for large-scale bike-sharing programs. Aiming at the BRP scheduling with travel uncertainty, a multiobjective integer programming model was established based on the consideration of the static demand of fix time period, station capacity limit, penalty cost, and other practical factors by Zhang and Zhang [39]. An algorithm based on “ant colony algorithm” is then given to solve the model. Liu [40] studied the FFBS bike repositioning problem with multiple depots, multiple visits, and multiple heterogeneous vehicles. Easily and hardly access nodes with different penalties are defined to represent different convenience levels of getting bikes from the FFBS. The objective of the repositioning is to minimize the weighted sum of the inconvenience level of getting bikes from the system and the total unmet demand

and the total operational time. To solve this problem, an enhanced version of chemical reaction optimization is developed.

From the previous review, the research on the bike-sharing rebalancing problem can be distinguished from the objective of the balanced strategy and the algorithm for solving it. The dispatching area and dispatching center are rarely discussed. For the FFBS, since the dockless bicycle does not have a centralized station, the starting and ending position of the vehicle are only related to the user’s personal travel destination. This may result in different characteristics of bicycles using and influencing factors from the PBS. Our work complements previous research. This paper comprehensively considers the scope of the rebalance area, dispatching route, and dispatching centers to provide a balanced strategy for FFBS system operators.

3. Method and Data

3.1. Community Detection

3.1.1. The Modularity. In this paper, we mainly used the Louvain algorithm proposed by Blondel et al. [14] to identify the community structure of the bike trip network. The Louvain algorithm is an optimization method based on the maximization of an objective function called modularity [6], defined as follows for the case of weighted networks:

$$Q_w = \frac{1}{2W} \times \sum_{ij} \left(w_{ij} - \frac{s_i s_j}{2W} \right) \times \delta(c_i, c_j). \quad (1)$$

In equation (1), W_{ij} is the weight of the edge connecting node i and node j . $s_i = \sum_j w_{ij}$ (called node strength) is the sum of the weights of the edges attached to node i . $s_j = \sum_i w_{ij}$ is the sum of the weights of the edges attached to node j . $W = 1/2 \sum_{ij} w_{ij}$ is the sum of all the edge weights. $\delta(c_i, c_j)$ is a function. When vertices i and j belong to the same community, $\delta(c_i, c_j)$ is equal to 1; otherwise, its value is 0. The modularity is used to quantify how good is a community subdivision among all possible ones. For a particular subdivision, how many edges are there inside the communities with respect to the number of edges among them can be measured by computing modularity. Its values range from -1 to $+1$. The 0 value occurs when a certain subdivision has no more intracommunity edges that one would expect by random chance. A negative value means that there is no advantage in splitting the network in communities, and the best solution is one community.

3.1.2. The Louvain Algorithm. The Louvain algorithm allows one to approach two critical issues of optimization methods: detecting communities in large networks in a short time and considering the hierarchical community structure. The algorithm is based on an iterative process and can be used for both weighted and unweighted networks. For a network, the steps to community detection using the Louvain algorithm are as follows:

Step 1: each node is assigned to a unique single community.

Step 2: neighbor nodes of each target node are preferentially included in the same community if the variation of the modularity ΔQ_w is positive. The ΔQ_w function measures the level of performance of the partition associated to the displacement of a node from a community C to another.

Step 3: this aggregation process proceeds until the modularity function Q_w reaches a maximum.

Step 4: a new network is then built whose nodes correspond to the communities obtained in Step 3; each link connecting a pair of communities is featured by a weight equal to the sum of the weights of the external links originally between them. The internal links are represented by a self-loop, whose weight is equal to the sum of their internal weights.

Step 5: Step 1 applies to the last network.

3.1.3. The Adjusted Rand Index. In order to measure bike trip network community changes over time, this study introduces the Rand index. From a mathematical standpoint, Rand index is related to the accuracy, but is applicable even when class labels are not used. The Rand index is computed as the ratio of the number of pairs of objects having the same label relationship in two partitions. The Rand index has two shortcomings: it does not take a constant value for two random partitions and does not provide suitable results when the data comprise categories that overlap with each other to some degree. As the adjusted Rand index (ARI), Hubert and Arabie [41] extended the basic Rand index and provided a method able to handle two partitions R and Q of the same dataset. The adjusted Rand index is defined as the following equation:

$$\omega_a = a - \frac{(a+c)(a+b)/d}{(a+c) + (a+b)/2 - (a+c) + (a+b)/d} \quad (2)$$

In equation (2), a is the number of pairs of objects belonging to the same class in R and to the same cluster in Q . b is the number of pairs of data objects belonging to the same class in R and to different clusters in Q . c is the number of pairs of objects belonging to different classes in R and to the same cluster in Q . d is the number of pairs of objects belonging to different classes in R and to different clusters in Q . The adjusted Rand index gives the degree of agreement between two partitions of a dataset by a value bounded above by 1. A high adjusted Rand index indicates a high level of agreement, while a value of 1 suggests a perfect agreement. In the case of random partitions, the adjusted Rand index gives a value of 0.

3.2. VRP Model. Vehicle routing problem (VRP) is generally defined as follows: for a series of loading points and unloading points, organize appropriate driving routes to pass them in an orderly manner, while meeting certain constraints (such as the demand for goods, the amount of

delivery, and the time of delivery), vehicle load capacity, total route length, demand time window, etc., to achieve the goal of a certain problem (such as the shortest distance, the least cost, the least time, and the least number of vehicles). Because the demand of bicycles will change, this study establishes a multivehicle and multitime window VRP model based on considering the time demand and location demand of bicycles.

3.2.1. The Working Conditions and Boundaries of the VRP Model. We set some of our model working conditions and boundaries:

- (1) The bicycle demand of the service center is updated every hour. In a time, if the service center needs to be rebalanced, the demand of this bike service center is unchanged during the dispatch process.
- (2) For a truck, the start and end of its dispatch route are in the same depot.
- (3) If the service center needs to be rebalanced, it has one and only one dispatch truck to serve.
- (4) The service center's bike demand is complemented as much as possible by taking advantage of demand differences between service centers. Bicycle dispatch operation logic when the truck reaches grid i can be expressed by the following equation:

$$q_i = \begin{cases} \max(D_i, C_i - G_{\max}) & \text{when } D_i < 0, \\ \min(D_i, C_i) & \text{when } D_i > 0. \end{cases} \quad (3)$$

In equation (3), q_i is the drop/collect demand of bicycles in service center i . D_i represents the needs of service center i ; C_i is the bike number loaded by the truck when the truck arrives at service center i . $D_i > 0$ means need to drop bikes. If bicycles loaded by the truck are more than the demand, then the drop number of bikes is D_i ; otherwise, the dropped number is C_i . If $D_i < 0$, the truck needs to collect the bikes and transfer them to other service centers as much as the truck capacity allows. G_{\max} is the truck capacity of the bicycle.

- (5) When the dispatch truck departs from the depot, the initial number of shared bicycles loaded is determined according to the drop/collect number of bicycles of the first service center D_1 . If $D_1 < 0$, the initial number of shared bicycles loaded is 0. And if $D_1 > 0$, the initial number is $\min(D_1, G_{\max})$.

3.2.2. The Objectives of the VRP Model. The dispatch objectives in the VRP of this study mainly consider two aspects:

- (a) Minimal total route length of all dispatch trucks
- (b) Maximum demand satisfaction rate

The objective function is given in the following formula:

$$\begin{cases} \min S_1 = \sum_{k=1}^m \left(d_{0i} S_{ik} + \sum_{j=1}^n \sum_{i=1}^n d_{ij} x_{ijk} + d_{j0} E_{jk} \right) \\ \max S_2 = \sum_{i=1}^m \left(\frac{q_i}{|D_i|} \right) \end{cases} \quad (4)$$

S_1 is the route length of all dispatch trucks. k is the k th dispatch truck. $x_{ijk} = 1$ or 0 . x_{ijk} represents whether truck k is from service center i to service center j . Its value is 1 or 0. d_{ij} is the distance between service center i and center j . d_{0i} is the distance from the depot to service center i . S_{ik} represents whether service center i is the first destination of truck k . d_{j0} is the distance from the depot to service center j . E_{jk} represents whether service center j is the last destination of truck k . S_2 is the ratio of drop/collect bike number to drop/collect demand.

The input variables and output variables of the model are shown in Table 1.

3.2.3. The Algorithm of Solving the VRP. Genetic algorithm is one of the commonly used methods to solve the VRP. In this study, an elite strategy was introduced into the genetic algorithm to directly add the best individuals from the previous generation to the next generation of the population. The evolution direction of the evolution operator is one way, that is, it only accepts the evolution of the scheduling cost in the lower direction.

Step 1: generate the initial population. Let the number of iterations $z = 0$, use an integer arrangement to encode N grids, and use the coding sequence as the grid access order.

Step 2: randomly generate a series of demand points, and then add each demand point to the current dispatch route in turn. The individual coding sequence in the population corresponds to a determined scheduling route. The NP sequence individuals of length N are randomly generated to form the initial population.

Step 3: check whether the truck meets the conditions for returning to the depot center: ① the truck capacity reaches full load, and the next service center needs to collect bikes; ② the truck is empty, and the next service center needs to drop bikes; ③ the time when the truck arrives at the next center is not within the time window of the center that needs to serve.

If they are met, add the demand point to the next delivery route; if not, add it to the current delivery route as shown in Figure 2.

Step 4: calculate fitness values for all individuals in the z -generation population. Individuals to be evolved are selected through roulette. The greater the fitness, the greater the probability that individuals will be selected.

Step 5: let $z = z + 1$; increase the maximum fitness of the z -generation to the $z + 1$ -generation population. Determine whether it has reached the number of iterations

$z = \text{NG}$; if yes, output the current population, that is, the optimal population, and execute Step 6.

Step 6: find the scheduling cost S_1 and delivery rate S_2 of the routes corresponding to all individual sequences in the optimal population. According to the optional range of cost and delivery rate provided by the decision maker, a set of candidate routes U that satisfy the decision limit is screened. Establish a feature matrix based on the route set and corresponding target values, and use TOPSIS to determine the optimal route.

Step 7: TOPSIS (Technique for Order Performance by Similarity to an Ideal Solution) is based on simultaneous minimization of distance from an ideal point (IP) and maximization of distance from a nadir point (NP), and the optimal route is selected according to the length and demand satisfaction rate of each route by using TOPSIS. Here, we omit the evaluation process of TOPSIS and directly list the evaluation results of the strategy.

3.3. Data. A network was established based on real bike trip OD data. The analysis was conducted employing trip data for the OFO bicycle-sharing system. We scanned the working status of these bicycles every 15 minutes in one week of September 2017. There are about 57.6 million bicycle status records in a day. For a bicycle ID, we first judge whether the bicycle is used by comparing whether its position has changed. If changed, we saved the time and position of the bicycle. Then, according to the average travel speed and a travel distance of the bicycle, the abnormal bicycle use record is rejected. Figure 3 is a summary of the number of times the shared bicycles are used in 24 hours. Based on the data presented in Figure 3, we extracted the four periods of the day (morning: 07:00–10:00; noon: 11:00–14:00; afternoon: 15:00–18:00; evening: 19:00–22:00).

First, since there are no fixed stations for the dockless bike, we refer to the methodology for assessing the impact of floating traffic [42]. The service area of FFBS is divided into grids. Because 71.25% of bicycle trips have more than 500 meters of trip distance, 500 meters is selected as the scale for dividing the grid of Shenzhen. Our final dataset comprises 220,042 unique bicycle trips distributed over 619 grids in Shenzhen (see Figure 4). Second, the bike trips were sorted in time sequences including trip ID, pickup time and location, and drop-off time and location. The involved location is merged into the grid closest to it. Third, aggregated on a grid level, our data can be used to compile an origin-destination (OD) matrix. In the last step, a spatial network is constructed from the OD matrix, taking grids as nodes, trips

TABLE 1: The input variables and output variables of the VRP model.

The input		The output	
1	Road network for calculating route length and the time spent on the road	1	Optimal location of the depot
2	Speed of the dispatch truck	2	Number of trucks
3	Capacity of the truck	3	Number of bikes loaded/collected at each service center point
4	Operation time of dropping/collecting bikes at the service center	4	Each truck's route length
5	Locations of the potential depot	5	Total route length of all dispatch trucks
6	Locations of the service center and their demands	6	Time schedule of the dispatch truck
7	Time window of the service center needs to be rebalanced	7	Rebalance rate of all service centers

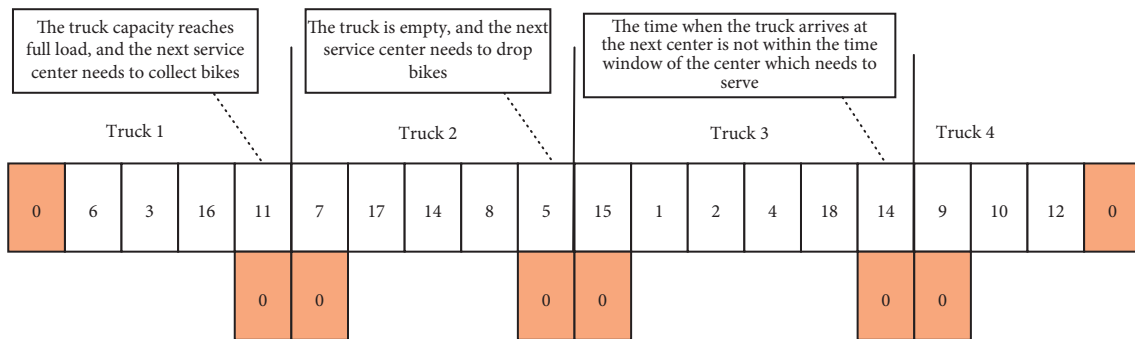


FIGURE 2: The conversion of demand point series into truck routes.

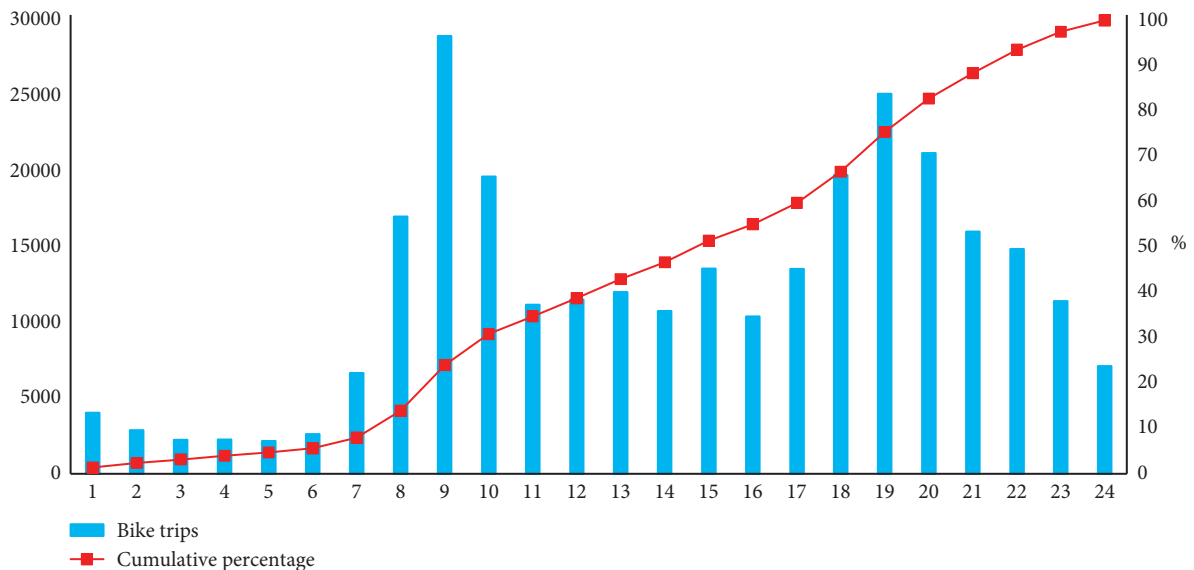


FIGURE 3: Floating bicycle riding trips' distribution across time within one day (24 hours).

between locations as edges, and the number of trips as the weight of edges. The nodes in the network are clustered according to the tightness of the connection, and the community division of the traffic network is obtained. We can then formulate a graph G with each grid describing a network node N , linked to every other grid in the network by a set of directed edges E weighted by a flow equal to the number of trips observed, given in the OD matrix.

4. Results and Discussion

4.1. *The Structure and Stability of Bike OD Network Communities.* In this section, we present an application of the community detection methodology described in Section 3 to support determining a reasonable dockless public bicycle dispatch area. Figure 5 shows the spatial distribution of community detection in four periods. In the figure, the same

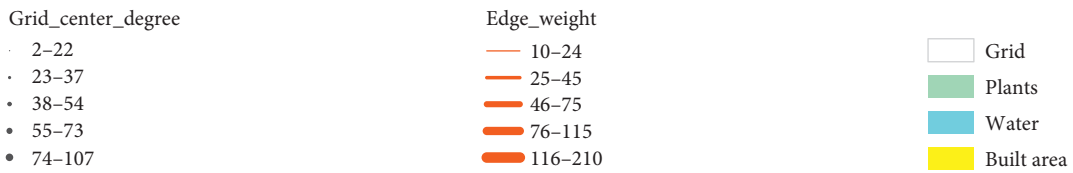
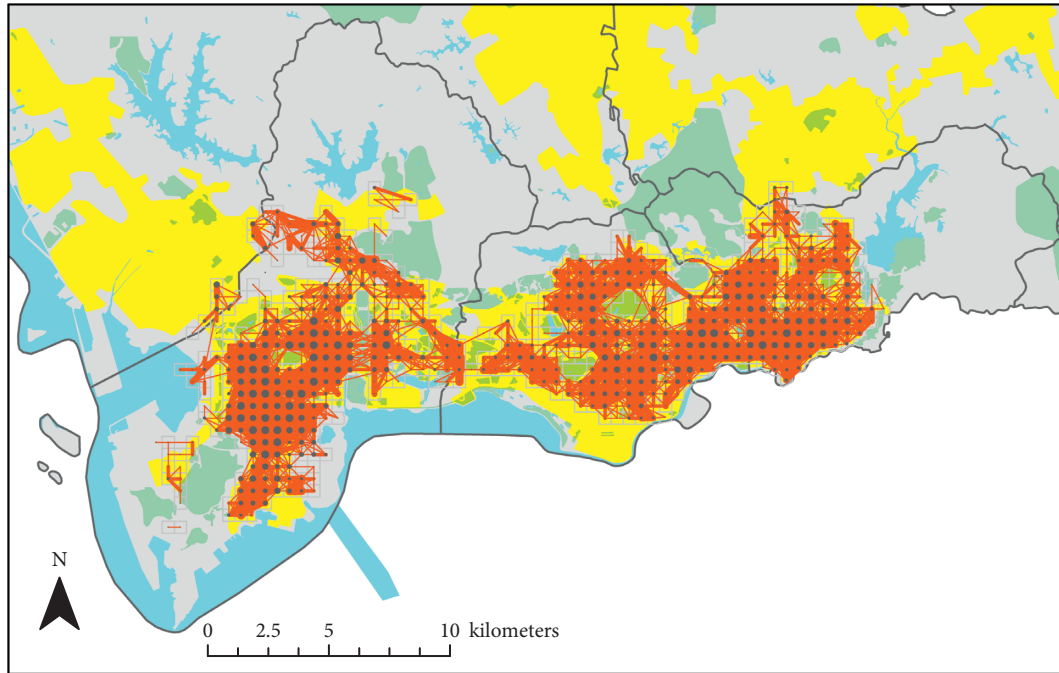


FIGURE 4: The OD network in Shenzhen.

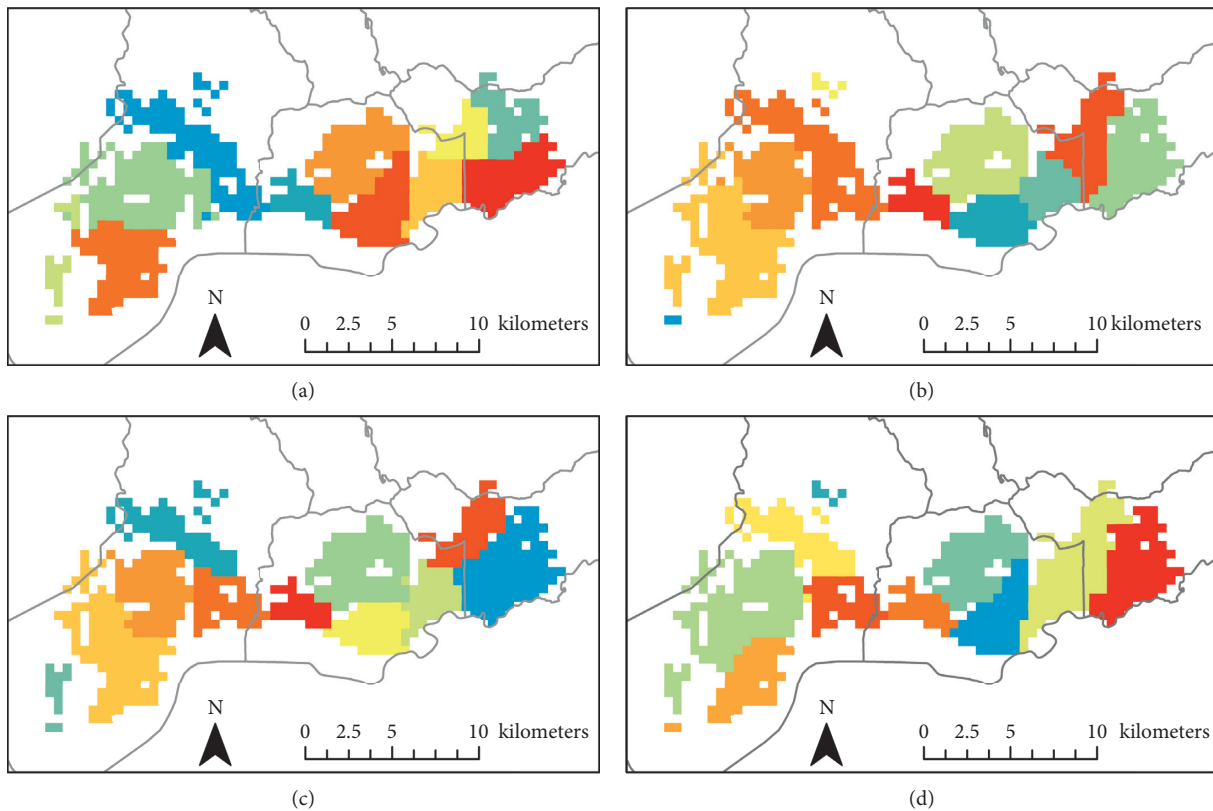


FIGURE 5: Spatial distribution of community detection in four periods. (a) Community detection in 07:00-10:00. (b) Community detection in 11:00-14:00. (c) Community detection in 15:00-18:00. (d) Community detection in 19:00-22:00.

color grid indicates that it belongs to the same community. In the four time periods, the boundaries between most communities are very clear and obvious. These obvious boundaries are mainly due to the division of high-level roads and other natural terrains. In addition, the grids of a community are also spatially adjacent to each other, and it is rare for a community to contain other community grids. This is because public bikes are mainly for short-distance travel, and there are few long-distance cross-regional cycling records. A few small grids that are spatially isolated from most grids will form a small community. In Table 2, we summarize and compare the number of communities for four periods. For the modularity, the community modularity of the four periods mostly exceeds 0.7 except for the evening period. This shows that the OD network community structure is good. In the morning and evening peak hours, the average degree of nodes (the number of pickup and drop-off in the grid) exceeds 100, and the other two periods are around 70. Judging from the number of communities, the number of communities in the four periods is very close, which is between 9 and 11.

In Table 3, the values of the adjusted Rand index for the case study are reported. The results in Table 4 show that the period 11:00–14:00 and the period 15:00–18:00 (values around 0.7951) show the best performance in the adjusted Rand index analysis. Partitions of the two periods have the highest similarities. The community similarity between evening peaks and afternoons is the lowest (values around 0.6073), which indicates that the difference of the bike spatial mobility pattern between evening peaks and afternoons is the largest during a day. In general, the communities in the four periods are compared in pairs, the ARI are all above 0.6, and the community structure is relatively stable. By the application of the Louvain method to the bike OD network of Shenzhen, we have shown that the mobility structure of the FFBS is obvious and stable. This high-quality and stable community division could suggest a bike dispatch area configuration of service area in a work day.

4.2. The Optimal Dispatch Depot and Route in a Dispatch Area

4.2.1. The Demo Dispatch Area. Based on the results in Section 5, a community of morning peak in Luohu was chosen as a dispatch area to analyze the optimal dispatch depot and route for the bike balance. The community contains a total of 28 grids; see Figure 5(a) for the location of the example community. Any two grids are connected, and the distance between the grids is the straight-line distance between their centroids. We used the mean-shift clustering method to identify the clustering area of bicycles based on the position of the bicycle by each hour in 07:00–10:00 [43]. When the bandwidth is 300 meters, a total of 78 bicycle gathering areas are obtained. Figure 6 shows the bicycle gathering area identified by mean-shift clustering. In Figure 6, each cluster has a center point, and the buffer analysis was proposed to obtain the range of the bicycle gathering area. 300 meters of buffers were established by ArcGIS based

TABLE 2: The number of communities for each period in the center Shenzhen bike network.

Time	Community number	Modularity	Average degree	Trips
07:00–10:00	11	0.717	102	63,665
11:00–14:00	11	0.764	63	39,097
15:00–18:00	11	0.744	80	49,830
19:00–22:00	10	0.698	108	67,450

on all cluster center points, thus to calculate the balance status of bikes in the bicycle gathering area.

We calculate the number of arrivals and departures of bicycles in each cluster and set a threshold. In a period, when the difference between arrival and departure times is more than 20, the bicycle gathering area is considered to need rebalancing service. By setting the threshold, there are 23 bicycle gathering areas that need to be rebalanced, called demand center, as shown in Figure 7. Different colors represent the time window requirements for rebalancing. The white number is the grid ID and represents a potential dispatch center. Table 4 lists the coordinates, requirements, and time of each demand center. In the drop/collect bike demand column, the negative sign indicates that the bicycle needs to be added, and the positive sign indicates that the bicycle needs to be transferred.

4.2.2. The Generation of the Optimal Strategy for the Dispatch Area. The initial variables set are as follows: the speed of the truck is 30 km/h, and it can load up to 100 bicycles. It takes 12–15 minutes for the demand center to collect bicycles. The algorithm of the VRP is implemented using MATLAB, and the algorithm iterations are 200. Every generation of population has 100 individuals, and everyone represents a dispatch route. The evolution rate and mutation rate are 0.8 and 0.05, respectively. The algorithm starts with taking grid ID 4182 in Figure 7 as a potential dispatch center to find the best dispatch route. The best dispatch route is selected by comparing objectives of the VRP model. Figure 8 is a process diagram of finding the optimal dispatching strategy when the dispatching depot is set on grid ID 4182. Figure 8(a) records the change in the minimum number of trucks in each generation. With the iteration of the algorithm, the vehicle number quickly decreased from 6 to 3. After 130 iterations, the minimum number of vehicles is stable at 3. Figure 8(b) records the change in minimum total route length in each generation. The convergence process is very fast, and finally, it is stable at about 33 km. The demand satisfaction rate in Figure 8(c) is also stable at about 99%. It is worth noting that dispatch routes corresponding to the three optimal objectives are not the same dispatch route, so TOPSIS is needed for choosing the best strategy. After TOPSIS comprehensive evaluation, when grid ID 4182 is used as the dispatch center, the best strategy result is 3 trucks and 35.8 km route length with 86.63% of demand satisfaction rate.

TABLE 3: The adjusted Rand index for different periods.

ARI	07:00–10:00	11:00–14:00	15:00–18:00	18:00–22:00
07:00–10:00	—	0.6659	0.6397	0.6129
11:00–14:00	0.6659	—	0.7951	0.6358
15:00–18:00	0.6397	0.7951	—	0.6073
18:00–22:00	0.6129	0.6358	0.6073	—

TABLE 4: The spatial coordinates and requirements of the demand center in the demo dispatch area.

Demand center	X	Y	Drop/collect demand	Time window
1	12701734.11	2578973.88	-32	06:00-07:00
2	12701795.72	2578366.42	-27	07:00-08:00
3	12701474.83	2579484.32	33	07:00-08:00
4	12701722.34	2578711.72	-25	07:00-08:00
5	12702581.54	2579159.77	73	07:00-08:00
6	12703056.19	2579551.61	-39	07:00-08:00
7	12702346.15	2578537.59	39	07:00-08:00
8	12702957.21	2578930.22	-37	07:00-08:00
9	12700627.12	2579357.52	-27	07:00-08:00
10	12702186.49	2578514.40	87	08:00-09:00
11	12701879.39	2578922.07	-63	08:00-09:00
12	12701553.92	2579136.82	32	08:00-09:00
13	12702440.59	2578885.01	130	08:00-09:00
14	12701450.30	2579466.74	-38	08:00-09:00
15	12701073.24	2578938.58	80	08:00-09:00
16	12702980.11	2579419.16	-74	08:00-09:00
17	12702978.05	2579848.00	-48	08:00-09:00
18	12702416.17	2579948.83	-40	08:00-09:00
19	12701963.87	2579839.53	-25	08:00-09:00
20	12702922.36	2578921.61	-56	08:00-09:00
21	12702473.52	2578904.86	59	09:00-10:00
22	12702413.21	2578535.34	43	09:00-10:00
23	12702971.81	2579505.02	-82	09:00-10:00

After the above calculation process for each potential dispatch center in Figure 7, finally, we obtained 28 records of the best strategy. It can be considered that there are 28 dispatch center options, and the optimal dispatch center is evaluated based on the truck number, route length, and demand satisfaction rate corresponding to the dispatch center. Here, the TOPSIS method is used again. Table 5 records the corresponding values of the top five dispatch centers and the last five dispatch centers. We can find that the best dispatch center is grid ID 913. The target value of the optimal scheduling strategy corresponding to different dispatch centers is obviously different. Therefore, the location of the dispatch center has a great impact on the quality of the scheduling strategy.

Figure 9 is the optimal scheduling route for demo dispatch area 0 represents the dispatch center which is grid ID 913, and other numbers represent demand centers. The optimal scheduling strategy requires three trucks. The routes of the truck and details of operation and time at the demand center are shown in Table 6. The departure time of the three trucks from the dispatch center and the number of bicycles loaded are not the same. The time window of the demand center is all satisfied, and the demand satisfaction rate of each demand center is counted.

4.3. The Validation of Optimal Dispatch Divisions. Luohu District was used to verify the effectiveness of using the trip network community detection to determine the dispatch area. In the morning peak hours, the bicycle trip network of Luohu is divided into four communities according to the best modularity as shown in Figure 10(a). So, we use manual grouping (Figures 10(b) and 10(c)) to generate four comparison groups. Grids with same color indicate that they belong to the same scheduling area. Manual grouping ensures that each dispatch area is about the same size with a clear boundary, and no dispatch area is contained by another scheduling area.

Table 7 shows the scheduling cost of different division groups of dispatch area in Luohu District. First, for demand satisfaction rate, the total demand satisfaction rate of three division methods was not much different with all exceeding 90%. The largest demand satisfaction rate is the result obtained by M2, which is close to 93%, but the gap with the average demand satisfaction rate of CD is only 2%. Second, for the VRP route length, the total dispatch route length of CD is the shortest, about 233.23 km. The total dispatch route length of M1 and M2 is about 250 km, at least 15 km more than CD. Third, for the dispatch truck number, CD needs 18 dispatch trucks,

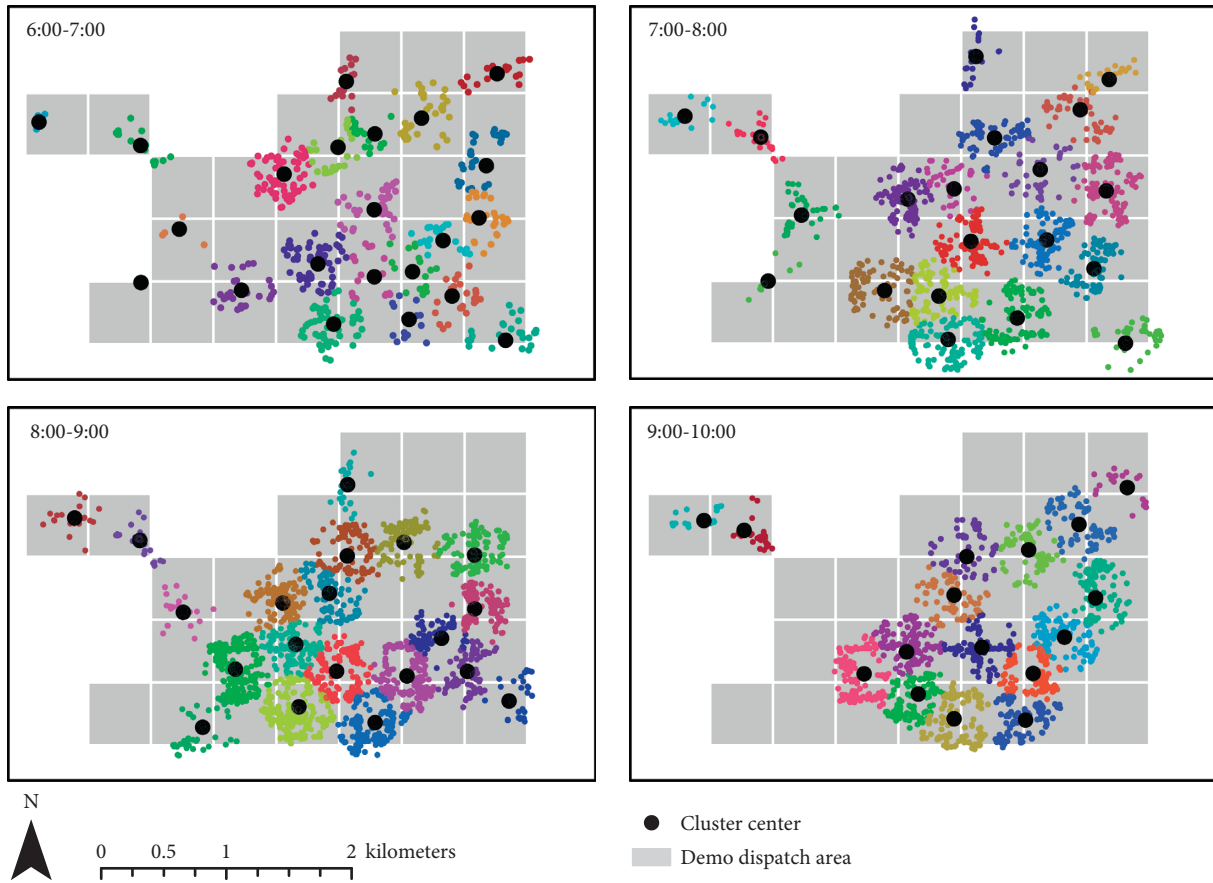


FIGURE 6: The bicycle gathering area identified by mean-shift clustering in four hours.

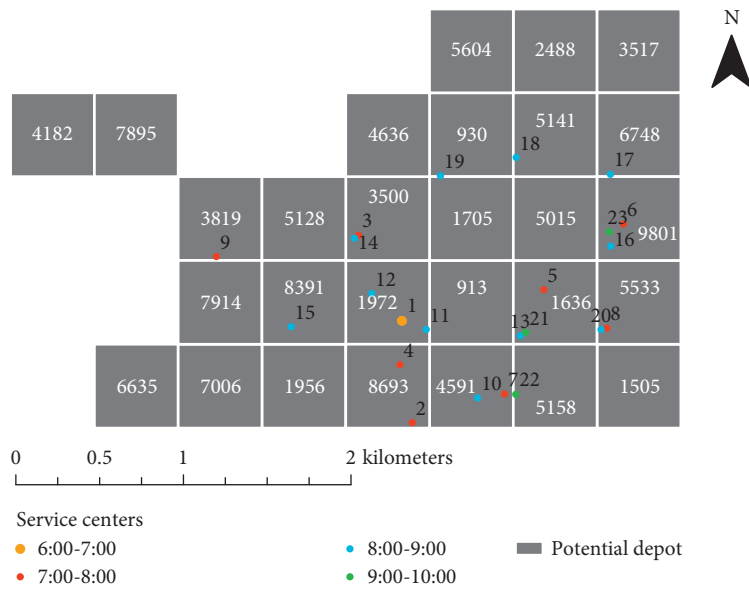


FIGURE 7: The demo dispatch area information in Futian.

while M1 and M2 need 19. In other words, in order to achieve the same demand satisfaction rate of the community detection group, manual group 1 and manual

group 2 need to pay a lot of cost in the dispatch distance and truck. By comparing the results of scheduling strategies in different situations, we found that the

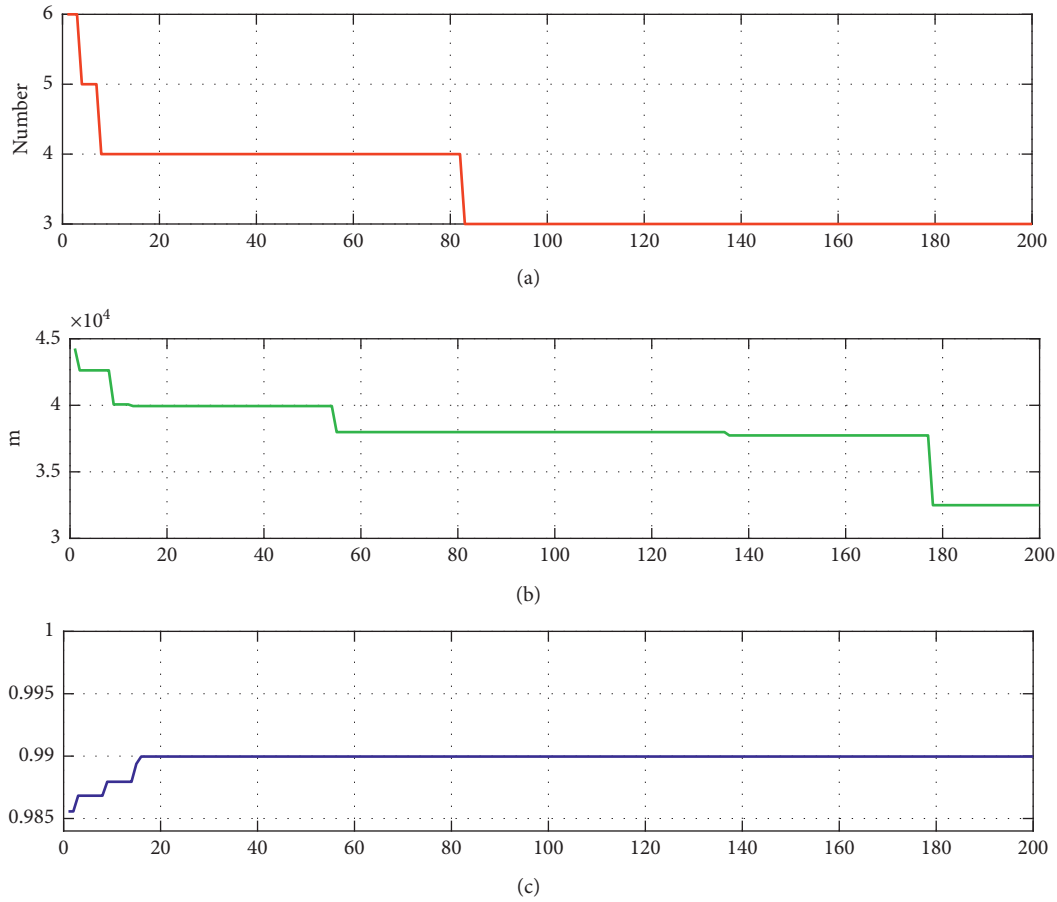


FIGURE 8: Process of seeking the optimal strategy. (a) Truck number. (b) Total length. (c) Demand satisfaction rate.

TABLE 5: The corresponding values of the top 5 and the last 5 dispatch centers.

Order	Depot grid ID	Truck number	Route length (km)	Demand satisfaction rate (%)	
Top 5	1	913	3	20.80	91.04
	2	5128	3	26.81	92.52
	3	4182	3	35.99	90.15
	4	1972	3	26.18	90.10
	5	9801	3	25.27	90.09
Last 5	5	3819	4	30.91	88.86
	4	4591	4	24.22	88.43
	3	5141	4	32.09	86.82
	2	1705	4	23.61	85.48
	1	4636	4	33.10	78.89

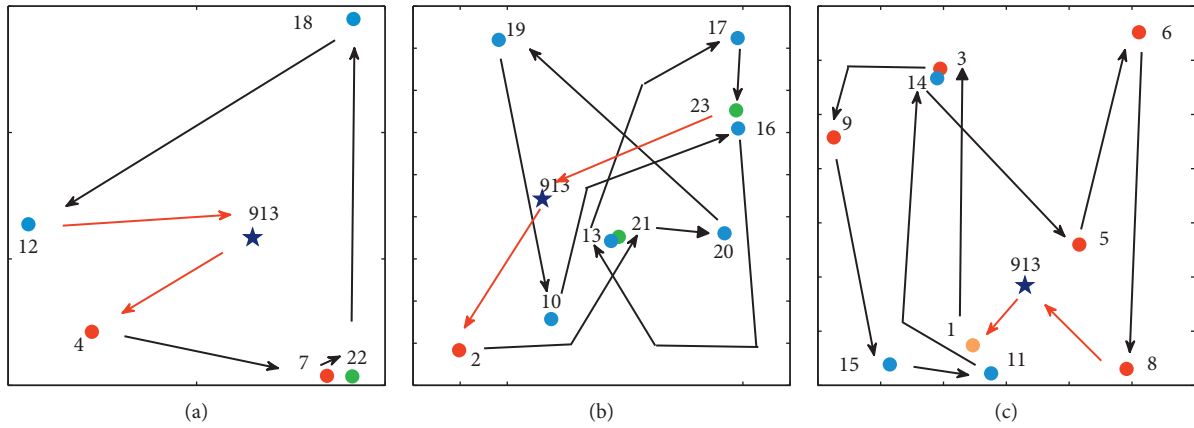


FIGURE 9: The optimal scheduling route of dispatch center 913. (a) Truck 1 route. (b) Truck 2 route. (c) Truck 3 route.

TABLE 6: The routes of the truck and details of operation and time at the demand center.

Truck	Route	Arrive/leave	Time window	Drop/collect bike number	Drop/collect demand	Demand satisfaction
1	0	—	—	25	—	—
	4	6:58/7:13	07:00-08:00	-25	-25	100%
	7	7:18/7:33	07:00-08:00	39	39	100%
	22	7:38/7:53	09:00-10:00	43	43	100%
	18	8:00/8:15	08:00-09:00	-40	-40	100%
12		8:20/8:35	08:00-09:00	32	32	100%
2	0	—	—	27	—	—
	2	6:58/7:12	07:00-08:00	-27	-27	100%
	21	7:13/7:25	09:00-10:00	59	59	100%
	20	7:26/7:38	08:00-09:00	-56	-56	100%
	19	7:40/7:52	08:00-09:00	-3	-27	11.10%
	10	7:55/8:09	08:00-09:00	87	87	100%
	16	8:10/8:22	08:00-09:00	-74	-74	100%
	13	8:24/8:36	08:00-09:00	87	130	66.92%
17	8:38/8:50	08:00-09:00	-48	-48	100%	
23	8:52/9:04	09:00-10:00	-52	-82	63.41%	
3	0	—	—	32	—	—
	1	5:59/6:11	06:00-07:00	-32	-32	100%
	3	6:12/6:24	07:00-08:00	33	33	100%
	9	6:26/6:38	07:00-08:00	-27	-27	100%
	15	6:39/6:51	08:00-09:00	80	80	100%
	11	6:53/7:05	08:00-09:00	-63	-63	100%
	14	7:06/7:18	08:00-09:00	-23	-38	60.52%
	5	7:21/7:33	07:00-08:00	73	73	100%
6	7:34/7:46	07:00-08:00	-39	-39	100%	
8	7:47/7:59	07:00-08:00	-34	-37	91.89%	
Total length		20.8 km		Demand satisfaction rate		91.04%

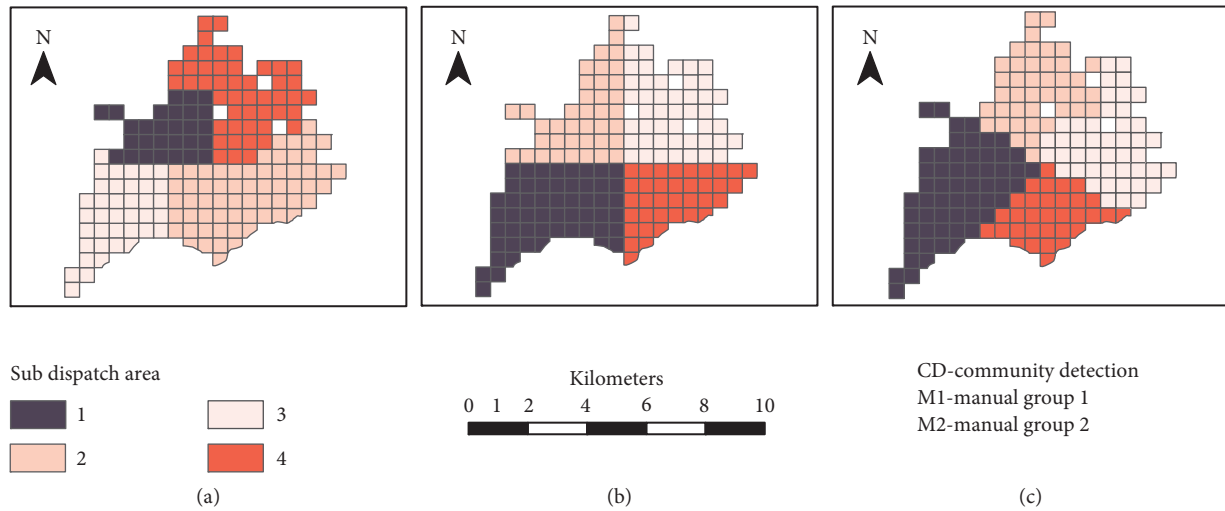


FIGURE 10: Three division groups of the dispatch area in Luohu District at morning peak.

TABLE 7: The scheduling cost of different division groups of the dispatch area in Luohu District.

Dispatch division	Subdispatch area	Depot grid	Truck number	Route length (km)	Demand satisfaction (%)
CD	1	913	3	20.80	91.04
	2	3128	7	109.92	89.80
	3	6221	6	64.01	86.37
	4	9764	2	38.50	93.99
	Total		18	233.23	90.3
M1	1	6122	9	119.85	87.27
	2	1972	4	48.71	85.91
	3	8237	2	30.21	96.77
	4	223	4	60.09	95.62
	Total		19	258.87	91.39
M2	1	5817	9	113.64	88.58
	2	930	4	57.00	92.77
	3	5307	2	28.75	95.93
	4	7356	4	49.81	93.56
	Total		19	249.20	92.71

CD: community detection; M1: manual group 1; M2: manual group 2.

dispatch area determined by bike trip OD community detection has stable characteristics of scheduling costs. In general, the scheduling strategy found through the trip OD community detection is optimal. This shows that the division of the scheduling area is a very important issue, which has a great impact on the dispatch cost. The method proposed in this paper can provide a reference for dividing the scheduling area.

5. Conclusions

This paper establishes a network based on the bike trip data and then uses a community discovery algorithm to segment the cycling network. The vehicle routing problem model with the shortest dispatch route and satisfaction rate dual goals is established and solved, and the effectiveness and superiority of the division of the dispatch area are verified. The main evidence for this article is as follows:

- (1) Many studies have shown that the bicycle riding has tidal characteristics, but our research found that the results of the community division of bicycle networks for different dispatches in a day are very similar, indicating that the flow of dockless shared bicycles is very stable and its range of activities has a clear boundary. From the perspective of the spatial characteristics of shared bicycle networks, the results of community division rarely show cross-regional phenomena. The interior of a community space will not contain another community, or a community will not be spatially separated by other communities. This provides the necessary basis for the division of bicycle dispatching areas.
- (2) The plan of the dispatch area has impacted on the operation efficiency of the PBS. Our research shows that, for a scheduling area, the target value of the optimal scheduling strategy corresponding to different dispatch centers is obviously different.

Therefore, the location of the dispatch center has a great impact on the quality of the scheduling strategy. By comparing the results of scheduling strategies in different situations, the dispatch area determined by bike trip OD community detection has stable characteristics of scheduling costs. In general, the scheduling strategy found through the trip OD community detection is optimal. The division of the scheduling area is a very important issue, which has a great impact on the dispatch cost.

- (3) This work is an attempt to combine big data and model technology to assist city management. We build a feasible framework to serve a balanced strategy for FFBS integrating dispatching area, route, and depot. This framework includes the construction of the bike trip network and subnet segmentation, mean-shift clustering based on the point position, VRP model, genetic algorithm, and TOPSIS evaluation method, which can provide reasonable dispatch area, optimal dispatch depot location, dispatch truck's route length, load action, and time window. Our work provides new ideas for regional traffic dispatching for the traffic management department and FFBS operator, which has certain practical reference significance.

Our study also has some limitations. First, the working day operational data used only contain one week, so the results of the analysis may be biased. The data we analyzed did not include data for nonworking days. Second, in the real world, there are many restrictions on decision-making. Although the VRP model we adopt was considering time, truck capacity, and driving speed, it cannot handle complex situations such as multiple dispatch centers, real-time road condition information, and more detailed dispatch operation time. Third, in our proposed framework, the result of subnet segmentation is crucial to the generation of scheduling strategies. The construction of cycling networks at different times will affect the division of communities, and further exploration is needed in the future.

Data Availability

The bike data were supplied by authors of this article. They are freely available. Requests for access to these data should be made to Qingfeng Zhou (zhouqingfeng@hit.edu.cn).

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] P. DeMaio, "Bike-sharing: history, impacts, models of provision, and future," *Journal of Public Transportation*, vol. 12, no. 4, p. 3, 2009.
- [2] S. A. Shaheen, S. Guzman, and H. Zhang, "Bikesharing in Europe, the americas, and asia," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2143, no. 1, pp. 159–167, 2010.
- [3] J. Dong, B. Chen, L. He et al., "A spatio-temporal flow model of urban dockless shared bikes based on points of interest clustering," *ISPRS International Journal of Geo-Information*, vol. 8, no. 8, p. 345, 2019.
- [4] X. Li, Y. Zhang, L. Sun, and Q. Liu, "Free-floating bike sharing in jiangsu: users' behaviors and influencing factors," *Energies*, vol. 11, no. 7, p. 1664, 2018.
- [5] Y. Xu, D. Chen, X. Zhang et al., "Unravel the landscape and pulses of cycling activities from a dockless bike-sharing system," *Computers, Environment and Urban Systems*, vol. 75, pp. 184–203, 2019.
- [6] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 99, no. 12, p. 782, 2002.
- [7] S. Fortunato, "Community detection in graphs," *Physics Reports*, vol. 486, pp. 75–174, 2009.
- [8] A. De Montis, S. Caschili, and A. Chessa, "Commuter networks and community detection: a method for planning sub regional areas," *The European Physical Journal Special Topics*, vol. 215, no. 1, pp. 75–91, 2013.
- [9] Z. Du, B. Yang, and J. Liu, "Understanding the spatial and temporal activity patterns of subway mobility flows," 2017, <https://arxiv.org/abs/1702.02456>.
- [10] H. Zhang, P. Zhao, Y. Wang, X. Yao, and C. Zhuge, "Evaluation of bus networks in China: from topology and transfer perspectives," *Discrete Dynamics in Nature & Society*, vol. 2015, Article ID 328320, 8 pages, 2015.
- [11] Austwick and M. Zaltz, "The structure of spatial networks and communities in bicycle sharing systems," *PLoS One*, vol. 8, Article ID e74685, 2013.
- [12] P. Borgnat, C. Robardet, P. Abry, P. Flandrin, JB. Rouquier, and N. Tremblay, "A dynamical network view of Lyon's vélo'v shared bicycle system," in *Dynamics on and of Complex Networks, Volume 2. Modeling and Simulation in Science, Engineering and Technology*, A. Mukherjee, M. Choudhury, F. Peruani, N. Ganguly, and B. Mitra, Eds., Birkhäuser, New York, NY, USA, 2013.
- [13] Y. Yao, "Analysis of network structure of urban bike-sharing system: a case study based on real-time data of a public bicycle system," *Sustainability*, vol. 11, no. 19, p. 5425, 2019.
- [14] V. D. Blondel, J. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 10, pp. 10008–10020, 2008.
- [15] A. Febbraro, N. Sacco, and M. Saeednia, "One-way carsharing: solving the relocation problem," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2319, pp. 113–120, 2012.
- [16] Z. Haider, "Inventory rebalancing through pricing in public bike sharing systems," *European Journal of Operational Research*, vol. 270, no. 1, pp. 103–117, 2018.
- [17] S. J. Patel, R. Qiu, and A. Negahban, "Incentive-based rebalancing of bike-sharing systems," in *Advances in Service Science*, pp. 21–30, Springer, Berlin, Germany, 2018.
- [18] S. Reiss and K. Bogenberger, "A relocation strategy for munich's bike sharing system: combining an operator-based and a user-based scheme," *Transportation Research Procedia*, vol. 22, pp. 105–114, 2017.
- [19] R. Wu, S. Liu, and Z. Shi, "Customer incentive rebalancing plan in free-float bike-sharing system with limited information," *Sustainability*, vol. 11, no. 11, p. 3088, 2019.
- [20] J. Pfrommer, J. Warrington, G. Schildbach, and M. Morari, "Dynamic vehicle redistribution and online price incentives in

- shared mobility systems,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 4, pp. 1567–1578, 2014.
- [21] L. Li and M. Shan, “Bidirectional incentive model for bicycle redistribution of a bicycle sharing system during rush hour,” *Sustainability*, vol. 8, no. 12, p. 1299, 2016.
- [22] S. Ghosh and P. Varakantham, “Incentivising the use of bike trailers for dynamic repositioning in bike sharing systems,” in *Proceedings of the International Conference on Automated Planning and Scheduling (ICAPS)*, Guangzhou, China, August 2017.
- [23] M. Benchimol, A. Benchimol, F. Chappert et al., “Balancing the stations of a self service “bike hire” system,” *RAIRO—Operations Research*, vol. 45, no. 1, pp. 37–61, 2011.
- [24] C. Contardo, C. Morency, and L.-M. Rousseau, “Balancing a dynamic public bikesharing system,” CIRRELT Technical Report, Centre Interuniversitaire de Recherche sur les Réseaux d’Entreprise, la Logistique et le Transport (CIRRELT), Montreal, Canada, 2012.
- [25] C. Kloim, P. Papazek, B. Hu, and R. Gunther, “Balancing bicycle sharing systems: an approach for the dynamic case,” in *Evolutionary Computation in Combinatorial Optimisation*, pp. 73–84, Springer, Berlin, Germany, 2014.
- [26] F. Cruz, “A heuristic algorithm for a single vehicle static bike sharing rebalancing problem,” *Computers & Operations Research*, vol. 79, pp. 19–33, 2016.
- [27] S. Zhang, “Bike-sharing static rebalancing by considering the collection of bicycles in need of repair,” *Journal of Advanced Transportation*, vol. 2018, Article ID 8086378, 18 pages, 2018.
- [28] Q. Tang, “A bilevel programming model and algorithm for the static bike repositioning problem,” *Journal of Advanced Transportation*, vol. 2019, Article ID 8641492, 19 pages, 2019.
- [29] D. Chemla, F. Meunier, and R. Wolfler Calvo, “Bike sharing systems: solving the static rebalancing problem,” *Discrete Optimization*, vol. 10, no. 2, pp. 120–146, 2013.
- [30] R. Nair and E. Miller-Hooks, “Fleet management for vehicle sharing operations,” *Transportation Science*, vol. 45, no. 4, pp. 524–540, 2011.
- [31] P. Angeloudis, J. Hu, and M. G. H. Bell, “A strategic repositioning algorithm for bicycle-sharing schemes,” *Transportmetrica A: Transport Science*, vol. 10, no. 8, pp. 759–774, 2014.
- [32] I. A. Forma, T. Raviv, and M. Tzur, “A 3-step math heuristic for the static repositioning problem in bike-sharing systems,” *Transportation Research Part B: Methodological*, vol. 71, pp. 230–247, 2015.
- [33] M. Casazza, A. Ceselli, D. Chemla, F. Meunier, and R. Wolfler Calvo, “The multiple vehicle balancing problem,” *Networks*, vol. 72, no. 3, pp. 337–357, 2018.
- [34] S. C. Ho and Y. S. Wai, “A hybrid large neighborhood search for the static multi-vehicle bike-repositioning problem,” *Transportation Research Part B: Methodological*, vol. 95, pp. 340–363, 2017.
- [35] T. L. Júnior, “The static bike relocation problem with multiple vehicles and visits,” *European Journal of Operational Research*, vol. 264, pp. 508–523, 2018.
- [36] S. Ghosh, P. Varakantham, Y. Adulyasak, and P. Jaillet, “Dynamic repositioning to reduce lost demand in bike sharing systems,” *Journal of Artificial Intelligence Research*, vol. 58, pp. 387–430, 2017.
- [37] W. Y. Szeto and C. Shui, “Exact loading and unloading strategies for the static multi-vehicle bike repositioning problem,” *Transportation Research Part B: Methodological*, vol. 109, pp. 176–211, 2018.
- [38] A. Pal and Z. Yu, “Free-floating bike sharing: solving real-life large-scale static rebalancing problems,” *Transportation Research Part C: Emerging Technologies*, vol. 80, pp. 92–116, 2017.
- [39] Z.-Y. Zhang and X. Zhang, “Shared bikes scheduling under users’ travel uncertainty,” *IEEE Access*, vol. 8, pp. 3123–3143, 2020.
- [40] Y. Liu, “A static free-floating bike repositioning problem with multiple heterogeneous vehicles, multiple depots, and multiple visits,” *Transportation Research Part C: Emerging Technologies*, vol. 92, pp. 208–242, 2018.
- [41] L. Hubert and P. Arabe, “Comparing partitions,” *Journal of Classification*, vol. 2, pp. 193–218, 1985.
- [42] B. Djsa and C. Xd, “Spatiotemporal evolution of ridesourcing markets under the new restriction policy: a case study in Shanghai,” *Transportation Research Part A: Policy and Practice*, vol. 130, pp. 227–239, 2019.
- [43] Q. Zhou, C. J. Wong, and X. Su, “Machine learning approach to quantity management for long-term sustainable development of dockless public bike: case of shenzhen in China,” *Journal of Advanced Transportation*, vol. 2020, no. 1, Article ID 8847752, 13 pages, 2020.

Research Article

Calibrating Path Choices and Train Capacities for Urban Rail Transit Simulation Models Using Smart Card and Train Movement Data

Baichuan Mo ¹, Zhenliang Ma ², Haris N. Koutsopoulos ³, and Jinhua Zhao ⁴

¹Department of Civil and Environmental Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139, USA

²Department of Civil Engineering, Monash University, Melbourne, VIC 3800, Australia

³Department of Civil and Environmental Engineering, Northeastern University, Boston, MA 02115, USA

⁴Department of Urban Studies and Planning, Massachusetts Institute of Technology, Cambridge, MA 02139, USA

Correspondence should be addressed to Zhenliang Ma; mike.ma@monash.edu

Received 10 January 2021; Revised 11 February 2021; Accepted 15 February 2021; Published 28 February 2021

Academic Editor: Erfan Hassannayebi

Copyright © 2021 Baichuan Mo 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.

Transit network simulation models are often used for performance and retrospective analysis of urban rail systems, taking advantage of the availability of extensive automated fare collection (AFC) and automated vehicle location (AVL) data. Important inputs to such models, in addition to origin-destination flows, include passenger path choices and train capacity. Train capacity, which has often been overlooked in the literature, is an important input that exhibits a lot of variabilities. The paper proposes a simulation-based optimization (SBO) framework to simultaneously calibrate path choices and train capacity for urban rail systems using AFC and AVL data. The calibration is formulated as an optimization problem with a black-box objective function. Seven algorithms from four branches of SBO solving methods are evaluated. The algorithms are evaluated using an experimental design that includes five scenarios, representing different degrees of path choice randomness and crowding sensitivity. Data from the Hong Kong Mass Transit Railway (MTR) system is used as a case study. The data is used to generate synthetic observations used as “ground truth.” The results show that the response surface methods (particularly constrained optimization using response surfaces) have consistently good performance under all scenarios. The proposed approach drives large-scale simulation applications for monitoring and planning.

1. Introduction

Urban rail systems are important components of the urban transportation system. Given their high reliability and large capacity, they have attracted high passenger demand. However, high demand also leads to problems such as overcrowding and disruptions, which decrease the level of service and impact passengers. To maintain service reliability and develop efficient response strategies, it is crucial for operators to better understand passenger demand and flow patterns in the network.

Transit network loading (or simulation) models for metro systems, powered by automated collected data, provide a useful instrument for network performance

monitoring. They enable operators to characterize the level of service and make decisions accordingly. A typical network loading model requires origin-destination (OD) matrix, supply information, and path choice fractions as input. The supply information includes the transit network topology, actual vehicle movement data, and vehicle capacity. Thanks to the wide deployment of automated fare collection (AFC) and automated vehicle location (AVL) systems, the OD demand and train movement data can be directly obtained. However, obtaining the corresponding path choices and quantifying reasonable vehicle capacity remains a challenge. According to Liu et al. [1] and Preston et al. [2], train capacity, defined as the maximum train load when remaining passengers in the platform denied boarding, may vary

depending on the crowding levels in trains and on platforms and passenger attitudes. The calibration of path choices and train capacity can improve the accuracy of network loading models for performance monitoring. Thus, these models can provide better information to operators to adjust operating strategies, relieve congestion, and improve efficiency.

Traditionally, path choices are inferred with data from on-site surveys that are used to estimate path choice models. However, surveys are time-consuming and labor-intensive, limiting their real-world usage. To overcome these disadvantages, path choice estimation methods based on AFC data have been proposed in the literature. AFC systems provide the exact locations and times of passengers' entry and exit transactions, which can be used to extract OD demand and passengers' journey times. They provide rich information for analyzing passenger behavior [3].

In an urban rail system operated near its capacity, five critical parameters are correlated with each other: OD demand, journey time, left behind (or denied boarding), path choices, and train capacity. The relationship of these parameters can be explained in Figure 1. OD demand is the input and journey time is the output (OD exit flow is a combination of the two), which can both be observed from the AFC data. Path choices, train capacity, and left behind are not observable in the AFC data. Journey time is directly affected by path choices and left behind (left behind can increase the waiting time). Left behind is directly affected by path choices and train capacity. This figure indicates the complexity of path choice estimation using AFC data. The dependencies of different parameters (e.g., path choices vs. train capacity) should be captured.

In the context of path choice estimation, the AFC data-based methods can be categorized into two groups: path-identification methods [4–7] and parameter-inference methods [8–12]. The former studies aim to identify the exact path chosen by each user and even the train they boarded. Path attributes are used to evaluate how likely a path is chosen for a passenger's trip from their observed origins to their observed destinations. The latter studies formulate probabilistic models to describe passengers' decision-making behavior. Bayesian inference is usually used to estimate the corresponding parameters and thus derive the path choice fractions. Despite using different methods, the key components for those AFC data-based studies are similar. They all attempt to match the model-derived journey times with the observed journey times from AFC data. However, many of these studies either assume a known constant train capacity or specify a known link-impedance function. As shown in Figure 1, journey times depend on both path choices and train capacity. An unreasonable setting of train capacity may cause calibration bias of path choices. Simultaneous calibration of both parameters is more reasonable.

Train capacity is a vague concept. Normally trains may not reach their designed physical capacity for various reasons (e.g., passengers may decide not to board due to the crowding Liu et al. [1]). Therefore, assuming a fixed physical capacity or fixed link-impedance function (in many previous studies) may not be a reasonable assumption in real-world

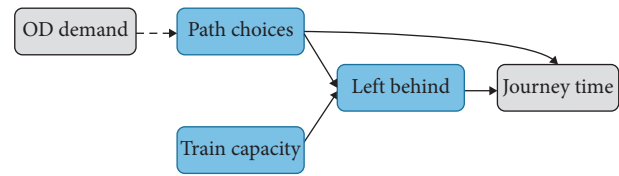


FIGURE 1: Relationship among critical parameters in urban rail systems.

situations, only a few studies have explored the calibration of actual train capacity in the rail system. Liu et al. [1] proposed the concept of “willingness to board” (WTB) to describe the varied capacity in a bus system and estimated passengers’ WTB using a least square method. Xu and Yong proposed a passenger boarding model which revealed that the number of actually boarding passengers in a crowded train was closely related to the number of queuing passengers and train load. Mo et al. [13] proposed an effective capacity model that recognized train capacity may vary across stations depending on the corresponding number of queuing passengers and train load. The calibration of train capacity or WTB usually requires the AFC data with passengers’ boarding and journey time information. However, this information may also be affected by path choices, which were neglected in previous studies.

To fill these research gaps, we propose a simulation-based optimization (SBO) model to calibrate path choices and train capacity simultaneously and also explore the efficiency of typical SBO solution algorithms. The calibration problem is formulated as an optimization problem using AFC and AVL data. The formulation can capture the interaction among these variables and their impact on journey times. Seven optimizers (solving algorithms) from four branches of SBO-solving methods are implemented for comparative analysis. They include generic algorithm (GA), simulated annealing (SA), Nelder–Mead simplex algorithm (NMSA), mesh adaptive direct search (MADS), simultaneous perturbation stochastic approximation (SPSA), Bayesian optimization (BYO), and constrained optimization using response surfaces (CORS). We compare these SBO solving algorithms within a limited computational budget, defined by the number of function evaluations. Data from the Hong Kong Mass Transit Railway (MTR) system provide the foundation for a realistic case study. The major contribution of this paper is twofold:

- (i) Proposing an optimization model to simultaneously estimate path choices and train capacities using AFC and AVL data, it addresses the typical assumption of fixed and known train capacities in existing path choice estimation studies using smart card data
- (ii) Validating the model using a busy urban rail network and analyzing the performance of SBO solution algorithms using systematic experiments, it represents different degrees of users’ randomness in path choice and their sensitivity to crowding

The remainder of the paper is organized as follows. In Section 2, we illustrate the SBO problem formulation.

Section 3 briefly describes the various SBO methods used in this study. The proposed framework is used in a case study with data from the Hong Kong MTR network in Section 4. The results are used to compare the performance of different algorithms. Section 5 concludes the paper by summarizing the main findings and discussing future research directions.

2. Methodology

The paper aims to calibrate simultaneously the train capacity and path choices using readily available data in the closed fare payment systems (require ticket validation at both tap-in and tap-out stations). To capture the interaction among different variables in Figure 1, we use a schedule-based network loading model with capacity constraints (described in Section 2.1). It outputs a list of performance metrics given a set of inputs including OD demand, timetables/AVL, network, train capacity, and path choices. The calibration of path choice and capacity is formulated as an optimization model that attempts to minimize the error between network loading model outputs (e.g., journey time, which is a function of path choices and train capacity) and the corresponding quantities directly observed from the AFC data.

2.1. Transit Network Loading Model. Transit network loading (TNL) models aim to assign passengers over a transit network given the (dynamic) OD entry demand and path choices. In this study, we adopt an event-driven schedule-based TNL model proposed by Mo et al. [13]. The model takes OD entry demand (number of tap-in passengers by time), path choices, train arrival and departure times from stations, train capacity, and infrastructure information (e.g., network topology) as inputs and outputs the passengers' tap-out times, train loads, waiting times, and other network performance indicators of interest.

Figure 2 illustrates the main functions of the TNL model [13]. Three objects are defined: train, waiting queue (on the platform), and passengers. An event is defined as a train arrival at, or departure from, a station. Events are ordered chronologically. New and transferring passengers join the waiting queue on the platform and board a train based on a first-come-first-board (FIFB) discipline. The number of successfully boarding passengers depends on the available train capacity.

The TNL model works by generating a train event list (arrivals and departures) based on the actual train movement data (AVL) and then sequentially processing the ordered events until all events are processed for the time period of interest. The processing of an individual event is based on the following rules:

- (i) If the event is an arrival (Figure 2(a)), the train offloads passengers and updates its state (e.g., train load and in-vehicle passengers). Alighting passengers who need to transfer are assigned to the waiting queues on the corresponding transfer platforms (e.g., passengers transferring to platform B in Figure 2(a)). Passengers who tap out will be removed from the system. New tap-in passengers who entered the

station between two events are added into the queue (e.g., new tap-in passengers in platform A in Figure 2(a)). Then, the waiting queue objects for all platforms are updated accordingly.

- (ii) If the event is a departure (Figure 2(b)), passengers board trains based on a FIFB priority rule. If the on-board passengers reached the train capacity, the remaining passengers at the platform will be denied boarding and wait for the next available train. Finally, the state of the train (train load and in-vehicle passengers) and the waiting queue at the platform are updated accordingly.

More specifically, for each passenger in the simulation model, we first calculate his/her probability of choosing each available path based on the path's attributes and path choice parameters (see Section 2.2, for details). Path attributes include in-vehicle time, number of transfers, and transfer walking time. Then, each passenger is assigned with a specific path based on the choice probability. Based on the path information, the passenger walks to a specific platform, joins the waiting queue, and waits for available trains to board. The boarding and alighting behavior are as described above.

2.2. Problem Formulation. Consider a general urban rail network in a specific time period T , represented as $G = (S, A)$, where S is the set of stations and A is the set of directed links. We divide T into several time intervals with equal length τ (e.g., $\tau = 15$ min). Denote the set of all time intervals as $\mathcal{T} = \{1, 2, \dots, T/\tau\}$. Define a *time-space (TS) node* as i_m , where $i \in S$ and $m \in \mathcal{T}$. i_m represents station i in time interval m .

For an OD pair (i, j) ($i, j \in S$), the *OD entry flow* ($q^{i_m, j}$) represents the number of passengers entering station i during time interval m and exiting at station j . Let the set of all OD entry flows be \mathbf{q}^e . The *OD exit flow* (q^{i, j_n}) represents the number of passengers who exit at station j in the time interval n with origin i . $q^{i_m, j}$ and q^{i, j_n} are inputs and outputs of the TNL model, respectively.

Let the set of all paths between (i, j) be $\mathcal{R}(i, j)$. We assume that the path choice behavior can be formulated as a C-logit model [14], which is an extension of the multinomial logit (MNL) model to correct the correlation among paths due to overlapping [15]. The path choice fraction for path $r \in \mathcal{R}(i, j)$ in time interval m ($p_r^{i_m, j}$) is formulated as follows:

$$p_r^{i_m, j} = \frac{e^{\mu(\beta_X \cdot X_{r,m} + \beta_{CF} \cdot CF_r)}}{\sum_{r' \in \mathcal{R}(i,j)} e^{\mu(\beta_X \cdot X_{r',m} + \beta_{CF} \cdot CF_{r'})}}, \quad \forall r \in \mathcal{R}(i, j), m \in \mathcal{T}, i, j \in S, \quad (1)$$

where μ is the scale parameter of the Gumbel distribution of the error term [16], which is usually normalized to 1. Larger (smaller) μ means the choice behavior is more deterministic (random). $X_{r,m}$ is the vector of attributes for path r in time interval m (e.g., in-vehicle time, number of transfers, and transfer walking time). CF_r is the commonality factor of path r which measures the degree of similarity of path r with

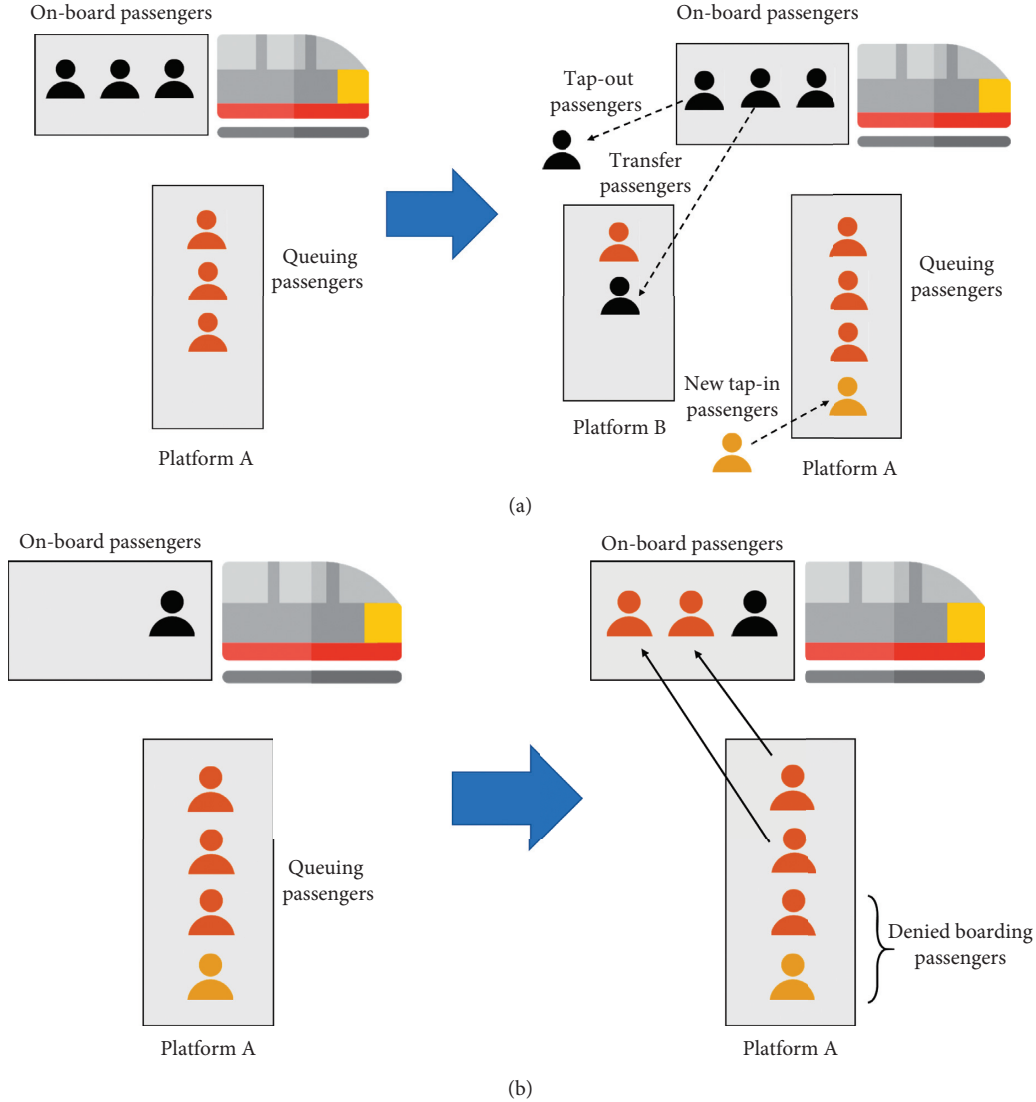


FIGURE 2: Main functions of the event-based transit network loading model. (a) Train arrival. (b) Train departure.

the other paths of the same OD. β_X and β_{CF} are the corresponding coefficients to be estimated. Let β be the vector that combines β_X and β_{CF} (i.e., $\beta = [\beta_X, \beta_{CF}]$).

CF_r is defined as follows:

$$CF_r = \ln \sum_{r' \in \mathcal{R}(i,j)} \left(\frac{D_{r,r'}}{D_r D_{r'}} \right)^\gamma, \quad (2)$$

where $D_{r,r'}$ is the number of common stations of path r and r' , D_r and $D_{r'}$ are the number of stations for path r and r' , respectively, and γ is a fixed positive parameter. Let the set of all path choice fractions be \mathbf{p} .

The values of β can be bounded from above and below. The boundaries can be obtained from the prior knowledge

and previous survey results. Denote the upper bound as U_β and lower bound as L_β ($L_\beta \leq \beta \leq U_\beta$), where U_β and L_β are both vectors with the same cardinality as β .

According to Mo et al. [13], the actual train capacity utilized by passengers is determined by three factors: (a) waiting passenger distribution on the platform, (b) train load and distribution across the train, and (c) passengers' willingness to board a crowded train. Thus, train capacity is not constant. Instead, it is dynamic and changes across stations and trains depending on the crowding level of the train and the platform. Mo et al. [13] model the capacity of train k at station i ($C_{k,i}$) is as

$$C_{k,i} = \begin{cases} \theta_0 n_i + \theta_1 H_{k,i} + \theta_2 Q_{k,i} & \text{if station } i \text{ is in the list of congested stations,} \\ \theta_0 n_i & \text{otherwise,} \end{cases} \quad \forall k, i, \quad (3)$$

where n_i is the number of cars of train i , $H_{k,i}$ is the load of train k when it arrives at station i , $Q_{k,i}$ is the number of passengers waiting on the platform when train k arrives at station i , and θ_0 , θ_1 , and θ_2 are parameters to be estimated ($\theta_0, \theta_1, \theta_2 > 0$). Specifically, θ_0 is a measure of service standard. $\theta_0 n_i$ can be seen as the base capacity, that is, the train load that represents acceptable service standards. At uncongested stations, passengers are assumed not to board when the train load is greater than $\theta_0 n_i$. At congested stations, passengers may still board a train even if it is already crowded [1], which makes the effective train capacity higher than $\theta_0 n_i$. θ_1 captures the effect that the effective capacity is higher when train load is higher. This is because passengers may worry if they did not board this crowded train, and they cannot board the following trains as well [1]. θ_2 captures the effect that more waiting passengers at the platform may push more passengers to board, leading to higher effective capacity.

In the discussion that follows, let θ be the vector of these three parameters. We assume that the values that these parameters can take is $L_\theta \leq \theta \leq U_\theta$, where L_θ and U_θ are the corresponding lower and upper bounds, respectively.

The goal is to calibrate θ and β vectors (used by the TNL model) based on indirect observations. Two sets of observations are used for the calibration: observed OD exit flows and observed journey time distribution (JTD). Both of them can be obtained from the AFC data.

Let the ground truth (observed) OD exit flow be \tilde{q}^{i,j_n} . Let $f_{i,j_t}(x)$ be the model-derived JTD of passengers with origin i who exit at station j during time interval t . Let $\tilde{f}_{i,j_t}(x)$ be the corresponding observed JTD extracted from the AFC data. Since $f_{i,j_t}(x)$ and $\tilde{f}_{i,j_t}(x)$ are estimated from passengers' journey time observations, only the OD pairs with more than E passengers exiting in a specific time interval are considered, where E is a predetermined threshold to ensure enough sample size. Denote the set of corresponding OD pairs and exit time intervals as \mathcal{E} , where $\mathcal{E} = \{(i, j_n): \tilde{q}^{i,j_n}, q^{i,j_n} > E, \forall i, j \in S, n \in \mathcal{T}\}$.

The calibration problem is formulated as an optimization problem:

$$\min_{\beta, \theta} w_1 \sum_{i,j \in S, m \in \mathcal{T}} (q^{i,j_n} - \tilde{q}^{i,j_n})^2 + w_2 \sum_{(i,j_n) \in \mathcal{E}} D_{\text{KL}} \left(\left\| f_{i,j_n} \tilde{f}_{i,j_n} \right\| \right), \quad (4a)$$

$$q^{i,j_n} = \text{TNL}(\mathbf{p}, \mathbf{q}^e, \theta) \quad \forall i, j \in S, m \in \mathcal{T}, \quad (4b)$$

$$f_{i,j_n}(x) = \text{TNL}(\mathbf{p}, \mathbf{q}^e, \theta_2) \quad \forall (i, j_n) \in \mathcal{E}, \quad (4c)$$

$$p_r^{i_m, j} = \frac{e^{\mu(\beta_X \cdot X_{r,m} + \beta_{CF} \cdot CF_r)}}{\sum_{r' \in \mathcal{R}(i,j)} e^{\mu(\beta_X \cdot X_{r',m} + \beta_{CF} \cdot CF_{r'})}} \quad \forall p_r^{i_m, j} \in \mathbf{p}, \quad (4d)$$

$$L_\beta \leq \beta \leq U_\beta, \quad (4e)$$

$$L_\theta \leq \theta \leq U_\theta. \quad (4f)$$

The objective function (equation (4)) has two parts: the square error between model-derived OD exit flows and the corresponding observations and the difference between model-derived and observed JTD. w_1 and w_2 are weights used to balance the scale and the importance of the two parts. The difference of the two distributions is expressed using Kullback–Leibler (KL) divergence (D_{KL}):

$$D_{\text{KL}} \left(\left\| f_{i,j_n} \tilde{f}_{i,j_n} \right\| \right) = \int_x f_{i,j_n}(x) \cdot \log \frac{f_{i,j_n}(x)}{\tilde{f}_{i,j_n}(x)} dx. \quad (5)$$

TNL ($\mathbf{p}, \mathbf{q}^e, \theta$) is the black-box function that corresponds to the TNL model, which can output the model-derived OD exit flows and JTD for a given set of path choices and train capacity. Since the TNL model has no analytic form, equation (4) is a SBO problem with upper and lower bound constraints. In the following section, we discuss seven different algorithms appropriate for the solution of SBO problems. These algorithms belong to four general approaches of SBO solving methods.

It is worth noting that $X_{r,m}$ (i.e., the path attributes vector) is known and fixed in this study. It is assumed to represent the historical path conditions based on which passengers make their habitual choices. Different from typical transit/traffic assignment problems where path choices are estimated by assuming user equilibrium (for planning purposes), the AFC data-based estimation aims to find the actual realized path choices based on real-world observations (i.e., OD entry-exit flows). Since passengers make decisions before knowing the actual travel or waiting times, $X_{r,m}$ should reflect passengers' historical perceptions of path attributes and should not change within the model estimation process. Therefore, though $C_{k,i}$ captures the actual path crowding information, it should *not* be included in the path choice formulation as passengers make decisions before knowing the actual crowding.

3. Simulation-Based Optimization Algorithms

There are four major classes of methods for solving the SBO problems, including the heuristic methods, direct search methods, gradient-based methods, and response surface methods (Osorio and Bierlaire [17]; Amaran et al. [18]). Heuristic methods are partial search algorithms that may provide a sufficiently good solution to an optimization problem, especially with incomplete or imperfect information or limited computation capacity. Direct search methods are derivative-free methods that are based on the sequential examination of trial points generated by a certain strategy. They are attractive as they are easy to describe and implement. More importantly, they are suitable for objective functions where gradients do not exist everywhere. Gradient-based approaches (or stochastic approximation methods) attempt to optimize the objective function using estimated gradient information. These methods aim to imitate the steepest descent methods in derivative-based optimization. Finite difference schemes can be used to estimate gradients but they may involve a large number of

expensive function evaluations if the number of decision variables is large. Response surface methods are useful in the context of continuous optimization problems. They focus on learning input-output relationships to approximate the underlying simulation by a predefined functional form (also known as a metamodel or surrogate model). This functional form can then be used for optimization leveraging powerful derivative-based optimization techniques.

In this study, we use seven representative algorithms belonging to these four classes of SBO methods to address the aforementioned path choice and train capacity calibration problem. Table 1 summarizes the main characteristic of these algorithms. The summary of all algorithms is described in Table 1.

In the discussion that follows, let Θ be the combined vector of β and θ (i.e., $\Theta = [\beta, \theta]$ is the vector of all coefficients to be estimated). Let N be the dimension of Θ (i.e., $\Theta \in \mathbb{R}^N$).

3.1. Genetic Algorithm (GA). GA is a heuristic method for solving both constrained and unconstrained optimization problems, which belongs to the larger class of evolutionary algorithms inspired by natural selection, the process that drives biological evolution. The GA repeatedly modifies a population of individual solutions as an evolution process [26]. The GA can be used to solve a variety of optimization problems that are not well suited for standard optimization algorithms, such as the SBO problem where the objective function (or constraints) is nondifferentiable and highly nonlinear.

The evolution starts from a population of randomly generated individuals and is an iterative process, with the population in each iteration called a generation. In each generation, the genetic algorithm selects individuals at random from the current population to be parents and uses them to produce the children for the next generation. Over successive generations, the population “evolves” toward an optimal solution. The genetic algorithm uses three main procedures at each step to create the next generation from the current population. (1) Selection: select the individuals, called parents, who contribute to the population of the next generation. Individuals with better objective function values are more likely to be selected. (2) Crossover: combine two parents to form children for the next generation. (3) Mutation: apply random changes to individual parents to form children.

In this study, we adopted a blend crossover and Gaussian mutation methods. The probability of crossover is set as 0.8 and the probability of mutating is set as 0.4. And, the population size is set as 6 given the limited computational budget. The algorithm is implemented by the Python `deap` package [19].

3.2. Simulated Annealing (SA). SA is a heuristic method for solving optimization problems [27]. The method is based on the physical process of heating a material and then slowly lowering the temperature to decrease defects, thus minimizing the system energy.

At each iteration of the SA algorithm, a new point is randomly generated. The distance of the new point from the current point, or the extent of the search, is based on a probability distribution with a scale proportional to the temperature. A distorted Cauchy–Lorentz visiting distribution is used in this study [20]. The algorithm accepts not only all new points that lower the objective function but also, with a certain probability, points that raise the objective function. By accepting points that raise the objective function, the algorithm avoids being trapped in local minima. An annealing schedule is selected to systematically decrease the temperature as the algorithm proceeds. As the temperature decreases, the algorithm reduces the extent of its search to converge to a minimum.

In this study, the SA algorithm in Python `Scipy` package is adopted for the implementation with all model parameters set as default [28].

3.3. Nelder–Mead Simplex Algorithm (NMSA). NMSA is a simplex method for finding a local minimum [29]. NMSA in N dimensions maintains a set of $N + 1$ test points arranged as a *simplex*. Denote the initial value of Θ as Θ^{ini} . The initial simplex set ($N + 1$ points) is generated as $\{\{\Theta: \Theta = \Theta^{\text{ini}} + e_i, \forall i = 1, \dots, N\}\} \cup \{\Theta^{\text{ini}}\}$, where $e_i \in \mathbb{R}^N$ is the unit vector in the i th coordinate and σ is the step size which is set as 0.05 in this study [21].

Based on the initial simplex, the model evaluates the objective function for each test point, in order to find a new test point to replace one of the old test points. The new candidate can be generated through simplex centroid reflections, contractions, or other means depending on the function value of the test points. The process will generate a sequence of simplexes, for which the function values at the vertices get smaller and smaller. The size of the simplex is reduced, and finally, the coordinates of the minimum point are found.

Four possible operations, reflection, expansion, contraction, and shrink, are associated with the corresponding scalar parameters: α_1 (reflection), α_2 (expansion), α_3 (contraction), and α_4 (shrink). In this study, we set the value of these parameters as $\{\alpha_1, \alpha_2, \alpha_3, \alpha_4\} = \{1, 2, 0.5, 0.5\}$ as suggested in [21]. The algorithm is implemented by the Python `scikit-learn` package with all parameters set as default. Since NMSA is designed for unconstrained problem, we turned the bound of Θ into a big penalized term in the objective function for this algorithm. For more details regarding the NMSA, one can refer to [21].

3.4. Mesh Adaptive Direct Search (MADS). The MADS algorithm is a direct search framework for nonlinear optimization [30]. It seeks to improve the current solution by testing points in the neighborhood of the current point (the incumbent). The neighborhood points are generated by moving one step in each direction from the incumbent on an iteration-dependent mesh. Each iteration of MADS consists of a SEARCH stage and an optional POLL stage. The SEARCH stage evaluates a finite number of points proposed by the searching strategy (e.g., moving one step around from

TABLE 1: Algorithms' summary.

Type	Algorithm	Constraints	Stochastic	Source
Heuristic method	Genetic algorithm (GA)	Yes	Yes	Fortin et al. [19]
	Simulated annealing (SA)	Yes	Yes	Tsallis and Stariolo [20]
Direct search	Nelder–Mead simplex algorithm (NMSA)	No	No	Gao and Han [21]
	Mesh adaptive direct search (MADS)	Yes	Yes	Abramson et al. [22]
Gradient-based	Simultaneous perturbation	Yes	Yes	Spall et al. [23]
	Stochastic approximation (SPSA)	Yes	Yes	Snoek et al. [24]
Response surface	Bayesian optimization (BYO)	Yes	Yes	Snoek et al. [24]
	Constrained optimization using Response surfaces (CORS)	Yes	Yes	Regis and Shoemaker [25]

the current point). Whenever the SEARCH step fails to generate an improved mesh point, the POLL step is invoked. The POLL step conducts local exploration near the current incumbent, which also intends to find an improved point on the mesh. Once an improved point is found, the algorithm updates the current point and constructs a new mesh. According to [30], the mesh size parameters approach zero as the number of iteration approaches infinity, which demonstrates the convergence of the MADS algorithm.

In this paper, we use a variant of the MADS method called ORTHO-MADS, which leverages a special orthogonal positive spanning set of polling directions. More details regarding the algorithm can be found in [22]. NOMAD 3.9.1 [31] with the Python interface is used for the MADS algorithm application. The hyperparameters are tuned based on the NOMAD user guide. The direction type is set as orthogonal, with $N + 1$ directions generated at each poll. Latin hypercube search is not applied.

3.5. Simultaneous Perturbation Stochastic Approximation (SPSA). SPSA is a descent direction method for finding local minimums. It approximates the gradient with only two measurements of the objective function, regardless of the dimension of the optimization problem. Denote the objective function in equation (4) as $Z(\Theta)$. The estimated parameters in the k th iteration is denoted as $\Theta^{(k)}$. Then, one iteration for the SPSA is performed as

$$\Theta^{(k+1)} = \Theta^{(k)} - a_k \cdot \widehat{\nabla} Z(\Theta^{(k)}), \quad (6)$$

where

$$\widehat{\nabla} Z(\Theta^{(k)}) = \frac{Z(\Theta^{(k)} + c_k \Delta_k) - Z(\Theta^{(k)} - c_k \Delta_k)}{2c_k \Delta_k}, \quad (7)$$

$$a_k = \frac{a}{(k + 1 + A)^\alpha}, \quad (8)$$

$$c_k = \frac{c}{(k + 1)^\gamma}, \quad (9)$$

where Δ_k is a random perturbation vector, whose elements are obtained from a Bernoulli distribution with the probability parameter equal to 0.5. $\{\alpha, \gamma, a, c, A\}$ are tuned as $\{0.602, 0.101, 0.001, 0.007, 0.1M\}$ in this study according to the numerical tests and guidelines from prior empirical studies of Gomez-Dans [32]. M is the maximum number of iterations.

3.6. Bayesian Optimization (BYO). BYO constructs a probabilistic model of the objective function and exploits this model to determine where to evaluate the objective function for the next step. The philosophy of BYO is to use all of the information available from previous evaluations, instead of simply relying on the local gradient and Hessian approximations. This enables BYO to find the minimum of difficult nonconvex functions with relatively few function evaluations.

BYO assumes a prior distribution for the objective function values and uses an acquisition function to determine the next point to evaluate. In this study, we use the Gaussian process as the prior distribution for the objective function due to its flexibility and tractability. For the acquisition function, we tested three common criteria: probability of improvement (POI), expected improvement (EI), and upper confidence bound (UCB) [24]. The EI criterion is used in this path choice calibration problem due to its best performance in our problem. The BYO is implemented in Python with the bayes_opt package. More details regarding the BYO can be found in [24].

3.7. Constrained Optimization Using Response Surfaces (CORS). CORS is a response surface method for global optimization. In each iteration, it updates the response surface model based on all previously probed points and selects the next point to evaluate. The principles for the next point selection are (a) finding new points that have lower objective function value and (b) improving the fitting of the response surface model by sampling feasible regions where little information exists. Hence, the next point is selected by solving the minimization problem of the current response surface function subject to constraints that the next point should be more than a certain distance away from all previous points [25].

An algorithm following the CORS framework requires two components: (a) a scheme for selecting an initial set of points for objective function evaluation and (b) a procedure for globally approximating the objective function (i.e., a response surface model). In this study, the initial sampling is conducted using the Latin hypercube methods, with the initial sampling number equal to $0.2 \times$ the total number of function evaluations allowed. The radial basis function (RBS) is used as the response surface model. For the subsequent sampling, a modified version of the CORS algorithm with space re-scaling is used. Details about the algorithm can be found in [25, 33].

4. Case Study

The proposed modeling framework is tested using data from the Hong Kong MTR network. MTR is the operator of the Hong Kong urban rail network, which provides services for the urbanized areas of Hong Kong Island, Kowloon, and the New Territories. The system currently consists of 11 lines with 218.2 km (135.6 miles) of rail, serving 159 stations including 91 heavy rail stations and 68 light rail stops. It serves over 5 million trips on an average weekday. Most of the passengers use a smart card fare payment system named Octopus. For the urban heavy rail lines, trip transactions are recorded when passengers enter and exit the system, providing information about the tap-in and tap-out stations and corresponding timestamps.

4.1. Experimental Design. We use AFC data on a typical weekday afternoon peak period (18:00–19:00) in March 2017 for the model application. Li [34] conducted a revealed-preference (RP) path choice survey of more than 20,000 passengers in the MTR system and used them to estimate a path choice model. The estimation results are shown in A. The following attributes were used in the specification of the model: (a) total in-vehicle time, (b) the number of transfer times, (c) relative walking time (total walking time divided by total path distance), and (d) the commonality factor (equation (2)). Future research may consider more path choice attributes such as perceived crowding levels and estimated waiting times.

As the real-world path choice information and train capacity are usually unavailable, we validate the models with synthetic data. To generate the synthetic data, we first extract the OD entry flow ($q^{i_m, j}$) from the real-world AFC records. We assume a synthetic Θ as the “true” path choice and train capacity parameters. The TNL model with the true OD entry flow, train timetable, and synthetic Θ as inputs is used to simulate the travel of passengers in the system and record people’s tap-in and tap-out time. The input timetable is treated as the *synthetic AVL data*. The resulting passengers’ tap-in and tap-out times are treated as the *synthetic AFC data*. The synthetic data, including “true” passenger path choices and train capacity, are used to evaluate the performance of the model under the various solution algorithms. All OD pairs of the whole network are considered in the experiments.

To compare the different SBO solving algorithms, we design five test scenarios summarized in Table 2. Each scenario has a different synthetic Θ . The selection of synthetic Θ can represent different assumptions about passengers’ choice behavior and sensitivity to crowding. For the reference scenario, we use the path choice parameters in Table 3 as the synthetic β and use the estimated train capacity parameters in [13] as the synthetic θ .

Passengers’ actual path choice behavior is assumed to be random (each path is equally likely to be selected) or deterministic. For the random path choice scenario, we set all synthetic choice parameters as 0, which means all available paths are equally likely to be chosen. For the deterministic (the word “deterministic” here just represents the degree of

randomness is low. The “truly” deterministic corresponds to all parameters go to $\rightarrow -\infty$) path choice scenario, we set all synthetic choice parameters as the lower bounds (i.e., the maximum absolute value possible). Under this scenario, a slight difference in attributes between two paths can lead to a high difference in choice probability (i.e., this is close to passengers following the shortest path). As for the train capacity, the synthetic θ for these two scenarios is the same as the reference scenario.

Passengers’ sensitivity to crowding may also vary. If all passengers are not sensitive to the crowding, train capacity can be modeled as a fixed value. However, if passengers become more sensitive to the crowding, the actual train capacity may largely depend on the crowding level in the train and on the platform. Therefore, passengers’ sensitivity to crowding can be reflected by the scale of θ_1 and θ_2 [13]. For the crowding-sensitive scenario, we set the synthetic train capacity parameters as $\theta_0 = 225$, $\theta_1 = 0.2$, and $\theta_2 = 0.2$. Compared to the reference scenario, θ_1 and θ_2 are higher to represent higher sensitivity. And, θ_0 is decreased to offset the capacity increase caused by the increase of θ_1 and θ_2 . As for the crowding-insensitive scenario, we set the synthetic train capacity parameters as $\theta_0 = 235$, $\theta_1 = 0$, and $\theta_2 = 0$, which can be seen as a fixed-capacity model.

4.2. Case Study Settings. The lower and upper bounds of all parameters ($L_\beta, U_\beta, L_\theta, U_\theta$) are shown in Table 2. Θ^{ini} is set as $(L_\Theta + U_\Theta)/2$ for all scenarios. To compare different algorithms, a fixed computational budget, 100 function evaluations, is applied to all algorithms. All algorithms except for NMSA (deterministic algorithm) are replicated for 5 times (with different random seeds) to decrease the impact of randomness.

4.3. Reference Scenario Results. The convergence results of the reference scenario are depicted in Figure 3. Each point represents the average value over all replications. We found that the performance of different algorithms varied. Given the limited number of function evaluations, CORS, BYO, and SPSA converge to a relatively small objective function. GA, MADs, and SA have relatively large objective function values upon termination. In terms of convergence speed, the response surface methods (BYO and CORS) have the fastest convergence speed. They also reach the lowest objective function value. This is consistent with conclusions regarding the performance of the SBO algorithms when used in the transportation domains [17, 35–37].

Figure 3 also summarizes the behavior of the algorithm stability. The vertical line indicates the $1/4 \times$ standard deviations over the five replications. NMSA is a deterministic algorithm and not affected by randomness. BYO and CORS show high randomness in the first half iterations. However, as the number of function evaluations increases, the standard deviation of the objective function decreases, and the results become stable. GA, SA, and MADs are unstable compared to other algorithms. This means that the heuristic algorithms (GA and SA) are not suitable for the calibration problem studied in this paper. The instability of

TABLE 2: Scenario design.

Parameter category	Synthetic Θ	Scenarios					Bound
		Reference	Path choice		Train capacity		
			Random	Deterministic	Crowding-sensitive	Crowding-insensitive	
Path choice	In-vehicle time	-0.147	0	-2.0	-0.147	-0.147	[-2, 0]
	Relative walking time	-1.271	0	-5.0	-1.271	-1.271	[-5, 0]
	Number of transfers	-0.573	0	-3.0	-0.573	-0.573	[-3, 0]
	Commonality factor	-3.679	0	-10.0	-3.679	-3.679	[-10, 0]
Train capacity	θ_0	232	232	232	225	235	[220, 260]
	θ_1	0.0732	0.0732	0.0732	0.2	0	[0, 0.2]
	θ_2	0.0607	0.0607	0.0607	0.2	0	[0, 0.2]

TABLE 3: Path choice model estimation results.

	Estimate	Std. error	t -value	
In-vehicle time	-0.147	0.011	-13.64	***
Relative walking time	-1.271	0.278	-4.56	***
Number of transfers	-0.573	0.084	-6.18	***
Commonality factor	-3.679	1.273	-2.89	**
$\rho^2 = 0.54$				

***: $p < 0.01$; **: $p < 0.05$.

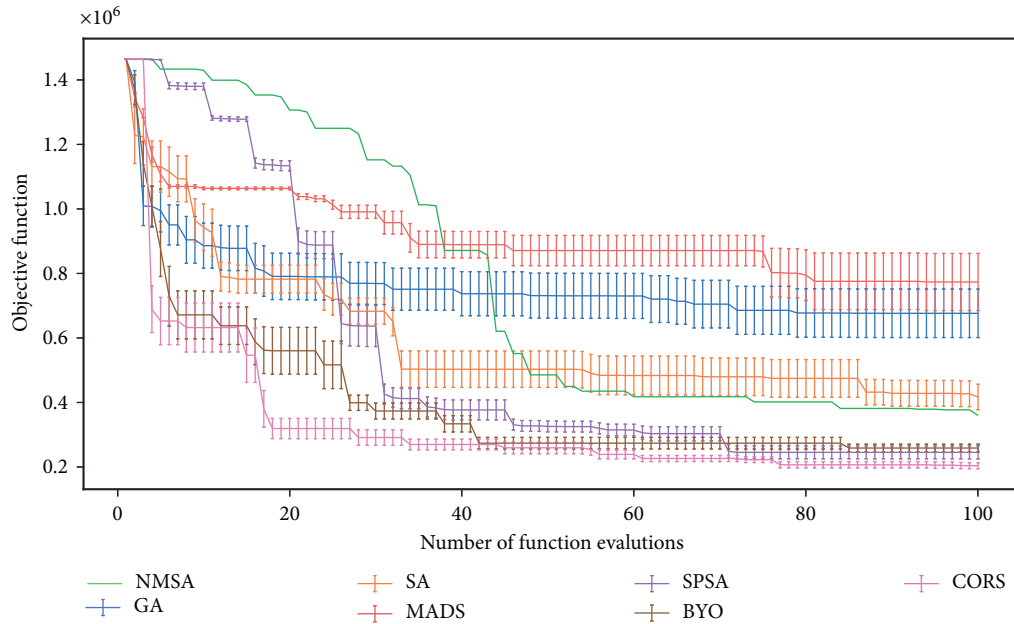


FIGURE 3: Convergence results of reference scenario. The error bar indicates $1/4 \times$ standard deviation. NMSA has no error bar because it is a deterministic algorithm.

MADS may be because it may converge to nonstationary points [38].

Table 4 compares the parameters estimated by different algorithms with the synthetic ones. Although some algorithms can reach similar objective function values, they result in different estimated parameters. For example, CORS and SPSA have similar objective function values. However, SPSA performs better in path choice estimation, while CORS performs better in train capacity estimation. We also observe that the train capacity parameters are relatively harder to estimate. This may be because most of the stations in the rail

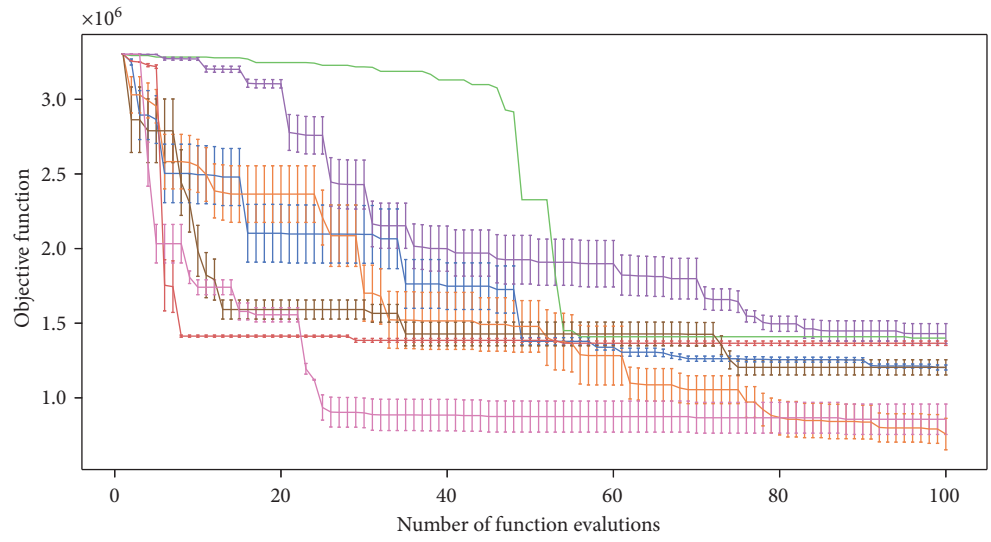
system are not congested and all passengers can board the trains. Thus, the objective function is not very sensitive to train capacity parameters.

4.4. Sensitivity Analysis

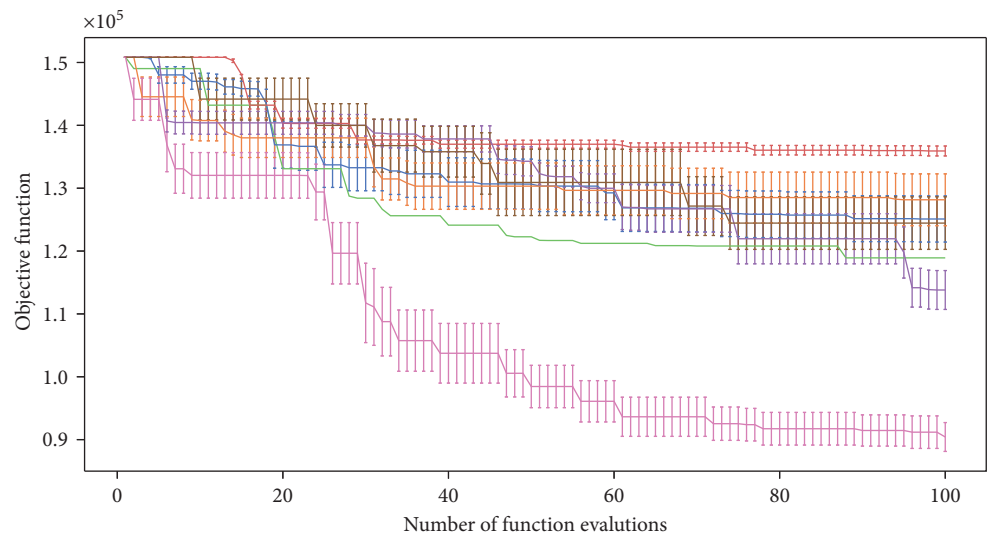
4.4.1. *Impact of Randomness in Path Choice Behavior.* Figure 4 shows the estimation results for the two path choice-related scenarios: random and deterministic. The estimated parameters are shown in Tables 5 and 6. For the

TABLE 4: Estimation results of the reference scenario.

Category	Variable name	"True"	Estimated parameters						
			GA	SA	NMSA	MADS	SPSA	BYO	CORS
Path choice	In-vehicle time	-0.147	-0.392	-0.327	-0.342	-0.454	-0.170	-0.207	-0.229
	Relative walking time	-1.271	-2.205	-3.010	-3.020	-0.302	-2.257	-2.493	-2.486
	Number of transfers	-0.573	-1.143	-0.787	-0.389	-1.248	-0.598	-0.776	-0.756
	Commonality factor	-3.679	-6.482	-6.851	-7.250	-7.834	-4.419	-5.434	-5.716
Train capacity	θ_0	232	239	243	259	252	241	234	243
	θ_1	0.073	0.117	0.118	0.146	0.040	0.162	0.110	0.069
	θ_2	0.061	0.069	0.110	0.080	0.080	0.163	0.100	0.086
Objective function	—	676,392	416,923	359,663	773,526	245,269	258,688	203,885	



(a)



(b)

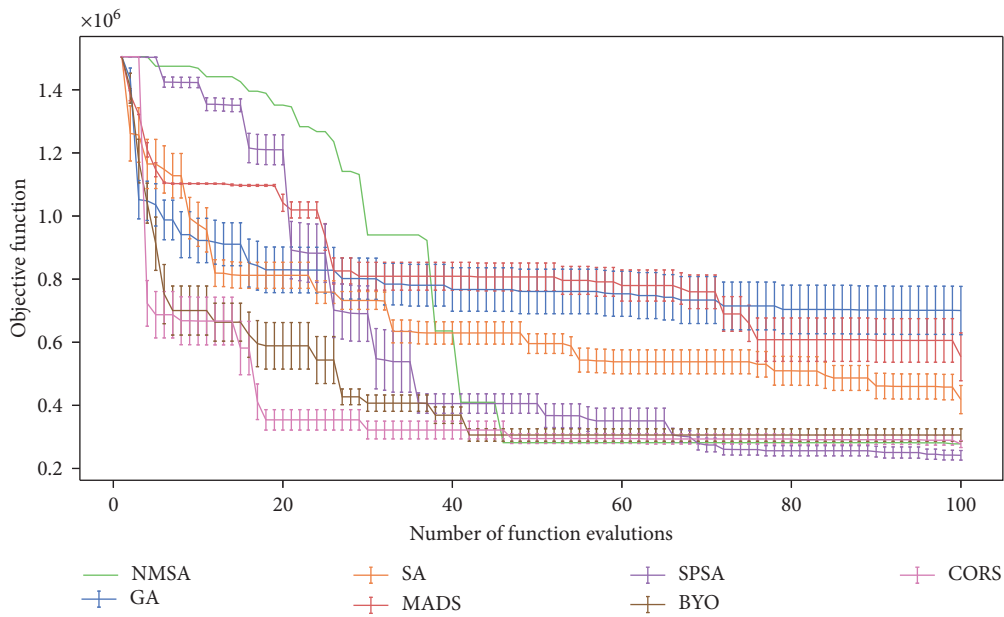
FIGURE 4: Algorithm performance in the two path choice scenarios. (a) Random. (b) Deterministic.

TABLE 5: Estimation results of the random path choice scenario.

Category	Variable Name	"True"	Estimated Parameters						
			GA	SA	NMSA	MADS	SPSA	BYO	CORS
Path choice	In-vehicle time	0	0	-0.072	-0.050	0	-0.108	-0.037	0
	Relative walking time	0	-2.151	-1.139	-1.807	-1.000	-1.719	-3.725	-0.702
	Number of transfers	0	-0.348	-0.185	-0.435	-1.334	-0.631	-0.207	0
	Commonality factor	0	-5.997	-1.945	-9.991	-5.432	-5.127	-4.155	-8.000
Train capacity	θ_0	232	243	224	254	232	241	248	223
	θ_1	0.073	0.067	0.050	0.124	0.048	0.106	0.079	0.016
	θ_2	0.061	0.037	0.072	0.136	0.134	0.112	0.159	0.072
Objective function	—	1,202,761	756,321	1,399,836	1,365,291	1,429,942	1,203,696	855,627	

TABLE 6: Estimation results of the deterministic path choice scenario.

Category	Variable Name	"True"	Estimated Parameters						
			GA	SA	NMSA	MADS	SPSA	BYO	CORS
Path choice	In-vehicle time	-2	-1.240	-1.243	-1.205	-1.160	-1.544	-1.537	-1.830
	Relative walking time	-5	-3.180	-3.358	-2.819	-2.480	-3.728	-3.807	-4.492
	Number of transfers	-3	-1.575	-1.551	-1.419	-1.524	-1.786	-1.761	-2.661
	Commonality factor	-10	-5.307	-5.251	-4.735	-4.920	-6.346	-6.379	-8.819
Train capacity	θ_0	232	237	232	228	237	239	232	237
	θ_1	0.073	0.095	0.076	0.095	0.180	0.097	0.110	0.123
	θ_2	0.061	0.101	0.069	0.091	0.062	0.110	0.106	0.093
Objective function	—	125,100	128,157	118,915	135,922	113,805	124,448	63,220	



(a)

FIGURE 5: Continued.

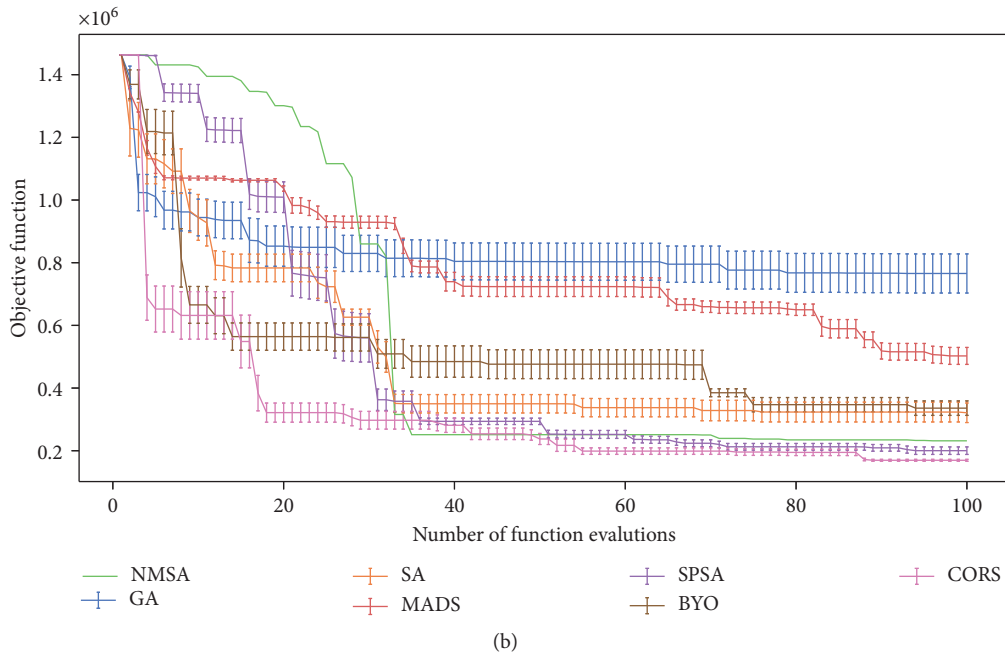


FIGURE 5: Algorithm performance in the two train capacity scenarios. (a) Crowding-insensitive. (b) Crowding-sensitive.

TABLE 7: Estimation results of the crowding-insensitive train capacity scenario.

Category	Variable Name	“True”	Estimated Parameters						
			GA	SA	NMSA	MADS	SPSA	BYO	CORS
Path choice	In-vehicle time	-0.147	-0.392	-0.181	-0.254	-0.460	-0.191	-0.197	-0.177
	Relative walking time	-1.271	-2.153	-2.044	-2.636	-2.294	-2.284	-2.469	-2.025
	Number of transfers	-0.573	-1.127	-1.614	-1.279	-0.490	-0.760	-0.908	-1.011
	Commonality factor	-3.679	-6.489	-6.500	-7.492	-7.750	-5.299	-5.474	-5.130
Train capacity	θ_0	235	239	245	249	230	241	238	236
	θ_1	0	0.088	0.109	0.096	0.050	0.096	0.093	0.084
	θ_2	0	0.05	0.058	0.108	0.050	0.150	0.078	0.063
Objective function	—	700,441	418,196	277,835	553,765	241,533	305,846	277,212	

TABLE 8: Estimation results of the crowding-sensitive train capacity scenario.

Category	Variable Name	“True”	Estimated Parameters						
			GA	SA	NMSA	MADS	SPSA	BYO	CORS
Path choice	In-vehicle time	-0.147	-0.472	-0.217	-0.228	-0.332	-0.177	-0.195	-0.196
	Relative walking time	-1.271	-2.533	-1.575	-2.735	-1.568	-2.118	-1.763	-2.534
	Number of transfers	-0.573	-0.759	-1.169	-1.323	-0.816	-0.495	-0.892	-0.734
	Commonality factor	-3.679	-6.489	-6.324	-7.040	-7.834	-4.361	-6.046	-5.021
Train capacity	θ_0	225	238	244	238	245	244	237	237
	θ_1	0.2	0.166	0.149	0.121	0.112	0.140	0.085	0.099
	θ_2	0.2	0.080	0.114	0.125	0.144	0.129	0.123	0.110
Objective function	—	765,621	320,745	231,228	502,341	199,753	335,558	169,057	

random scenario, all “true” (synthetic) path choice parameters are set as zero, which means all paths are equally likely to be chosen. We observe that, in this scenario (Figure 4(a)), CORS and SA algorithms perform the best with the lowest objective function. Compared to the

reference scenario in Section 4.3, the decreased performance of BYO and SPSA may be due to the “true” β is close to the upper bound ($U_\beta = 0$). The Gaussian posterior distribution in BYO and gradient estimation in SPSA can suffer from instability in the boundary. From Table 5, we observe the

parameters of in-vehicle time and number of transfers are better estimated than those of relative walking time and commonality factors.

Figure 4(b) shows the results of the deterministic scenario. The initial objective function is relatively small (1.5×10^5) compared to the reference scenario (1.5×10^6). All algorithms only reduce the objective function by around 1/3 except for the CORS algorithm. The good performance of CORS may come from global searching with the Latin hypercube method. It is better suited to explore the points near boundaries. Although the objective function does not decrease too much, the estimated parameters are still acceptable (see Table 6).

4.4.2. Impact of Crowding Sensitivity. Figure 5 shows the estimation results of the two scenarios related to train capacity (i.e., crowding-sensitive and crowding-insensitive). In the crowding-insensitive scenario (Figure 5(a)), the conclusions are similar to the reference scenario. CORS, BYO, NMSA, and SPSA converge to low objective function values and outperform other algorithms. The performance of NMSA and MADS is improved compared to the reference scenario. In the crowding-sensitive scenario, we still observe a good performance by the CORS, NMSA, and SPSA algorithms. The performance of BYO is slightly reduced. The results shown in Tables 7 and 8 indicate that θ_0 (base capacity) is hard to estimate. This may be because trains at most stations do not reach the capacity. Therefore, for many OD pairs, the OD exit flows (directly related to the objective function) are not sensitive to the base capacity parameter.

5. Conclusion

In this paper, we propose an SBO framework to calibrate train capacity and path choice model parameters simultaneously for metro systems using AFC and AVL data. The advantage of the proposed framework lies in capturing the collective effect of both path choices and train capacity on passenger journey times. Seven representative algorithms from four main branches of SBO methods are applied and compared with respect to their solution accuracy, convergence speed, and stability. We applied the proposed framework using data from the Hong Kong MTR network and compared the performance of the different algorithms. Overall, the results show that some algorithms result in a reasonable estimation of the parameters of interest. These results also support the effectiveness of the proposed SBO framework for calibrating these key parameters using AFC and AVL data. Especially, the response surface methods (particularly CORS) exhibit consistently good performance. The SBO framework is flexible to accommodate a wide range of path choice and train capacity models in transit simulation models.

This paper has some limitations. First, we validate the framework and evaluate the algorithmic performance only using synthetic AFC and AVL data. Therefore, the complexities of noise and uncertainties in actual data do not play any role. This is caused by the absence of real-world path

choice and train capacity information. Future research can collect real-world path choice and train capacity data to conduct more realistic model validation. Second, we assumed that the path choice behavior is similar for the whole network (same β values). Given the real-world path choice behavior is possibly more diverse and heterogeneous, future research can explore clustering different OD pairs with different β values based on individual mobility characteristics [39].

Appendix

(A). Passenger Path Choice Model for MTR System

These results are from [34]. The C-logit model formulation is the same as equations (1) and (2). A total number of 31,640 passengers completed the questionnaire. After filtering duplicate responses, 26,996 responses were available. The model results are shown in Table 3. The main explanatory variables are the total in-vehicle time, relative transfer walking time, and number of transfers. All variables are statistically significant with the expected signs. Paths with high in-vehicle time, walking time, and number of transfers are less likely to be chosen by passengers.

Data Availability

The AFC and AVL 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.

Authors' Contributions

B. Mo, Z. Ma, H.N. Koutsopoulos, and J. Zhao conceptualized and designed the study; B. Mo and Z. Ma collected the data; B. Mo, Z. Ma, and H.N. Koutsopoulos analysed and interpreted the results; B. Mo and H.N. Koutsopoulos prepared the draft. All authors reviewed the results and approved the final version of the manuscript.

Acknowledgments

The authors would like to thank the Hong Kong Mass Transit Railway (MTR) for their support and data availability for this research. Also, the authors acknowledge MIT Libraries for providing funding for the open-access publication of the paper.

References

- [1] Z. Liu, S. Wang, W. Chen, and Y. Zheng, "Willingness to board: a novel concept for modeling queuing up passengers," *Transportation Research Part B: Methodological*, vol. 90, pp. 70–82, 2016.

- [2] J. Preston, J. Pritchard, and B. Waterson, "Train overcrowding," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2649, no. 1, pp. 1–8, 2017.
- [3] H. N. Koutsopoulos, Z. Ma, P. Noursalehi, and Y. Zhu, "Transit data analytics for planning, monitoring, control, and information," in *Mobility Patterns, Big Data and Transport Analytics*, C. Antoniou, L. Dimitriou, and F. Pereira, Eds., pp. 229–261, Elsevier, Amsterdam, Netherlands, 2019.
- [4] T. Kusakabe, T. Iryo, and Y. Asakura, "Estimation method for railway passengers' train choice behavior with smart card transaction data," *Transportation*, vol. 37, no. 5, pp. 731–749, 2010.
- [5] F. Zhou and R.-H. Xu, "Model of passenger flow assignment for urban rail transit based on entry and exit time constraints," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2284, no. 1, pp. 57–61, 2012.
- [6] P. Kumar, A. Khani, and Q. He, "A robust method for estimating transit passenger trajectories using automated data," *Transportation Research Part C: Emerging Technologies*, vol. 95, pp. 731–747, 2018.
- [7] Y. Zhu, H. N. Koutsopoulos, and N. H. Wilson, "Passenger itinerary inference model for congested urban rail networks," *Transportation Research*, vol. 123, no. 7, 2020.
- [8] Y. Sun and R. Xu, "Rail transit travel time reliability and estimation of passenger route choice behavior," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2275, no. 1, pp. 58–67, 2012.
- [9] L. Sun, Y. Lu, J. G. Jin, D.-H. Lee, and K. W. Axhausen, "An integrated bayesian approach for passenger flow assignment in metro networks," *Transportation Research Part C: Emerging Technologies*, vol. 52, pp. 116–131, 2015.
- [10] J. Zhao, F. Zhang, L. Tu et al., "Estimation of passenger route choice pattern using smart card data for complex metro systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 4, pp. 790–801, 2017.
- [11] X. Xu, L. Xie, H. Li, and L. Qin, "Learning the route choice behavior of subway passengers from a/c data," *Expert Systems with Applications*, vol. 95, pp. 324–332, 2018.
- [12] B. Mo, Z. Ma, H. Koutsopoulos, and J. Zhao, "Assignment-based path choice estimation for metro system using smart card data," in *Proceedings of the 24th International Symposium on Transportation & Traffic Theory (ISTTT)*, Beijing, China, July 2020.
- [13] B. Mo, Z. Ma, H. N. Koutsopoulos, and J. Zhao, "Capacity-constrained network performance model for urban rail systems," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2674, no. 5, pp. 59–69, 2020.
- [14] E. Cascetta, A. Nuzzolo, F. Russo, and A. Vitetta, "A modified logit route choice model overcoming path overlapping problems. specification and some calibration results for interurban networks," in *Proceedings of the 13th International Symposium on Transportation and Traffic Theory*, Lyon, France, July 1996.
- [15] C. G. Prato, "Route choice modeling: past, present and future research directions," *Journal of Choice Modelling*, vol. 2, no. 1, pp. 65–100, 2009.
- [16] M. E. Ben-Akiva and S. R. Lerman, *Discrete Choice Analysis: Theory and Application to Travel Demand*, MIT press, Cambridge, MA, USA, 1985.
- [17] C. Osorio and M. Bierlaire, "A simulation-based optimization framework for urban transportation problems," *Operations Research*, vol. 61, no. 6, pp. 1333–1345, 2013.
- [18] S. Amaran, N. V. Sahinidis, B. Sharda, and S. J. Bury, "Simulation optimization: a review of algorithms and applications," *Annals of Operations Research*, vol. 240, no. 1, pp. 351–380, 2016.
- [19] F. A. Fortin, F. M. De Rainville, M. A. Gardner, M. Parizeau, and C. Gagné, "DEAP: evolutionary algorithms made easy," *Journal of Machine Learning Research*, vol. 13, pp. 2171–2175, 2012.
- [20] C. Tsallis and D. A. Stariolo, "Generalized simulated annealing," *Physica A: Statistical Mechanics and Its Applications*, vol. 233, no. 1-2, pp. 395–406, 1996.
- [21] F. Gao and L. Han, "Implementing the nelder-mead simplex algorithm with adaptive parameters," *Computational Optimization and Applications*, vol. 51, no. 1, pp. 259–277, 2012.
- [22] M. A. Abramson, C. Audet, J. E. Dennis, and S. L. Digabel, "Orthomads: a deterministic mads instance with orthogonal directions," *SIAM Journal on Optimization*, vol. 20, no. 2, pp. 948–966, 2009.
- [23] J. C. Spall, "Multivariate stochastic approximation using a simultaneous perturbation gradient approximation," *IEEE Transactions on Automatic Control*, vol. 37, no. 3, pp. 332–341, 1992.
- [24] J. Snoek, H. Larochelle, and R. P. Adams, "Practical bayesian optimization of machine learning algorithms," in *Proceedings of the 25th International Conference on Neural Information Processing Systems*, pp. 2951–2959, Krong Siem Reap, Cambodia, December 2012.
- [25] R. G. Regis and C. A. Shoemaker, "Constrained global optimization of expensive black box functions using radial basis functions," *Journal of Global Optimization*, vol. 31, no. 1, pp. 153–171, 2005.
- [26] D. Whitley, "A genetic algorithm tutorial," *Statistics and Computing*, vol. 4, pp. 65–85, 1994.
- [27] P. J. Van Laarhoven and E. H. Aarts, "Simulated annealing," in *Simulated Annealing: Theory and Applications* Springer, Berlin, Germany, 1987.
- [28] Scipy, "Scipy dual annealing algorithm," 2019, https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.dual_annealing.htmlURL:..
- [29] J. A. Nelder and R. Mead, "A simplex method for function minimization," *The Computer Journal*, vol. 7, no. 4, pp. 308–313, 1965.
- [30] C. Audet and J. E. Dennis, "Mesh adaptive direct search algorithms for constrained optimization," *SIAM Journal on Optimization*, vol. 17, no. 1, pp. 188–217, 2006.
- [31] C. Audet, S. Le Digabel, and C. Tribes, "Nomad user guide Rapport technique," 2009.
- [32] J. Gomez-Dans, "A simultaneous perturbation stochastic approximation optimisation code in python," 2012.
- [33] P. Knysh and Y. Korkolis, "Blackbox: a procedure for parallel optimization of expensive black-box functions," 2016, <https://arxiv.org/abs/1605.00998>.
- [34] W. Li, "Route and transfer station choice modeling in the mtr system," Working paper, 2014.
- [35] Q. Cheng, S. Wang, Z. Liu, and Y. Yuan, "Surrogate-based simulation optimization approach for day-to-day dynamics model calibration with real data," *Transportation Research Part C: Emerging Technologies*, vol. 105, pp. 422–438, 2019.
- [36] B. Mo, Z. Ma, H. Koutsopoulos, and J. Zhao, "Calibrating route choice for urban rail system: a comparative analysis using simulation-based optimization methods," in *Proceedings of the Transportation Research Board 99th Annual Meeting*, Washington, D.C, USA, January 2020.

- [37] B. Mo, "Network performance model for urban rail systems," Master's Thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 2020.
- [38] M. A. Abramson and C. Audet, "Convergence of mesh adaptive direct search to second-order stationary points," *SIAM Journal on Optimization*, vol. 17, no. 2, pp. 606–619, 2006.
- [39] B. Mo, Z. Zhao, H. N. Koutsopoulos, and J. Zhao, "Individual mobility prediction: an interpretable activity-based hidden markov approach," 2021, <https://arxiv.org/abs/2101.03996>.

Research Article

Profit Maximization Model with Fare Structures and Subsidy Constraints for Urban Rail Transit

Qing Wang ^{1,2}, Paul Schonfeld ³, and Lianbo Deng ¹

¹School of Traffic and Transportation Engineering, Rail Data Research and Application Key Laboratory of Hunan Province, Central South University, Changsha 410075, China

²Department of Civil and Environmental Engineering, University of Washington, Seattle, WA 98105, USA

³Department of Civil and Environmental Engineering, University of Maryland, College Park 1173 Glenn Martin Hall, College Park, MD 20742, USA

Correspondence should be addressed to Lianbo Deng; lbdeng@csu.edu.cn

Received 15 November 2020; Revised 29 December 2020; Accepted 13 January 2021; Published 25 January 2021

Academic Editor: Erfan Hassannayebi

Copyright © 2021 Qing Wang 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.

This paper analyzes government subsidies based on the service design (i.e., headway) and fare structures of an urban rail transit system while considering necessary financial support from the government. To capture the interactions among the operator performance, government subsidies, and passengers in an urban rail transit system, a profit maximization model with nonnegative profit constraint is formulated to determine the optimal fare and headway solutions. Then, the social welfare that results from the operator profit maximization model is analyzed. Finally, a numerical example from Changsha, China, is employed to verify the feasibility of the proposed model. The major results consist of optimized solutions for decision variables, i.e., the fares and train headways, as well as subsidies to the operator. The fare elasticity factor under two fare structures significantly affects fares and demand. As the fare elasticity factor increases, the social welfare gradually decreases and a deficit occurs at low fares and demand, while subsidies rise from 0 to ¥24658.00 and ¥38089.16 under the flat fare and distance-based fare structures.

1. Introduction

In recent decades, large-scale investment by local authorities in China has greatly promoted the pace of urban rail transit (URT) construction and operation. According to the “Annual Statistics and Analysis Report of URT 2019”, up to the end of 2019, there were 208 URT lines in (mainland) China, distributed in 40 cities, including Shanghai, Beijing, Guangzhou, and Nanjing, with a total length of 5180.6 km in operation, and the ridership has exceeded 237.1 billion passengers per year. In general, operators in most cities are overdependent on the government’s subsidies (data source: China Association of Metros, 2019 [1]).

A comprehensive review of the transportation issues was conducted by Farahani et al. [2], which discussed and compared the models and solution methods of transportation network design problem. Although many studies have investigated the optimization of public transportation,

the literature on methods for optimizing URT system operation with subsidy constraints while considering different fare structures is still relatively scarce. For instance, Li and Love [3] conducted a retrospective analysis of a rail line that was procured using a public-private partnership in conjunction with land value capture. They showed that the economic viability of that URT system could be ensured by considering the land value capture. Canca et al. [4] developed a mathematical programming model that maximized net profit by simultaneously determining the infrastructure network and line planning problem. The effect of a surcharge-reward scheme relieving crowding and queuing congestion in a URT system was investigated in Tang et al. [5], who formulated a bilevel model to design and optimize the surcharge-reward scheme. Since fares are closely related to operator profit and subsidies, the implementation of fare differentiation is one of the practical policies adopted in public transport management [6]. Further studies on the

relations between fares and operator performance as well as passenger travel behavior have been conducted. For example, a model was developed to optimize the transit fare structure with demand elasticity, but with a fixed service frequency [7]. Then, Chien and Spasovic [8] proposed a model that optimized fares by considering demand without excessively simplifying spatial characteristics and demand patterns. The work of Chien and Spasovic [8] was extended by Sharaby and Shiftan [9], who studied the impacts of fares on demand and travel behavior. The optimization of fares has often been related to the service frequency (or its inverse, the headway, in this study), which has been investigated by many researchers. Chien and Tsai [10] constructed an optimization model for maximizing profits and considered the impact of varying demand on headway and fare. Jin et al. [11] proposed a social welfare maximization model for optimizing fare levels by considering the demand and service quality of public transport. Wang and Deng [12] developed a model for optimizing distance-based fare structure (DBF) and headway by considering the maximum operator profit and minimum per capita subsidy for passengers. Besides, passengers consider many factors in their travel choices, including service levels, generalized travel time, and fares [13]. These factors can be subsumed into the passenger travel behavior of public transportation systems.

Different fare structures are mainly used as one of the indicators for evaluating passenger travel behavior, operator performance, and other aspects. For flat fare structure (FF) optimization, Wang et al. [14] investigated public transit service (i.e., flat fare and frequency) operation strategies in a bimodal network. Jin et al. [11] focused on a flat fare structure and found that low fares are preferable from the viewpoint of maximizing social welfare. For DBF optimization, Tsai et al. [15] proposed a profit maximization model for maximizing DBF and service headway. Through sensitivity analyses, the results indicate that the optimized fare and headway decrease as the demand increases, which results in a profit increase. In addition, some studies also proposed models that optimized other fare structures, such as a zonal fare [13], a sectional fare [16], and an Origin-Destination (OD) fare [17]. Besides, several papers have discussed the feasibility and importance of FF and DBF [18]. The present study only considers FF and DBF since these two are the most widely used fare structures in public transportation.

Another stream of the literature related to our work focuses on operator performance and passenger travel behavior. A series of studies have been conducted to analyze the operators' performance. A model that optimizes operator performance such as frequency and vehicle fleet sizes with financial policies was formulated by Jara-Díaz and Gschwender [19], which investigated the effect that overall economic policies may have on the operation of public service. The efficiency of and the substitutability between different management policies have been analyzed in [20]. The model features operator performance between cars and transit. Several studies have analyzed the passengers' travel characteristics. Gkritza et al. [21] pointed out that riders are sensitive to changes in absolute fare levels as well as relative

price. Considering the effect of fares on passenger travel behavior, Nassi and Costa [22] evaluated a region's optimal fare system by using the analytic hierarchy process (AHP). Table 1 highlights the novelties of the model proposed in this paper through comparison with previous studies.

As reviewed above, previous studies have optimized fare structures and subsidies separately, but no published study has compared the impact of subsidies for different fare structures by considering social welfare. More related to our study, several scholars have investigated operation performance with subsidy constraints. To explore different financial constraints, Zhou et al. [23] proposed a maximum social welfare model for optimizing bus transit systems. Through numerical study, the results showed that the effects of subsidies on social welfare differed for fixed and flexible-route bus systems. Wang and Deng [12] studied the impact of per capita subsidy on passengers and proposed an efficiency-oriented model for maximizing the efficiency of per capita subsidy. A break-even subsidy model for optimizing fares and headways has been developed by Wang et al. [24]. This study identified the effect of two fare structures and headway on operational subsidies.

We recognize that the subsidy to operators may be related to operator performance and fare structures. Our work is extended from Zhou et al. [23] and Wang et al. [24] by considering the impact of fare structures and operator performance on operations. Therefore, we formulate a profit maximization model for operators who charge fares that optimize social welfare and determine the headway optimized in response to the government's financial constraints. The major contributions of this paper are summarized as follows:

- (i) This research comprehensively considers subsidy constraints and fare structures of URT system optimization to determine the operator performance and passenger travel behavior. A profit maximization model, with a many-to-many demand pattern, for optimizing fares and headways to maximize the operator profit is developed by considering flat fare (FF) and distance-based fare (DBF) structures as well as a subsidy constraint.
- (ii) This paper compares the performance of FF and DBF structures through numerical studies. It is found that FF requires more subsidies and is more attractive for long-distance passengers, while DBF is more profitable and attractive for short-distance passengers.
- (iii) We investigate the operator performance under the government's subsidy constraints and different fare structures and compare the effectiveness of the two fare structures at attracting passenger demand and maximizing social welfare. Through the operator profit maximization model with fare structures and subsidy constraints, we obtain the levels of the demand and fares that require no subsidy.
- (iv) We obtain the optimal function of fare and headway. Besides, the fare levels significantly affect operator performance, thus affecting subsidies to

TABLE 1: Related studies.

Citation	Considering social welfare	Fare structures		Considering subsidy	Demand pattern
		FF	DBF		
Chien and Spasovic [8]	✓	✓	—	—	✓
Chien and Tsai [10]	—	✓	✓	—	✓
Wang and Deng [12]	—	—	✓	✓	✓
Huang et al. [17]	—	—	✓	✓	✓
Jara-Díaz and Gschwender [19]	✓	✓	—	✓	—
Basso and Silva [20]	—	✓	—	✓	—
Zhou et al. [23]	—	✓	—	✓	—
Wang et al. [24]	—	✓	✓	✓	✓
Sun et al. [25]	✓	✓	—	✓	—
Ling et al. [26]	—	✓	—	✓	—
This study	✓	✓	✓	✓	✓

operators. Comparing the effects of changes in fare elasticity factor on subsidies, we find that the fare elasticity factor affects the DBF fare rate (i.e., the variable component of DBF fare) more than the FF fare and affects the revenue more than the operating cost under FF and DBF.

The paper is organized as follows. In the next section, preliminaries are described, including the URT network characteristics and important functions related to optimization models. Section 3 presents the operator profit models under FF and DBF and provides the solution discussion for two models. The performance and application of our proposed models are evaluated through numerical experiments in Section 4. Finally, conclusions with major findings as well as prospective research directions are provided in Section 5.

2. Preliminaries

In this paper, a URT line is represented by (S, E) , which contains the station set and section set. Let $S = \{1, 2, \dots, N\}$ be the set of nodes for stations and let $E = \{e_{ij}|i, j \in S\}$ be the set of sections for the line. For each section $e_{ij} \in E$, the distance between stations i and j ($i, j \in S$) is d_{ij} . Let $D_S, S = 1, 2, \dots, N$ represent the total length of the line between the OD stations. The following assumptions are made before formulating the models.

- (i) *Assumption 1.* The URT trains are assumed to have the same number of railcars and the same fixed dwell times at each station.
- (ii) *Assumption 2.* The study period is assumed to be one hour, i.e., demand is an average hourly passenger flow of the day. We neglect here the differences between peak and off-peak hours in order to focus on subsidies, fare structures, and operator profit maximization model.
- (iii) *Assumption 3.* All revenue of operators is obtained from fares, and no other revenue sources are considered (e.g., advertisement revenue). This means that the subsidies found here are only related to operations.

- (iv) *Assumption 4.* The average waiting time of passengers at all stations along a URT line is the same constant fraction of the headway, i.e., usually half of the headway if passengers and trains arrive uniformly over time.

2.1. Fare Structures. Since FF and DBF are considered here the fare per passenger trip can be written as follows:

$$P = \begin{cases} \bar{P}, & \text{(for FF),} \\ \hat{P} = p_0 + \hat{p}d_{ij}, & \text{(for DBF),} \end{cases} \quad (1)$$

where the fare, \hat{P} , for DBF includes a fixed component p_0 and a variable component, \hat{p} .

2.2. Elastic Demand Function. Let Q_{ij} be the URT passenger volume from stations i to j and q_{ij} be the potential demand during the study period. Referring to Wang and Deng [12] and Wang et al. [24], we can obtain the passenger elastic demand function:

$$Q_{ij} = q_{ij}(1 - e_w w_t - e_r r_t - e_p P), \quad (2)$$

where w_t, r_t are the waiting time per passenger and train riding time, respectively, and e_w, e_r, e_p are parameters for waiting time, riding time, and fare, respectively.

The total riding time of passengers between stations i and j is the sum of the train running time, (d_{ij}/v) , and train dwell time, $|j - i - 1| \cdot t_0$, where t_0 is the average train dwell time at each station. The passenger riding time, r_t , can be expressed as

$$r_t = \frac{d_{ij}}{v} + |j - i - 1| \cdot t_0. \quad (3)$$

According to assumption 4, the average passenger waiting time is $w_t = \sigma \cdot H$, where $\sigma = 0.5$ is the waiting time parameter. Thus, (2) can be rewritten as

$$Q_{ij} = q_{ij}(1 - e_w \sigma H - e_r r_t - e_p P). \quad (4)$$

To ensure the nonnegativity of the demand, the elastic demand function should satisfy the following condition:

$$0 \leq 1 - e_w \sigma H - e_r r_t - e_p P \leq 1. \quad (5)$$

2.3. Operating Costs Function. Following Wang et al. [24], the total operating cost consists of three components: train operating cost (C_T), rail line maintenance and operation cost (C_L), and station service and operation cost (C_S); i.e.,

$$C = C_T + C_L + C_S. \quad (6)$$

The train operating cost includes the variable cost, Nc_O (i.e., the cost of trains operating on the line), and the fixed cost, $\beta \varepsilon Nc_O$ (i.e., the cost of reserve trains waiting to be operated), which can be expressed as

$$C_T = Nc_O + \beta \varepsilon Nc_O. \quad (7)$$

The number and cost of reserve trains are, respectively, ε times and β times those of the operating trains. In (7), the number of operating trains, $N = (T_R/H)$, equals the train round trip time, T_R , divided by the headway, H . The train

round trip time is $T_R = 2((D_s/v) + L_S t_0) + t_z$, comprising the nonstop line-haul travel time (D_s/v), dwell time, $L_S t_0$, and the train reversing time t_z .

The second term is the sum of the variable cost, (γ_1/H) (the cost for line use related to the operating frequency $1/H$) and the fixed cost $\gamma_0 D_s$ (i.e., the cost of the rail line maintenance related to the total length of the line), which can be expressed as

$$C_L = \gamma_0 D_s + \frac{\gamma_1}{H}. \quad (8)$$

The last term comprises the fixed cost for a station operating and variable cost (i.e., the number of passengers served per hour). The service costs grow linearly with the passenger volume at each station, which can be expressed as

$$C_S = \Lambda_0 L_S + \Lambda_1 \sum_{i,j \in S} q_{ij} (1 - e_w \sigma H - e_r r_t - e_p P). \quad (9)$$

Then, the operating cost function (C) under FF and DBF can be reformulated as follows:

$$C = \begin{cases} \tilde{C} = (1 + \beta \varepsilon) \frac{T_R}{H} \cdot c_O + \left(\gamma_0 D_s + \frac{\gamma_1}{H} \right) + \Lambda_0 L_S + \Lambda_1 \sum_{i,j \in S} Q_{ij}, & \text{(for FF),} \\ \hat{C} = (1 + \beta \varepsilon) \frac{T_R}{H} \cdot c_O + \left(\gamma_0 D_s + \frac{\gamma_1}{H} \right) + \Lambda_0 L_S + \Lambda_1 \sum_{i,j \in S} Q_{ij}, & \text{(for DBF).} \end{cases} \quad (10)$$

2.4. Revenue Function. According to assumption 3, the revenue of the URT system is a sum of the fares paid by all passengers. The revenue function (R) under FF and DBF can be expressed as

$$R = \begin{cases} \tilde{R} = \tilde{P} \cdot \sum_{i,j \in S} q_{ij} (1 - e_w \sigma \tilde{H} - e_r r_t - e_p \tilde{P}), & \text{(for FF),} \\ \hat{R} = \sum_{i,j \in S} \hat{P} \cdot q_{ij} (1 - e_w \sigma \hat{H} - e_r r_t - e_p \hat{P}), & \text{(for DBF).} \end{cases} \quad (11)$$

3. Model Formulation

3.1. Operator Profit Maximization Model. In this section, operator profit maximization model is analyzed with a subsidy constraint under two fare structures (i.e., FF and DBF). The decision variables are headway H and fare P . In view of (1)–(11), the operator profit ($\tilde{\pi}(\tilde{H}, \tilde{P})$ or $\hat{\pi}(\hat{H}, \hat{P})$) maximization problem can be formulated as follows. For FF,

$$\begin{aligned} \tilde{\pi}(\tilde{H}, \tilde{P}) = & \tilde{P} \cdot \sum_{i,j \in S} q_{ij} (1 - e_w \sigma \tilde{H} - e_r r_t - e_p \tilde{P}) \\ & - \left((1 + \beta \varepsilon) \frac{T_R}{H} \cdot c_O + \left(\gamma_0 D_s + \frac{\gamma_1}{H} \right) + \Lambda_0 L_S + \Lambda_1 \sum_{i,j \in S} q_{ij} (1 - e_w \sigma \tilde{H} - e_r r_t - e_p \tilde{P}) \right), \end{aligned} \quad (12)$$

subject to

$$\tilde{R} - \tilde{C} + \tilde{S}_{\text{flat}} \geq 0. \quad (13)$$

For DBF,

$$\begin{aligned} \hat{\pi}(\hat{H}, \hat{P}) = & \sum_{i,j \in S} \hat{P} \cdot q_{ij} (1 - e_w \sigma \hat{H} - e_r r_t - e_p \hat{P}) \\ & - \left((1 + \beta \epsilon) \frac{T_R}{\hat{H}} \cdot c_O + \left(\gamma_0 D_s + \frac{\gamma_1}{\hat{H}} \right) + \Lambda_0 L_S + \Lambda_1 \sum_{i,j \in S} q_{ij} (1 - e_w \sigma \hat{H} - e_r r_t - e_p \hat{P}) \right), \end{aligned} \quad (14)$$

subject to

$$\hat{R} - \hat{C} + \hat{S}_{\text{distance}} \geq 0. \quad (15)$$

Constraint (i.e., $R - C + S \geq 0$) guarantees the non-negativity of the operator profit. Thus, the profit should be nonnegative after the government's subsidies.

3.2. Solution Discussion. It is easy to verify that the operator profit function is concave with respect to decision variables, i.e., fare and headway (more details are shown in Appendix A.). Therefore, we consider the first-order conditions of (12) or (14); i.e., set to zero the partial derivative of the objective function $\tilde{\pi}(\tilde{H}, \tilde{P})$ (or $\hat{\pi}(\hat{H}, \hat{P})$) with respect to \tilde{H} (or \hat{H}) and \tilde{P} (or \hat{P}), and obtain the functions for the optimal fare and headway as follows.

For FF,

$$\begin{cases} \tilde{P}^* = \frac{\sum_{i,j \in S} q_{ij} (1 - e_w \sigma \tilde{H} - e_r r_t + \Lambda_1 e_p)}{2e_p \sum_{i,j \in S} q_{ij}}, \\ \tilde{H}^* = \sqrt{\frac{(1 + \beta \epsilon) T_R \cdot c_O + \gamma_1}{e_w \sigma (\tilde{P} - \Lambda_1) \sum_{i,j \in S} q_{ij}}}. \end{cases} \quad (16)$$

For DBF,

$$\begin{cases} \hat{P}^* = \frac{\sum_{i,j \in S} d_{ij} q_{ij} (1 - e_w \sigma \hat{H} - e_r r_t + e_p \Lambda_1)}{2e_p \sum_{i,j \in S} q_{ij} d_{ij}}, \\ \hat{H}^* = \sqrt{\frac{(1 + \beta \epsilon) T_R \cdot c_O + \gamma_1}{e_w \sigma \sum_{i,j \in S} q_{ij} ((p_0 + \hat{p} d_{ij}) - \Lambda_1)}}. \end{cases} \quad (17)$$

In the subsidization scheme, a government provides subsidies to compensate for operating deficit if and when the operator faces a negative profit [27]. The subsidy should satisfy the following.

For FF,

$$\tilde{S}_{\text{flat}} = \max\{\tilde{C} - \tilde{R}, 0\}. \quad (18)$$

For DBF,

$$\hat{S}_{\text{distance}} = \max\{\hat{C} - \hat{R}, 0\}. \quad (19)$$

Substituting (16) and (17) into (18) and (19), we can obtain the subsidy to an operator under for FF and DBF as follows.

For FF,

$$\begin{aligned} \tilde{S}_{\text{flat}}^* = & \max \left\{ (1 + \beta \epsilon) \frac{c_O T_R}{\tilde{H}^*} + \left(\gamma_0 D_s + \frac{\gamma_1}{\tilde{H}^*} \right) + \Lambda_0 L_S \right. \\ & \left. + (\Lambda_1 - \tilde{P}^*) \sum_{i,j \in S} q_{ij} (1 - e_w \sigma \tilde{H}^* - e_r r_t - e_p \tilde{P}^*), 0 \right\}. \end{aligned} \quad (20)$$

For DBF,

$$\begin{aligned} \hat{S}_{\text{distance}}^* = & \max \left\{ (1 + \beta \epsilon) \frac{c_O T_R}{\hat{H}^*} + \left(\gamma_0 D_s + \frac{\gamma_1}{\hat{H}^*} \right) + \Lambda_0 L_S \right. \\ & + \sum_{i,j \in S} (\Lambda_1 - \hat{P}^*) q_{ij} (1 - e_w \sigma \hat{H}^* - e_r r_t \\ & \left. - e_p (p_0 + \hat{p} d_{ij})), 0 \right\}. \end{aligned} \quad (21)$$

In general, the user surplus is computed as the integral of the fare function (that is, the inverse of the demand function) concerning the total passenger volume. Following Sun et al. [25], let $B(P)$ be the inverse demand function of the elastic demand function [28].

$$B(P) = Q_{ij}^{-1} = \frac{1 - e_w w_t - e_r r_t - (Q / \sum_{i,j \in S} q_{ij})}{e_p}. \quad (22)$$

Then, the user surplus U can be expressed as follows:

$$U = \begin{cases} \tilde{U} = \int_0^{Q_{ij}} (B(\tilde{P}) - \tilde{P})dq = \frac{1}{2e_p} \sum_{i,j \in S} q_{ij} (1 - e_w \sigma \tilde{H} - e_r r_t - e_p \tilde{P})^2, & \text{(for FF),} \\ \hat{U} = \int_0^{Q_{ij}} (B(\hat{P}) - \hat{P})dq = \frac{1}{2e_p} \sum_{i,j \in S} q_{ij} (1 - e_w \sigma \hat{H} - e_r r_t - e_p \hat{P})^2, & \text{(for DBF).} \end{cases} \quad (23)$$

We now compute the social welfare resulting from the fare and headway obtained by the profit maximization model with subsidy constraints. Combining profit and user surplus, social welfare ($Y(\tilde{P}, \tilde{H})$ or $Y(\hat{P}, \hat{H})$) can be written as follows.

For FF,

$$\begin{aligned} Y(\tilde{P}, \tilde{H}) = & (\tilde{P} - \Lambda_1) \cdot \sum_{i,j \in S} q_{ij} (1 - e_w \sigma \tilde{H} - e_r r_t - e_p \tilde{P}) \\ & - (1 + \beta \varepsilon) \frac{T_R}{\tilde{H}} \cdot c_O - \left(\gamma_0 D_S + \frac{\gamma_1}{\tilde{H}} \right) \\ & - \Lambda_0 L_S + \frac{1}{2e_p} \sum_{i,j \in S} q_{ij} (1 - e_w \sigma \tilde{H} - e_r r_t - e_p \tilde{P})^2. \end{aligned} \quad (24)$$

For DBF,

$$\begin{aligned} Y(\hat{P}, \hat{H}) = & (\hat{P} - \Lambda_1) \cdot \sum_{i,j \in S} q_{ij} (1 - e_w \sigma \hat{H} - e_r r_t - e_p \hat{P}) \\ & - (1 + \beta \varepsilon) \frac{T_R}{\hat{H}} \cdot c_O - \left(\gamma_0 D_S + \frac{\gamma_1}{\hat{H}} \right) \\ & - \Lambda_0 L_S + \frac{1}{2e_p} \sum_{i,j \in S} q_{ij} (1 - e_w \sigma \hat{H} - e_r r_t - e_p \hat{P})^2. \end{aligned} \quad (25)$$

4. Numerical Study

We illustrate an application of the proposed models for Changsha's Metro Line 2 in China. A numerical study investigates the effects of the key model variables and subsidies for different fare structures. In the following analysis, the baseline values of parameters are set as follows:

The average speed is assumed to be 40 km/h, while the average train dwell time, train reserving time, and passenger waiting time at each station are set to be 1/120 h, 0.08 h, and 0.5 h, respectively. The hourly operating cost is ¥1950/vehicle-hour, the unit fixed cost of the line is ¥ 3800/km, and the unit fixed cost of each station is ¥ 4200/km-hr. The demand elasticity parameters for waiting time, riding time, and fare are set at 0.6, 0.15, and 0.1, respectively. The upper and lower boundary of the train operating headway are set at 1/30 (or 2 minutes) and 1/5 (or 12 minutes), while those of the fare are set at 0 and 12, respectively. The values of other input parameters are shown in Table 2. Note that additional references for these parameters can be found in Wang and Deng [12], Wang et al. [24], and China Railway Fourth

Survey and Design Institute Group Co., Ltd. [29]. The values for the demand elasticity parameters are estimated based on historical data [12, 24]. The actual data of Changsha's URT line 2 are obtained from a survey conducted during the planning period, as shown in [29].

4.1. Numerical Results. The optimized results for the operator profit (OP) models under FF and DBF are presented in Table 3. The results are slightly different for two fare structures, while the headways and fare levels are extremely sensitive to the objective. For comparison, optimized solutions are provided from the OP models with fare structures and subsidy constraints. Decision variables at which the OP model maximizes profit are ¥ 4.88 for fare and 8.56 min for headway under FF. For DBF, the corresponding optimized values are ¥ (1.97 + 0.294 $d_{i,j}$) for fare and 9.82 min for headway.

The subsidy is zero and fares are at a higher level for both FF and DBF under the OP models in this numerical study, which means that the subsidy constraint is not binding. For comparison, it must be noted that the problem studied in Sun et al. [25] differs from the one presented here. Sun et al. [25] reported that the financial constraint is binding at optimality in public transit subsidization, and the operators break even after subsidies. However, the subsidy constraint is not binding at optimality when considering the OP models, i.e., $\tilde{S}_{\text{flat}} = 0$ (or $\hat{S}_{\text{distance}} = 0$). Thus, when the OP is positive, no subsidy is needed. In the case of high demand and fares, this is reasonable because operators seek to maximize their profits. The previous study considered the situation where the optimal profit was negative and proposed an efficiency-oriented subsidy optimization method that seeks to maximize the per capita subsidy, so there was an operating deficit in Case [12]12.

4.2. Fare Structures Discussion. The elastic demand function used in this paper is sensitive to the trip length. The travel behavior (i.e., demand and trip length) of passengers under FF and DBF is shown in Figure 1, which plots the demand and fares vs. trip length. When a passenger's trip length is 9.93 km, FF and DBF fares are equal. When the trip is below 9.93 km, the demand with DBF exceeds that with FF, but the fare with DBF is lower than with FF.

Taking the maximum demand gap for example, in the first set of the data (i.e., the first bars indicating demands under FF and DBF in Figure 1(a)), DBF demand is 1.5 times that under FF, whereas FF fare is 2.25 times that under DBF. Therefore, the FF revenue is higher than that under DBF (as shown in Table 1). In Figure 1(b), the DBF demand declines

TABLE 2: Notation.

Parameters	Description	Baseline value
c_O	Average train operating cost per hour (¥/h-vehicle)	1950
e_w	Elasticity parameter for wait time (1/h)	0.6
e_r	Elasticity parameter for riding time (1/h)	0.15
e_p	Elasticity parameter for the fare (1/¥)	0.1
L_S	Number of stations	19
$\frac{P_w}{P_w}$	Lower boundary of the fare (¥)	0
$\frac{P_w}{P_w}$	Upper boundary of the fare (¥)	7
t_0	Train dwell time at each station (h)	1/120
t_Z	Train reversing time (h)	0.08
v	Train speed (km/h)	40
β	Idle trains multiplier (the cost of the nonoperating trains is β -times of the operating trains)	0.24
γ_0	Fixed maintenance costs per line kilometer (¥/km)	3800
γ_1	Cost parameter related to the rail line frequency (¥/h)	525
ε	Reserve factor (the nonoperating trains are ε -times the number of the operating trains)	0.25
σ	Ratio of waiting time to headway (1/h)	0.5
$\underline{\tau}$	Lower boundary of the headway (h)	1/30 h (or 2 minutes)
$\bar{\tau}$	Upper boundary of the headway (h)	1/5 h (or 12 minutes)
Λ_0	Fixed cost parameter for each station (¥/km)	4200
Λ_1	Service cost parameter per passenger at each station (¥)	0.5

TABLE 3: Optimized solution.

Optimized solution	FF	DBF
Fare (¥)	4.88	$1.97 + 0.294d_{ij}$
Headway (min)	8.56	9.82
Revenue (¥/h)	283754.21	245907.52
Operating costs (¥/h)	214949.64	218266.28
Operator profit (¥/h)	68804.57	27641.24
Passenger surplus (¥/h)	146131.37	225264.98
Total social welfare (¥/h)	214935.94	252906.22
Subsidy (¥/h)	0	0
Demand (pass./h)	57967	70989

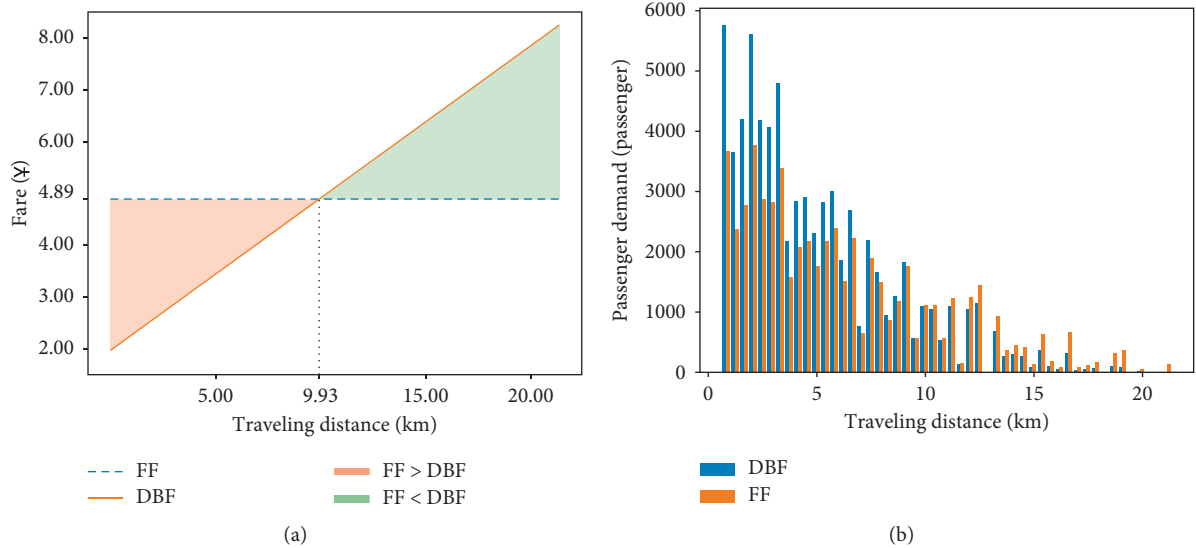


FIGURE 1: Demand and fare for a given operation plan.

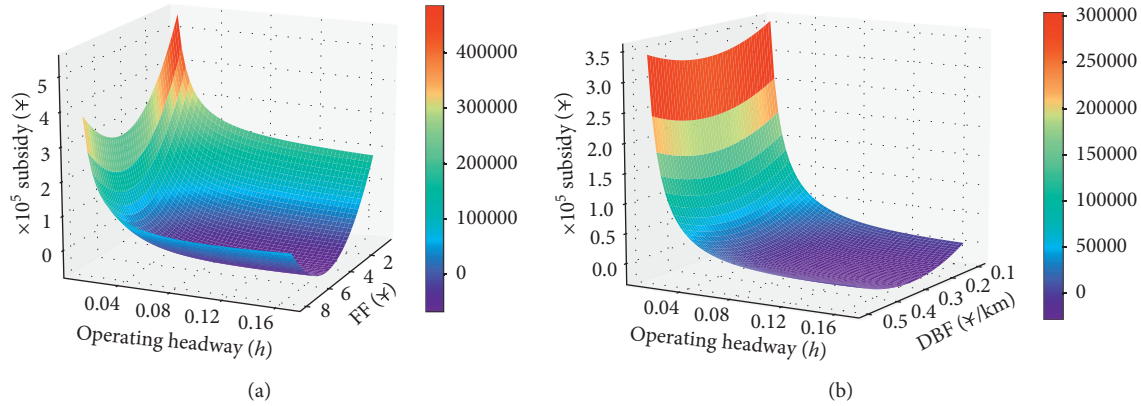


FIGURE 2: Decision variables vs. subsidy under FF and DBF. (a) FF. (b) DBF.

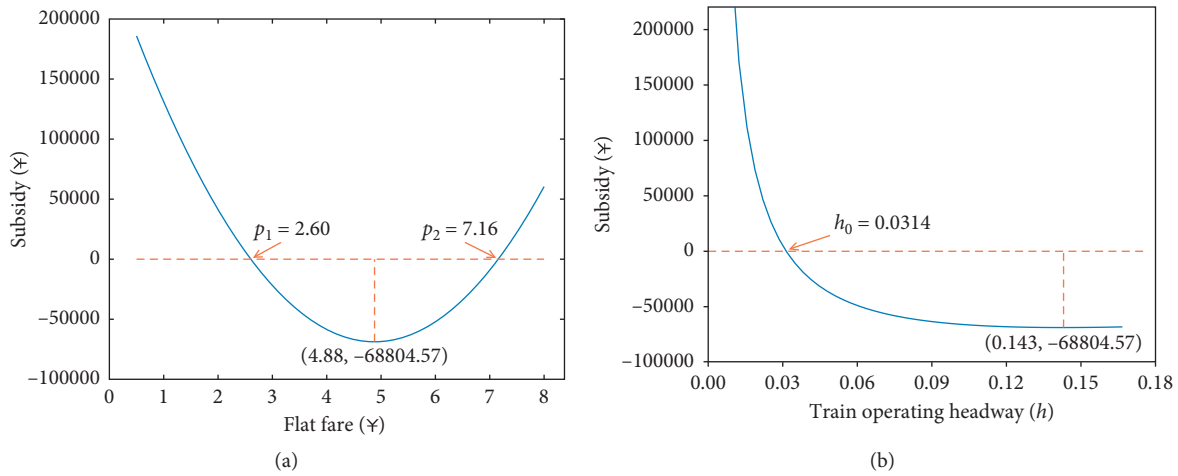


FIGURE 3: Subsidy vs. decision variables under FF.

when the trip exceeds 9.93 km. The DBF demand is almost zero when the trip exceeds 17 km, but there are still some passengers under FF. This occurs because the profit loss caused by the reduction in demand exceeds the profit gain caused by the increase in fares (i.e., the increase in the fare is not enough to compensate for the reduction in demand). For example, the DBF fare is about ¥5.88 when the trip is 17 km, which is 1.32 times higher than FF, but the demand is less than 0.5 of that under FF (in Figure 1).

4.3. Subsidy Discussion and Operator Performance

4.3.1. Effects of Subsidies on Operator Profit. Figures 2(a) and 2(b) show the changes in subsidies as the fares and headway change under FF and DBF. The trend under the two fare structures is similar, and the subsidy has a negative correlation with fare and headway, as expected. As can be seen, the subsidy under DBF decreases faster as the headway increases, compared to the change of the subsidy with the

change of the fare. In contrast, the subsidy under FF changes more significantly as the fare changes. The blue part of Figure 2 shows the operator needs a lower subsidy, while the red part shows the operator needs a higher subsidy. In contrast, at the same headway, higher fares increase revenue. Note that the vertical scale in Figure 2 extends below zero, to allow for a possible negative profit.

Figures 3 and 4 show numerical results associated with various subsidies under FF and DBF. Considering the relation between subsidy and fare with fixed optimal headway \tilde{H}^* (see Figure 3(a)), as FF fare increases, the subsidy decreases from a peak value towards a minimum value, dropping to zero when $\tilde{P}_1 = 2.60$. However, the subsidy reaches its vertex (minimum value of ¥ -68804.57) when the FF fare is ¥ 4.88 (i.e., the operator needs no subsidies when the profit exceeds ¥ 0.0). Beyond the value of FF fare at the vertex, the subsidy rises in a parabolic form from its minimum value, crossing the point of zero value where $\tilde{P}_2 = 7.16$. For a fixed optimal fare \tilde{P}^* (see Figure 3(b)), the subsidy becomes zero when $\tilde{H} = 0.0314$ h. The graph shows

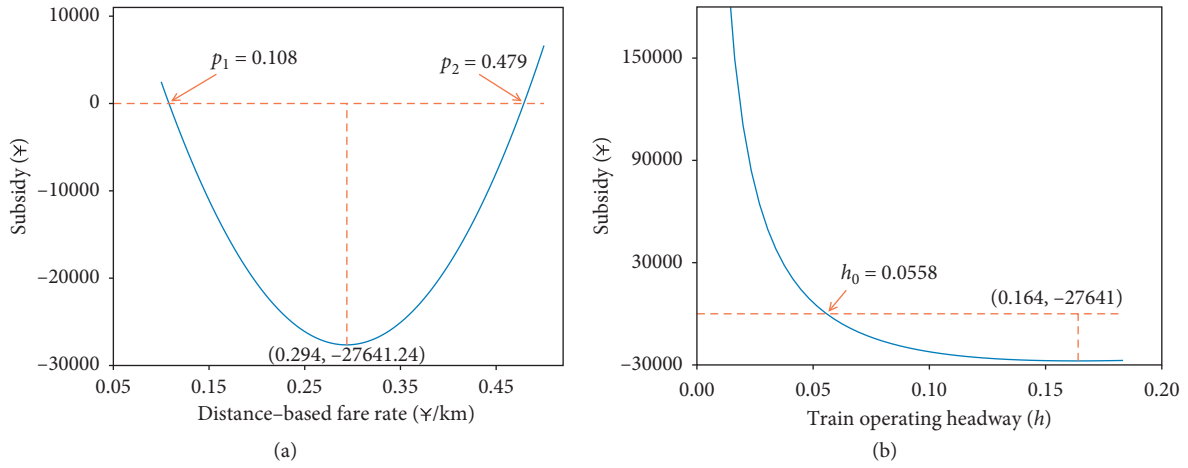


FIGURE 4: Subsidy vs. decision variables under DBF.

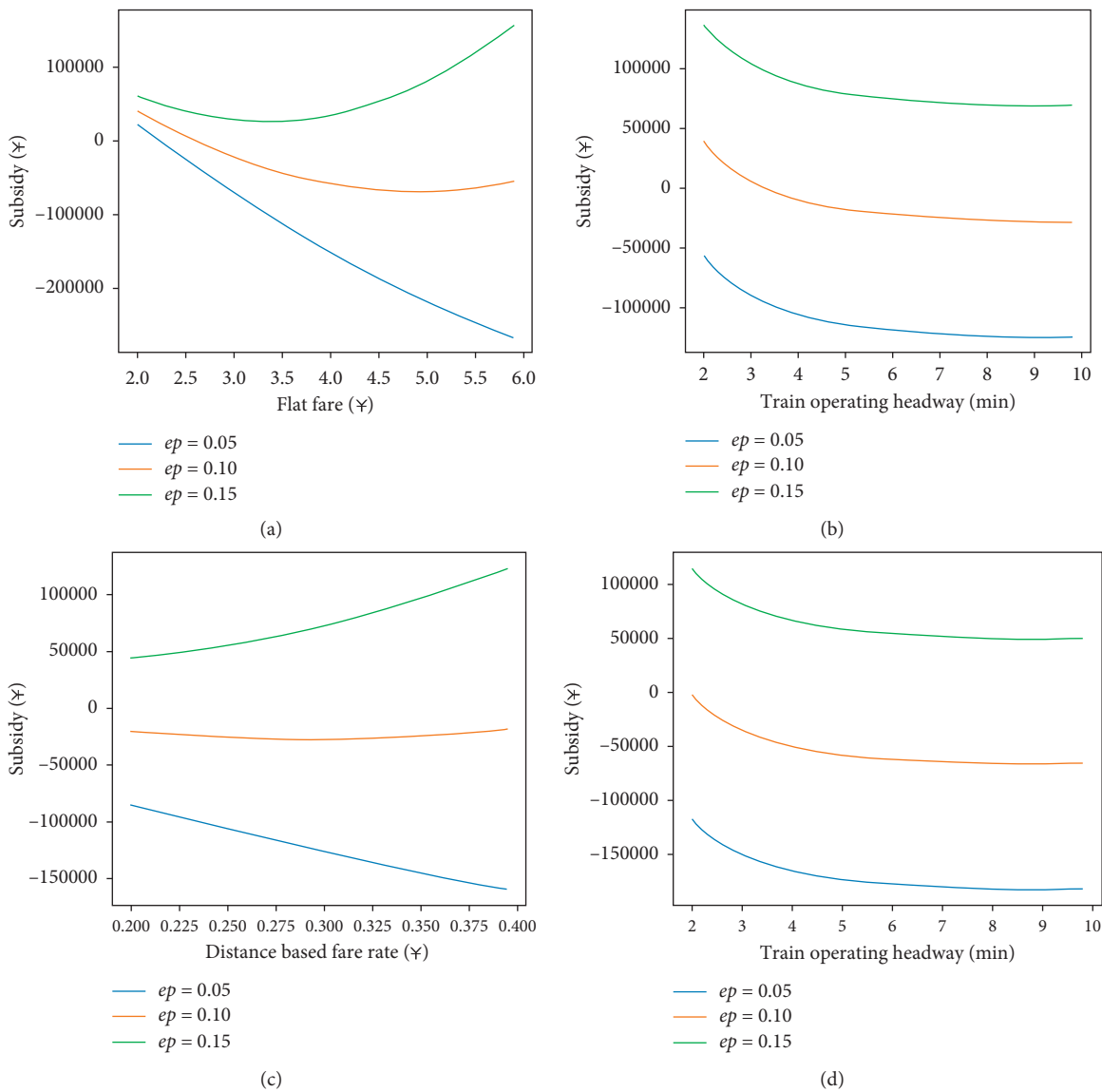


FIGURE 5: Subsidy vs. fare and headway under FF ((a) and (b)) and DBF ((c) and (d)) with three fare elasticity parameters.

TABLE 4: Optimal solutions with different fare parameters.

Optimal solution	$e_p = 0.05$		$e_p = 0.10$		$e_p = 0.15$	
	FF	DBF	FF	DBF	FF	DBF
Fare (¥)	9.64	$1.97 + 0.78d_{ij}$	4.88	$1.97 + 0.29d_{ij}$	3.3	$1.97 + 0.13d_{ij}$
Headway (minute)	5.93	7.07	8.56	9.82	10.71	11.79
Revenue (¥/h)	585051.39	447631.76	283754.21	245907.52	184112.53	173515.54
Operating cost (¥/h)	227269.12	230116.67	214949.64	218266.28	208770.53	211604.70
Operator profit (¥/h)	357782.27	217515.09	68804.57	27641.24	-24658.00	-38089.16
Passenger surplus (¥/h)	316619.46	549028.77	146131.37	225264.98	90194.69	123887.64
Social welfare (¥/h)	674401.73	766543.86	214935.94	252906.22	65536.69	85798.48
Subsidy (¥/h)	0	0	0	0	24658.00	38089.16
Demand (Pas./h)	60527	77814	57967	70989	55621	64951

the subsidy attains a minimum value of ¥ -68804.57 at a headway of 0.143 hours (8.56 minutes).

For DBF (see Figure 4), a similar trend is observed to that for FF. For a fixed optimized headway \tilde{H}^* (see Figure 4(a)), the optimized subsidy is ¥ -27641.24 when the fare is ¥ $(1.97 + 0.294d_{ij})$. When $\tilde{P}^* = E1.97 + 0.108d_{ij}$ and $\tilde{P}^* = E1.97 + 0.479d_{ij}$, the subsidy is zero. The reason is similar to that for FF. For a fixed optimized fare \tilde{P}^* (see Figure 4(a)), when the subsidy is minimal, the headway is 0.164 hours (9.82 minutes). When $\tilde{H} = 0.0558$ h, the subsidy is zero.

4.3.2. Effects of Fare Parameter on Operator Performance.

The fare levels significantly affect operator performance [7], thus affecting subsidies. Therefore, we explore the effects of fare elasticity parameter on subsidy. Figures 5(a)–5(d) compare the subsidies with the change of fare and headway for three elasticity factors: $e_p = 0.05$, $e_p = 0.10$, and $e_p = 0.15$.

Note that Figures 5(a) and 5(b) represent the changes of subsidy for different fare and headway under FF, while Figures 5(c) and 5(d) represent the changes under DBF. The effects of the fare and headway on the subsidy under FF and DBF show similar trends, but the fare and headway trends are somewhat different. It is found that a larger e_p -value requires higher subsidies. A smaller e_p -value indicates higher demand and fares, and hence a more profitable operation.

Table 4 further presents the optimized URT system solutions for three elasticity factors: $e_p = 0.05$, $e_p = 0.10$, and $e_p = 0.15$. Comparing the effects of changes in fare elasticity factor on operator performance, passenger behavior, and subsidies under FF and DBF, the fare elasticity factor has a more significant effect on fares and demand, and thus the subsidy varies more widely. For FF, the comparison shows that the FF fare decreases by 65.8%; i.e., the FF fare decreases from ¥9.64 to ¥3.3 for fare elasticity factors of $e_p = 0.05$ and $e_p = 0.15$, respectively. For DBF, the fare elasticity factor has a more significant effect on the variable component of DBF fare which decreases by 83.3%, i.e., the variable component of DBF fare decreases from ¥0.78/ d_{ij} to ¥0.13/ d_{ij} for fare elasticity factors of e_p of 0.05 and 0.15, respectively.

When e_p remains at the same level, it has a similar effect on the subsidies under the two fare structures. As e_p increases, the social welfare gradually decreases and a deficit

occurs at the lower fare and demand. Therefore, the subsidies rise from zero to ¥24658.00 and ¥38089.16 under the FF and DBF. In addition, the revenues of two fare structures differ greatly (reduced by 68.53% and 61.24%), whereas the operating cost of FF and DBF decrease by only 8.14% and 8.04%. The reason is that e_p significantly affects fares and demand, which have a greater effect on revenue than on operating cost. It is found that the subsidy is zero when e_p is 0.05 or 0.10 because if the fare and demand are high, the revenue exceeds the operating cost. The same is true for DBF.

5. Conclusions

This study focuses mainly on service design and subsidy issues and investigates subsidies to URT operators. The operator profit (OP) model primarily considers demand and train operation plan to pursue operating profit. This study extends the existing literature on operator performance and passenger characteristics under subsidy constraints by considering fare structures. The numerical examples explore the operator performance and passenger characteristics by comparing different fare structures. Operator performance is mainly measured by service level (e.g., service frequency) and operational subsidies needed from the government, while the comparative analysis under FF and DBF reflects the passengers' behavior. By analyzing the OP model, this paper provides some important findings.

- (1) This paper analyzes the operator's profit with fare structures and subsidy constraints while comparing the impact and performance of FF and DBF through proposed models and numerical results. Assuming that the passengers' travel behavior is homogenous, the results for different fare structures are slightly different (as in Table 2). The influence of decision variables on the operational subsidies under FF exceeds that under DBF (shown in Figures 3 and 4). In general, DBF is more attractive for short-distance passengers, while FF is more attractive for long-distance passengers (see Figure 1).
- (2) The subsidy is zero under the OP models in this numerical study, which means that the subsidy constraint is not binding. Profit maximization does

not require subsidies. If profit is negative when the social welfare is positive, then a subsidy may raise the profit to zero (“break-even”) but should not raise it further. Wang and Deng [12] considered the situation where the optimal profit was negative and proposed an efficiency-oriented subsidy optimization method that seeks to maximize the per capita subsidy, so there is an operating deficit in that case.

- (3) The fare structures and levels significantly affect operator performance, thus affecting subsidies to operators. Comparing the effects of changes in the fare elasticity factor on subsidies under FF and DBF, we find that a larger fare parameter value requires higher subsidies. A smaller fare parameter value indicates higher demand, so the operation is more profitable (see Figure 5 and Table 4).

Although the proposed model provides useful insights into operation and policy evaluation of URT, it neglects some important URT characteristics that should be considered in the future. In further studies, work should be pursued in the following areas.

- (1) In this paper, the subsidy is computed based on the headways and fares, which means the subsidy is treated as a financial constraint. From the perspective of management, it represents a cost-plus contract widely applied in China, in which operating losses are fully covered by the government. Future studies may consider additional policies and incentive mechanisms to induce operators to reduce operating costs and improve services.

- (2) The models presented here focus on the operator profit with subsidy constraints but ignore the effects of passenger choices and behaviors on subsidies. The current study may be extended to consider the effects of service levels and passenger travel choices on subsidies.
- (3) A linear form of elastic demand function is used in this paper, which depends on the travel times and fares (FF and DBF). The fares are the same for all passengers under FF while they vary with the trip length under DBF. In the future, fares that vary over time or for different passenger types (e.g., students, the elderly, and the disabled) may be considered.

Appendix

A. Proof Progress

To obtain the optimal solutions for the headway and fare, we set the partial derivative of the objective function $\tilde{\pi}(\tilde{H}, \tilde{P})$ (or $\hat{\pi}(\hat{H}, \hat{P})$) with respect to \tilde{H} (or \hat{H}) and \tilde{P} (or \hat{P}) to zero. For FF,

$$\begin{cases} \frac{\partial \tilde{\pi}(\tilde{H}, \tilde{P})}{\partial \tilde{P}} = \sum_{i,j \in S} q_{ij} (1 - e_w \sigma \tilde{H} - e_r r_t - 2e_p \tilde{P} + e_p \Lambda_1) = 0, \\ \frac{\partial \tilde{\pi}(\tilde{H}, \tilde{P})}{\partial \tilde{H}} = e_w \sigma \sum_{i,j \in S} q_{ij} (\Lambda_1 - \tilde{P}) + \frac{(1 + \beta \epsilon) T_R \cdot c_O + \gamma_1}{\tilde{H}^2} = 0. \end{cases} \quad (\text{A.1})$$

For DBF, we have

$$\begin{cases} \frac{\partial \hat{\pi}(\hat{H}, \hat{P})}{\partial \hat{P}} = \sum_{i,j \in S} d_{ij} q_{ij} (1 - e_w \sigma \hat{H} - e_r r_t - 2e_p (p_0 + \hat{p} d_{ij}) + e_p \Lambda_1) = 0, \\ \frac{\partial \hat{\pi}(\hat{H}, \hat{P})}{\partial \hat{H}} = e_w \sigma \sum_{i,j \in S} q_{ij} (\Lambda_1 - (p_0 + \hat{p} d_{ij})) + \frac{(1 + \beta \epsilon) T_R \cdot c_O + \gamma_1}{\hat{H}^2} = 0. \end{cases} \quad (\text{A.2})$$

Then, we obtain the optimality conditions.

For FF,

$$\begin{cases} \tilde{P}^* = \frac{\sum_{i,j \in S} q_{ij} (1 - e_w \sigma \tilde{H} - e_r r_t + \Lambda_1 e_p)}{2e_p \sum_{i,j \in S} q_{ij}}, \\ \tilde{H}^* = \sqrt{\frac{(1 + \beta \epsilon) T_R \cdot c_O + \gamma_1}{e_w \sigma (\tilde{P} - \Lambda_1) \sum_{i,j \in S} q_{ij}}}. \end{cases} \quad (\text{A.3})$$

For DBF,

$$\begin{cases} \hat{P}^* = \frac{\sum_{i,j \in S} d_{ij} q_{ij} (1 - e_w \sigma \hat{H} - e_r r_t + e_p \Lambda_1)}{2e_p \sum_{i,j \in S} q_{ij} d_{ij}}, \\ \hat{H}^* = \sqrt{\frac{(1 + \beta \epsilon) T_R \cdot c_O + \gamma_1}{e_w \sigma \sum_{i,j \in S} q_{ij} ((p_0 + \hat{p} d_{ij}) - \Lambda_1)}}. \end{cases} \quad (\text{A.4})$$

Substituting (A.3) and (A.4) into objective functions (12) or (13), and considering the constraint $\tilde{S}_{\text{flat}} = \max\{\tilde{C} - \tilde{R}, 0\}$ or $\tilde{S}_{\text{distance}} = \max\{\tilde{C} - \tilde{R}, 0\}$, we obtain the subsidy to an operator under FF and DBF as follows.

For FF,

$$\tilde{S}_{\text{flat}}^* = \max \left\{ (1 + \beta\varepsilon) \frac{c_O T_R}{\tilde{H}^*} + \left(\gamma_0 D_s + \frac{\gamma_1}{\tilde{H}^*} \right) + \Lambda_0 L_S + (\Lambda_1 - \tilde{P}^*) \sum_{i,j \in S} q_{ij} (1 - e_w \sigma \tilde{H}^* - e_r r_t - e_p \tilde{P}^*), 0 \right\}. \quad (\text{A.5})$$

For DBF,

$$\tilde{S}_{\text{distance}}^* = \max \left\{ (1 + \beta\varepsilon) \frac{c_O T_R}{\tilde{H}^*} + \left(\gamma_0 D_s + \frac{\gamma_1}{\tilde{H}^*} \right) + \Lambda_0 L_S + \sum_{i,j \in S} (\Lambda_1 - \tilde{P}^*) q_{ij} (1 - e_w \sigma \tilde{H}^* - e_r r_t - e_p (p_0 + \hat{p} d_{ij})), 0 \right\}. \quad (\text{A.6})$$

The second-order partial derivatives of $\tilde{\pi}(\tilde{H}, \tilde{P})$ (or $\hat{\pi}(\hat{H}, \hat{P})$) and with respect to \tilde{H} (or \hat{H}) and \tilde{P} (or \hat{P}) can be derived as follows. For FF,

$$\left\{ \begin{array}{l} \frac{\partial^2 \tilde{\pi}(\tilde{H}, \tilde{P})}{\partial \tilde{P}^2} = -2e_p \sum_{i,j \in S} q_{ij}, \\ \frac{\partial^2 \tilde{\pi}(\tilde{H}, \tilde{P})}{\partial \tilde{H}^2} = -2 \frac{(1 + \beta\varepsilon) T_R \cdot c_O + \gamma_1}{\tilde{H}^3}, \\ \frac{\partial^2 \tilde{\pi}(\tilde{H}, \tilde{P})}{\partial \tilde{P} \partial \tilde{H}} = \frac{\partial^2 \tilde{\pi}(\tilde{H}, \tilde{P})}{\partial \tilde{H} \partial \tilde{P}} = -e_w \sigma \sum_{i,j \in S} q_{ij}. \end{array} \right. \quad (\text{A.7})$$

For DBF,

$$\left\{ \begin{array}{l} \frac{\partial^2 \hat{\pi}(\hat{H}, \hat{P})}{\partial \hat{P}^2} = -2e_p \sum_{i,j \in S} q_{ij} d_{ij}^2, \\ \frac{\partial^2 \hat{\pi}(\hat{H}, \hat{P})}{\partial \hat{H}^2} = -2 \frac{(1 + \beta\varepsilon) T_R \cdot c_O + \gamma_1}{\hat{H}^3}, \\ \frac{\partial^2 \hat{\pi}(\hat{H}, \hat{P})}{\partial \hat{P} \partial \hat{H}} = \frac{\partial^2 \hat{\pi}(\hat{H}, \hat{P})}{\partial \hat{H} \partial \hat{P}} = -e_w \sigma \sum_{i,j \in S} q_{ij} d_{ij}. \end{array} \right. \quad (\text{A.8})$$

According to (A.7) and (A.8), we can obtain the following Hessian matrices under FF and DBF, respectively. For FF,

$$\begin{aligned} \text{Hessian}(\tilde{\pi}) &= \begin{bmatrix} \frac{\partial^2 \tilde{\pi}(\tilde{H}, \tilde{P})}{\partial \tilde{H}^2} & \frac{\partial^2 \tilde{\pi}(\tilde{H}, \tilde{P})}{\partial \tilde{H} \partial \tilde{P}} \\ \frac{\partial^2 \tilde{\pi}(\tilde{H}, \tilde{P})}{\partial \tilde{P} \partial \tilde{H}} & \frac{\partial^2 \tilde{\pi}(\tilde{H}, \tilde{P})}{\partial \tilde{P}^2} \end{bmatrix} \\ &= \begin{bmatrix} -2 \frac{(1 + \beta\varepsilon) T_R \cdot c_O + \gamma_1}{\tilde{H}^3} & -e_w \sigma \sum_{i,j \in S} q_{ij} \\ -e_w \sigma \sum_{i,j \in S} q_{ij} & -2e_p \sum_{i,j \in S} q_{ij} \end{bmatrix} \\ &= 4 \frac{(1 + \beta\varepsilon) T_R \cdot c_O + \gamma_1}{\tilde{H}^3} e_p \sum_{i,j \in S} q_{ij} - \left(e_w \sigma \sum_{i,j \in S} q_{ij} \right)^2. \end{aligned} \quad (\text{A.9})$$

For DBF,

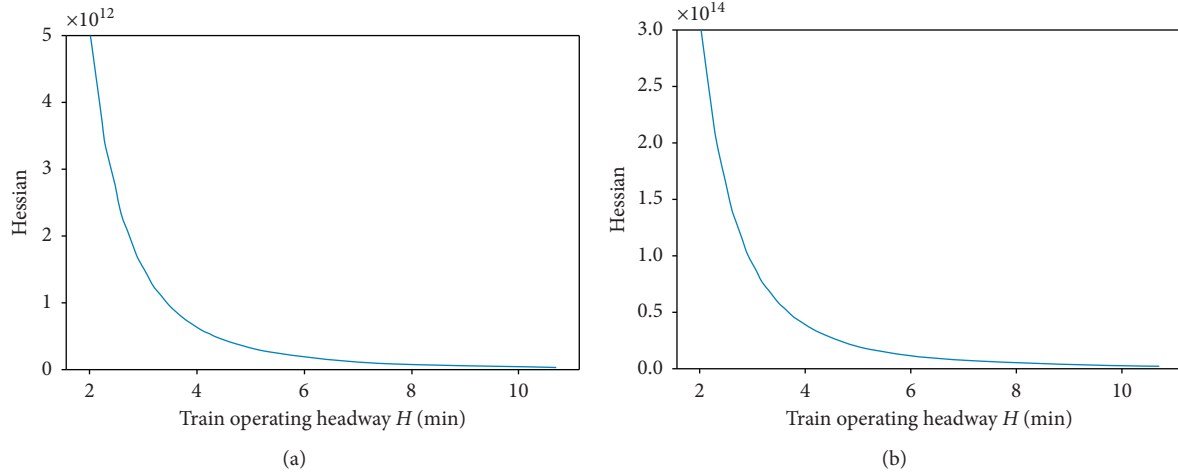


FIGURE 6: The value of Hessian matrices under the two different fare structures. (a) FF. (b) DBF.

$$\text{Hessian}(\tilde{\pi}) = \begin{bmatrix} \frac{\partial^2 \tilde{\pi}(\hat{H}, \hat{P})}{\partial \hat{P}^2} & \frac{\partial^2 \tilde{\pi}(\hat{H}, \hat{P})}{\partial \hat{H} \partial \hat{P}} \\ \frac{\partial^2 \tilde{\pi}(\hat{H}, \hat{P})}{\partial \hat{P} \partial \hat{H}} & \frac{\partial^2 \tilde{\pi}(\hat{H}, \hat{P})}{\partial \hat{H}^2} \end{bmatrix} = \begin{bmatrix} -2 \frac{(1 + \beta \epsilon) T_R \cdot c_O + \gamma_1}{\hat{H}^3} & -e_w \sigma \sum_{i,j \in S} q_{ij} d_{ij} \\ -e_w \sigma \sum_{i,j \in S} q_{ij} d_{ij} & -2e_p \sum_{i,j \in S} q_{ij} d_{ij}^2 \end{bmatrix} \quad (\text{A.10})$$

$$= 4 \frac{(1 + \beta \epsilon) T_R \cdot c_O + \gamma_1}{\hat{H}^3} e_p \sum_{i,j \in S} q_{ij} d_{ij}^2 - \left(e_w \sigma \sum_{i,j \in S} q_{ij} d_{ij} \right)^2.$$

As shown in Figure 6, all values of the Hessian matrices $\tilde{\pi}$ and $\hat{\pi}$ are greater than zero, that is, $\text{Hessian}(\tilde{\pi}) > 0$ and $\text{Hessian}(\hat{\pi}) > 0$. We can find that the Hessian matrices are negative definite through specific numerical analysis, but the sign of the Hessian matrices cannot be determined in the analytical solution. This implies that there is at least one feasible solution for operator profit (OP) models, and we can derive optimality conditions about the fare, headway, and subsidy from (A.3)–(A.6).

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 supported by the National Natural Science Foundation Council of China (nos. 71871226 and 71471179); the Graduate education reform project of Hunan Province (no. 150110005); and the China Scholarship Council (no. 201906370095).

References

- [1] China Association of Metros, *Data Source: Annual Statistics and Analysis Report of URT 2019*, China Association of Metros, Beijing, China, 2019.
- [2] R. Z. Farahani, E. Miandoabchi, W. Y. Szeto, and H. Rashidi, "A review of urban transportation network design problems," *European Journal of Operational Research*, vol. 229, no. 2, pp. 281–302, 2013.
- [3] X. Li and P. E. D. Love, "Employing land value capture in urban rail transit public private partnerships: retrospective analysis of Delhi's airport metro express," *Research in Transportation Business & Management*, vol. 32, Article ID 100431, 2019.
- [4] D. Canca, A. De-Los-Santos, G. Laporte, and J. A. Mesa, "Integrated railway rapid transit network design and line planning problem with maximum profit," *Transportation Research Part E: Logistics and Transportation Review*, vol. 127, pp. 1–30, 2019.
- [5] Y. Tang, Y. Jiang, H. Yang, and O. A. Nielsen, "Modeling and optimizing a fare incentive strategy to manage queuing and crowding in mass transit systems," *Transportation Research Part B: Methodological*, vol. 138, pp. 247–267, 2020.
- [6] I. K. Pitcher and S. G. Tesche, *A Review of Fare Structures for Melbourne, Australia*, Paper presented at the 82nd TRB Annual Meeting, Washington, DC, USA, 2003.
- [7] W. H. K. Lam and J. Zhou, "Optimal fare structure for transit networks with elastic demand," *Transportation Research*

- Record: Journal of the Transportation Research Board*, vol. 1733, no. 1, pp. 8–14, 2000.
- [8] S. I.-J. Chien and L. N. Spasovic, "Optimization of grid bus transit systems with elastic demand," *Journal of Advanced Transportation*, vol. 36, no. 1, pp. 63–91, 2002.
- [9] N. Sharaby and Y. Shiftan, "The impact of fare integration on travel behavior and transit ridership," *Transport Policy*, vol. 21, no. 4, pp. 63–70, 2012.
- [10] S. I.-J. Y. Chien and C. F. M. Tsai, "Optimization of fare structure and service frequency for maximum profitability of transit systems," *Transportation Planning and Technology*, vol. 30, no. 5, pp. 477–500, 2007.
- [11] Z. Jin, J.-D. Schmöcker, and S. Maadi, "On the interaction between public transport demand, service quality and fare for social welfare optimisation," *Research in Transportation Economics*, vol. 76, Article ID 100732, 2019.
- [12] Q. Wang and L. Deng, "Integrated optimization method of operational subsidy with fare for urban rail transit," *Computers & Industrial Engineering*, vol. 127, no. 127, pp. 1153–1163, 2019.
- [13] Y. Yang, L. B. Deng, Q. Wang, and W. L. Zhou, "Zone fare system design in a rail transit line," *Journal of Advanced Transportation*, vol. 2020, Article ID 2470579, 10 pages, 2020.
- [14] W. W. Wang, D. Z. W. Wang, H. Sun, and J. Wu, "Public transit service operation strategy under indifference thresholds-based bi-modal equilibrium," *Journal of Advanced Transportation*, vol. 50, no. 6, pp. 1124–1138, 2016.
- [15] F. M. Tsai, S. Chien, and C. H. Wei, "Joint optimization of temporal headway and differential fare for transit systems considering heterogeneous demand elasticity," *Journal of Transportation Engineering*, vol. 139, no. 1, pp. 30–39, 2012.
- [16] S. Sun and W. Y. Szeto, "Optimal sectional fare and frequency settings for transit networks with elastic demand," *Transportation Research Part B: Methodological*, vol. 127, pp. 147–177, 2019.
- [17] D. Huang, Z. Liu, P. Liu, and J. Chen, "Optimal transit fare and service frequency of a nonlinear origin-destination based fare structure," *Transportation Research Part E: Logistics and Transportation Review*, vol. 96, pp. 1–19, 2016.
- [18] C. Nuworsoo, A. Golub, and E. Deakin, "Analyzing equity impacts of transit fare changes: case study of Alameda-Contra Costa Transit, California," *Evaluation and Program Planning*, vol. 32, no. 4, pp. 360–368, 2009.
- [19] S. R. Jara-Díaz and A. Gschwender, "The effect of financial constraints on the optimal design of public transport services," *Transportation*, vol. 36, no. 1, pp. 65–75, 2009.
- [20] L. J. Basso and H. E. Silva, "Efficiency and substitutability of transit subsidies and other urban transport policies," *American Economic Journal: Economic Policy*, vol. 6, no. 4, pp. 1–33, 2014.
- [21] K. Gkritza, M. G. Karlaftis, and F. L. Mannering, "Estimating multimodal transit ridership with a varying fare structure," *Transportation Research Part A: Policy and Practice*, vol. 45, no. 2, pp. 148–160, 2011.
- [22] C. D. Nassi and F. C. d. C. d. Costa, "Use of the analytic hierarchy process to evaluate transit fare system," *Research in Transportation Economics*, vol. 36, no. 1, pp. 50–62, 2012.
- [23] Y. Zhou, H. S. Kim, P. Schonfeld, and E. Kim, "Subsidies and welfare maximization tradeoffs in bus transit systems," *The Annals of Regional Science*, vol. 42, no. 3, pp. 643–660, 2008.
- [24] Q. Wang, L. B. Deng, and G. M. Xu, "Operational subsidy optimization in urban rail transit under the break-even mode: considering two fare regimes," *Computers & Industrial Engineering*, vol. 149, 2020.
- [25] Y. Sun, Q. Guo, P. Schonfeld, and Z. Li, "Implications of the cost of public funds in public transit subsidization and regulation," *Transportation Research Part A: Policy and Practice*, vol. 91, pp. 236–250, 2016.
- [26] S. Ling, N. Jia, S. Ma, Y. Lan, and W. Hu, "An incentive mechanism design for bus subsidy based on the route service level," *Transportation Research Part A: Policy and Practice*, vol. 119, pp. 271–283, 2019.
- [27] Y. Sun and P. M. Schonfeld, "Optimization models for public transit operations under subsidization and regulation," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2530, no. 1, pp. 44–54, 2015a.
- [28] G. Kocur and C. Hendrickson, "Design of local bus service with demand equilibration," *Transportation Science*, vol. 16, no. 2, pp. 113–260, 1982.
- [29] China Railway Fourth Survey and Design Institute Group Co., Ltd, *Feasibility Study Report on the First Phase Project of Changsha Rail Line 2 [R]*, China Railway Fourth Survey and design Institute Group Co., Ltd., Wuhan, China, 2009.