**Coordinating last-train timetabling with app-based ride-hailing service under uncertainty**

Jia Ning1,2, Xinjie Xing3, Yadong Wang4, Yu Yao5, Liujiang Kang6, Qiyuan Peng1,\*

*1School of Transportation and Logistics, Southwest Jiaotong University, Sichuan, China*

*2Department of Technology and Data Intelligence, Jingdong Zhenshi Information Technology Co. Ltd., Beijing, China*

*3Management School, University of Liverpool, Liverpool, United Kingdom*

*4School of Economics & Management, Nanjing University of Science & Technology, Jiangsu, China*

*5College of Civil & Transportation Engineering, Hohai University, Jiangsu, China*

*6State Key Laboratory of Rail Traffic Control & Safety, Beijing Jiaotong University, Beijing, China*

\* Corresponding author. E-mail addresses: qiyuan-peng@swjtu.edu.cn

Abstract

Since urban rail transit (URT) service is normally not running on 24-hour operation in most cities, last-train timetabling is a prominent problem and challenges URT managers constantly. The rise of app-based ride-hailing (ARH) service opens up new opportunities and challenges for last-train operators to better serve late-night passengers. Specifically, when passengers cannot reach their destinations only through URT services during the last-train operation period, passengers could make good use of feasible train-to-train transfers to reach stations closer to their destinations, and transfer to flexible ARH services to reach their final destinations. However, uncertain road conditions and varying passenger travel preferences complicate the coordination of URT services with ARH services. By considering different passengers’ traveling preferences, various travel path choices, and uncertain ARH travel times, we formulate a two-stage mixed-integer stochastic optimization model to achieve an optimal last-train timetable design for getting more passengers to their destinations in a cost-effective and efficient way. In addition, we propose a genetic algorithm-based solution strategy which outperforms commercial solvers with its computational performance and has its practicability assured. Through our numerical experiments, we reveal insights about how different customers’ preferences and cost components affect the optimal results and provide operational suggestions accordingly for achieving better timetable performance.

Keywords

Last train timetabling; Ride-hailing services; Passenger path choice; Uncertainty; Genetic algorithm; Stochastic programming

1. Introduction

With the rapid development of the urban rail transit (URT) system, the proportion of public transport passenger flow shared by urban rail transit has gradually increased. However, URT services in most cities do not provide 24-hours services and will be closed from late night to early morning for system maintenance and operating cost savings. According to statistics, in 2022, the average daily service time of URT systems in mainland China is 17 hours, of which Beijing has the longest average service time of 18.7 hours per day. In this case, if the last train timetables of different URT lines are not coordinated at the transfer station, passengers may not be able to reach their destinations by last URT trains. Therefore, how to coordinate the last train timetables to meet the travel needs of more late-night passengers is one of the crucial issues in URT operations. However, it is difficult to transport *all* passengers to their destinations *only* through URT trains during last train operation period, especially for large URT networks [1]. Therefore, URT operators’ focus is shifting from independent planning to more effective coordination with other nighttime urban transportation modes to better serve passengers.

The rise of app-based ride-hailing (ARH) service has tremendously reshaped the landscape of urban transportation, and meanwhile, brought new opportunities and challenges to last-train operations. According to the ARH industry research report (https://www.chinairn.com), as of the end of 2022, China has 298 online ARH platforms (e.g., Uber and Didi Dache), with an average daily order volume of up to 30 million orders. Passengers are encouraged to use smartphone apps for hailing taxis or sharing private vehicles in real time or in advance so their travel requirements can be met more flexibly. In addition, ARH's 24-hour operation service ensures to a certain extent that passengers traveling late at night have transportation available. However, since ARH is much more expensive and less safe (especially for women) than URT, late-night passengers usually give priority to taking URT, especially long-distance passengers. The above operational and service characteristics of ARH and URT have brought opportunities for the cooperation between the two to further improve the service quality for late-night passengers, that is, a more flexible travel option through the combination of URT services and ARH services. Particularly, if passengers cannot reach their destinations by URT trains, they can first take URT trains to stations closer to their destinations, and then transfer to ARH services to reach their destinations.

Although the cooperation between ARH and URT has brought tremendous convenience and flexibility to passengers, it also inevitably increases operational difficulties for the operation of the last train:

*First*, the purpose of the last train timetable optimization is to optimize the feasibility of passengers transferring between any two last trains at any transfer station, in order to transport more passengers to their destinations through URT services. However, it is difficult to realize that all transfers in the URT network are feasible due to the limitation of the URT network topology [1]. It can further divert passengers who fail to transfer to road transport especially when ARH service is much more accessible. How to optimize the feasibility of train-to-train transfers in order to meet the needs of more passengers and attract more passengers to take URT services, is a challenge for URT operators.

*Second*, due to the visibility of real-time traffic information offered by ARH service, passengers’ behaviors are complicated by the large variety of possible path combinations even for just one origin-destination (OD) pair. For example, passengers may choose URT only, ARH only, or URT-ARH joint traveling with transfer at any point, so it increases the difficulties for URT operators to map out an accurate customer flow when performing timetabling optimization.

*Third*, individual passengers can have different traveling preferences with respect to different transportation modes. Namely, even if a passenger has an accessible candidate path to his/her destination through URT services but needs to take a long detour, he/she may take an ARH service instead or transfer to an ARH service halfway if he/she is more time-sensitive and such paths can take him/her to his/her destination faster. The proportion of such ‘exceptional’ passengers can be large as the estimated travel time is visible to passengers underground from time to time, so they can change their minds at any minute.

*Fourth*, although URT managers can also access the information from ride-hailing apps to support their last train timetabling for better coordination, such information may not 100% reliable especially since road conditions are often more unpredictable during late night. For example, more road works are taken place during the night; the end of major events in a city can cause huge congestion; temporary road inspections by policy can slow down traffic flows. In addition, the spatial and temporal distribution of available vehicles (i.e. ARH services) is also hard to predict and significantly affects passenger routing. To coordinate last train timetable design with ARH service, those uncertainties can significantly jeopardize the practicability and operational performance of the optimized plan.

To address the above-discussed challenges, this research proposes a stochastic mixed-integer formulation to optimize the performance of last train timetabling with ARH service as a substitution when uncertain road traffic and diversified passenger preferences are considered. Through the proposed research process, this paper aims at contributing to 1) construct an effective mathematical model that is able to effectively capture the operational details and constraints for URT and ARH coordination during the last-train operation period; 2) design effective and efficient algorithm to solve the problem and demonstrate its practicability; 3) reveal managerial insights to support URT managers’ decision-making and accommodate more passengers’ needs in late-night.

The remainder of this paper is organized as follows. Section 2 highlights the academic significance of this paper through a systematic literature review in the relevant field. Section 3 describes the underlying problem in more detail. Section 4 describes passenger candidate path types and the generation process. Section 5 constructs the stochastic mixed-integer formulation accordingly. Section 6 discusses and develops the solution strategy. Section 7 conducts a series of numerical experiments to demonstrate the effectiveness of our model and solution method, and reveal managerial insights. Section 8 wraps up the key findings and briefs the future work of this study.

1. Literature review

The last train timetabling problem aims to coordinate the arrival and departure times of different last trains to provide passengers with effective transfers, which is an important practical issue for URT companies that cannot provide 24-hour services. With the construction and expansion of URT networks in major cities around the world, the last train timetabling problem has received much attention in recent years.

* 1. Last train timetabling under deterministic environment

The majority of existing research on the last train timetabling are based on deterministic setting. The earlier research on the last train timetabling problem focused on the station transferability within pure URT networks, for example, Guo et al. [2], Kang and Meng [3], Kang et al. [4–8], Kang and Zhu [9], Nie et al. [10], Wang et al. [11], Yu et al. [12], Zhang et al. [13], Zhou et al. [14]. They assumed passenger path choices were fixed and thus the number of transfer passengers at each station was fixed. Based on such assumptions, they aimed to optimize the last train timetable in order to minimize the total transfer waiting times, maximize the number of feasible transfers, or maximize the number of passengers who can transfer smoothly. However, in practice, passengers will adjust their path choices when the last train timetable is modified [15,16]. To address this shortcoming, some researchers began to study the last train timetabling problem from the perspective of destination reachability, for example, Chen et al. [15], Long et al. [17], Ning et al. [1], Wang et al. [18], Yang et al. [19], Yao et al. [20], Zhou et al. [16]. In these papers, the last train timetable and passenger path choices were optimized simultaneously, with the purpose of maximizing the number of passengers who can reach their destinations within the pure URT network. For simplicity, most of them assumed that passengers with the same origin and destination will choose the same path. That is, they were more concerned about whether there is a destination-reachable path for each passenger, ignoring the diversity of passengers’ preference for path choices and/or the actual travel time and travel cost of each candidate path for passengers. To provide late-night passengers with higher-quality travel services, Zhou et al. [21] formulated passenger path choices by a Logit model with considering the ticket price, the total passenger travel time, and passengers’ waiting time at their origin stations. Then, they designed a genetic algorithm to optimize the departure time of last URT trains, where station dwell times and section running times were considered to be fixed. Yao et al. [20] observed that passengers are willing to take some detours to reach their destination via URT trains. Incorporating the detour routing strategy, they developed a bi-level optimization model, where the upper level was to optimize the last train timetable, and the lower level was to determine passenger path choices.

The majority of existing studies mainly focus on optimizing the last train timetable within pure URT networks. However, as one of the important modes of urban transportation, there is a need to coordinate with other transportation to provide passengers with better services. To this end, Long et al. [17] and Huang et al. [22] focused on improving the smoothness of transfers between intra-city traffic and inter-city traffic late at night by optimizing the last URT timetable. Besides, to improve the convenience of passengers traveling within the city late at night, Ning et al. [1], Huang et al. [23], Kang et al. [7], and Ning et al. [24] focused on making full use of transfer connections between the last URT train and other intra-city traffic as a substitute for those non-feasible train-train transfers. Specifically, Ning et al. [1] (i.e. one of our previous research) considered the coordination of the last URT train with other nighttime urban transportation modes by minimizing the total remaining travel distance for passengers from stations where they failed to transfer to their final destinations. However, how such unreachable passengers reach their final destinations after leaving the URT system has not been considered in Ning et al. [1]. Considering the existing bus network, Huang et al. [23] developed a mixed-integer linear programming (MILP) model to synchronically optimize the timetables of subway and bus networks based on the determined passenger distribution obtained from a user equilibrium assignment model. However, traditional buses would also be closed late at night, even earlier than URT services (e.g., in most cities of China). Hence, taking traditional buses as the substitute for last-train transfer failures may not be widely applicable. Kang et al. [7] proposed to customize bus bridging services for passengers who fail to transfer between the last URT trains. Assuming that passengers’ path choices were fixed, they presented a MILP model to integrate the optimization of the last train timetable and bus bridging service design problem. However, utilizing existing bus routes or designing customized bus routes to bridge URT trains still cannot cover the entire city for passengers’ door-to-door transportation requirements, especially considering the high operating costs of operating buses. In this sense, incorporating private transportation (i.e., ARH services) offers the possibility. For this reason, Ning et al. [24] (i.e., one of our previous research) developed a bi-objective mixed-integer nonlinear programming (MINLP) model for integrated optimization of last train timetabling and bridging service design, where both taxis and buses were considered. The results show that the cooperation between last URT trains and ARH has a significant effect on reducing passenger travel time and travel cost. As an extension to Ning et al. [24], this paper further explores the impact of uncertain road conditions and varying passenger travel preferences on the cooperation between URT services and ARH services.

* 1. Last train timetabling under uncertain environment

To the best of our knowledge, only a few papers have investigated the last train timetabling problem in uncertain environments, including uncertain transfer passengers and uncertain transfer walking times. However, these papers implemented last train timetable optimization under the independent operation of URT and did not consider passengers’ path choices. Specifically, Yang et al. [25] and Yang et al. [26,27] addressed the last train timetabling problem under the uncertainty of transfer passengers. Yang et al. [26] adopted the sample-based representation to model uncertain transfer demands and formulated a mean-variance utility-based model to determine the last train timetable. Subsequently, by using the same sample-based representation, the max-min reliability criterion, and percentile reliability criterion, Yang et al. [27] developed a non-linear model to maximize the number of successful transfer passengers and minimize the total running time of all the last trains. Yang et al. [25] developed a novel distributionally robust chance-constrained programming model to enhance the robustness of the last train timetabling in terms of maximizing the total number of successful transfer passengers. Chen et al. [28] synchronized the arrival and departure times of the last URT trains with considering heterogeneous transfer walking time. They represented the heterogeneity of transfer walking time by a given probability distribution (e.g., the log-normal distribution) and established a bi-objective non-linear model to balance between the maximal transfer accessibility and the minimal extension of dwell time.

* 1. The contributions of this paper

To the best of our knowledge, only Ning et al. [24] explored the cooperation between the last URT trains and private transportation modes (i.e., ARH services). This paper is an extension of Ning et al. [24] and aims to further improve the practical applicability of research findings. Specifically, this paper explores the cooperation between last URT trains and ARH services taking into account uncertain road conditions and different passenger travel preferences. Due to the high flexibility of ARH, this extension increases tremendous difficulties in model formulation and algorithm design. On the one hand, passengers have a large number of candidate paths, including taking the URT only, taking the ARH only, or taking a combination of URT and ARH (possibly transferring to ARH at any station). Passengers with the same origin and destination may choose different paths based on their preferences. On the other hand, passengers could prefer different path choices under different road conditions. This requires the same last train timetable to perform well under different road conditions, that is, to ensure that the paths favored by passengers are feasible as much as possible.

For clarity, Table 1 also shows a comparison of our research to papers in the relevant field. The main contributions of this paper are summarized as follows.

* We develop a two-stage stochastic optimization model to effectively capture the operational details and constraints for last URT trains and ARH coordination, where uncertain road conditions and diverse passenger travel preferences are considered. To our knowledge, this is the first study on the cooperation between last URT trains and ARH services with the consideration of uncertain environment.
* A genetic algorithm-based solution strategy is proposed to solve the problem which includes an excellent chromosome representation method and a feasibility processing module to effectively improve solution efficiency and solution quality.
* The effectiveness of our proposed model and algorithm is demonstrated on a real-world URT network, and some interesting findings are revealed to provide suggestions for URT operations and passenger travelling.

**Table 1.** Comparisons of the most relevant studies.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Paper | Passenger path choice | Passengers’ preference | Passenger travel time/cost | Cooperation with other transportation | Uncertainty |
| Zhou et al. [21] | √ | √ | √ | / | / |
| Yao et al. [20] | √ | × | √ | / | / |
| Kang et al. [7] | × | × | × | Bus bridging | / |
| Yang et al. [25] and Yang et al. [26,27] | × | × | × | / | Passenger demand |
| Chen et al. [28] | × | × | × | / | Transfer walking time |
| Ning et al. [24] | √ | × | √ | ARH services, Bus bridging | / |
| Our research | √ | √ | √ | ARH services | ARH travel time |

1. Problem Description

This study focuses on the coordination between ARH services and URT services. ARH services are operated by a number of private individuals. Passengers are encouraged to make reservations or place orders at any point in a day through app-based online platforms. Once the platform receives a job demand, it distributes the demand to a group of potential drivers to let them decide if the order will be taken. After a driver confirmed the job, the estimated pickup time and the total journey time will normally be given to the passenger via the platform. Typically, ARH services are flexible with 24-hour operations (although the number of operating vehicles would decrease late at night) and location availability. However, since late at night, a large number of ARH services are also approaching closing time, and passenger demand also decreases, drivers would pay more attention to minimizing the empty movement (i.e., without taking any passengers) when taking orders. Consequently, for passengers whose origin stations are located in remote areas and/or whose destinations are extremely far from their origin stations, it may be difficult to find available ARH services. Meanwhile, when passengers travel long distances, their travel costs of using AHR services are quite high. Moreover, due to highly uncertain road conditions (e.g., road congestion, inclement weather, number of vehicles available, and distance of the service vehicle from the passenger’s origin, etc.), ARH services can fluctuate significantly in travel times which may sometimes be unexpectedly longer than traveling by URT trains.

In contrast, URT services are much more reliable with their travel times and the associated timetables are often determined in advance by one or several URT companies. Once the train timetable is established, it will direct the running of all trains for a relatively long period (e.g., several months), although it may vary on certain dates (e.g., weekends or holidays). However, for nighttime maintenance and reduced operating costs, most URT systems are not running on a 24-hour basis but need to decide the service closure time for each URT line. Under such circumstances, the last URT trains of a day will be the final chances for late-night passengers to reach their destinations through URT services. Passengers may fail to transfer at transfer stations when their connecting trains are the last trains and have already departed from the transfer stations. In such cases, passengers would be forced to choose other alternative modes of transportation to continue to their destinations. Therefore, passengers usually plan their travel paths based on published last train timetables, their own travel preferences, and other alternative travel options before traveling.

When ARH service provides an alternative option, passengers can rely on ARH services to reach their destinations if they are not accessible for the last URT trains. However, such a solution can also complicate passengers’ travel patterns as passengers may not only use ARH services for unreachable destinations (via URT trains) but for other travel needs. For example, some passengers may have accessible candidate paths to their destinations via URT trains but require long detours. In contrast, it may be a better option for these passengers to opt for intermediate transfers to ARH services which could reduce travel times but also entail higher travel costs. Passengers expect to be more knowledgeable about road conditions in order to decide the better transfer stations for getting ARH services according to their requirements and preferences. Accordingly, under the coordination of URT services and ARH services, passengers’ travel paths can be full URT, full ARH, or URT-ARH coordination at a certain station. When it comes to the design of last-train timetabling, URT managers need cost-effectively and efficiently to transport as many passengers as possible to their intended destinations through good coordination with ARH services. Since passengers may transfer to ARH services at any URT station, the well-coordinated train timetable should transport passengers to their favored URT stations, which can be passengers’ destination stations or the stations at which passengers plan to transfer to ARH services.

To address the above problem, we formulate a two-stage stochastic optimization model, combining the last URT train timetabling and passenger path selections considering passenger's individual preferences and uncertain ARH travel times. Particularly, a last-URT timetable is determined in the first stage and passengers will make their travel path selections accordingly in the second stage. To support our model formulation, the following assumptions are proposed and justified:

**Assumption 1.** The late-night passenger demand is given and fixed, including their origin stations, destination stations, and the number of passengers. In other words, we do not consider that late-night passenger demand fluctuates with the travel environment.

This assumption is used to simplify problem complexity [7,15–17,19]. Besides, Li et al. [29] proposed a method to forecast the late-night passenger demand based on taxi FCD data, URT automatic fare collection (AFC) data, and bus smart card swipe data.

**Assumption 2.** The passenger transfer walking time is known and fixed.

This assumption is also used to simplify problem complexity. In reality, transfer walking times may vary from passenger to passenger due to different walking speeds [5,6,9]. As in the existing literature [15,22], we set the transfer walking time a bit longer than average to make sure it works for the majority of passengers.

**Assumption 3.** The capacity of the last URT train service is sufficient to carry all passengers. In other words, we do not take the capacity of the URT train into account.

This assumption is reasonable because the late-night passenger demand is relatively low [5,7,9,15,16,19,22].

1. Preliminary – passenger path generation

The two-stage stochastic optimization model is established based on pre-constructed sets of passenger candidate paths. Therefore, before the model formulation, we first construct a set of candidate paths for each OD pair of passengers, where indicates the set of passenger OD pairs.

With the coordination between ARH services and URT services, there are three types of passenger candidate paths (as illustrated in Figure 1): i) **ARH path**: a passenger takes an ARH service from the origin directly to the destination. ii) **URT path**: a passenger takes URT services from the origin directly to the destination. During the journey, the passenger may transfer from one URT line to another. iii) **Joint path**: a passenger takes URT services from the origin to a URT-ARH transfer station, and then transfers to an ARH service to the destination. During the journey using URT services, passengers may make multiple transfers between different URT lines. Notably, it is possible that a passenger can use ARH-URT (i.e., ARH first then transfer to URT) coordination to reach the destination, but we do not include this option in our path candidates. This is because road conditions can be very uncertain so that it is unsure if passengers taking ARH services can arrive on time to catch the last connecting train. In addition, we consider passengers whose origin stations are covered by URT but whose destination stations may or may not be covered by URT. For passengers whose destination stations are not covered by URT, their candidate paths only include ARH paths and joint paths (i.e., without URT paths).



**Figure 1.** Three types of passenger candidate paths.

We further introduce the path set with each path denoted by a sequence of stations (i.e., the origin station, transfer stations, and departure station). The ‘departure station’ here refers to the station where passengers depart from the URT network. Thus, the departure station can be passengers’ intended destination (for a URT path), passengers’ origin station (for an ARH path), and a URT-ARH transfer station where passengers will transfer from URT service to ARH service (for a joint path).



**Figure 2.** An example of generating candidate paths for passengers traveling from station s1 to station s9 (only last URT trains are considered as the serviced trains).

For any OD pair (denoted by ), where represents the origin station and represents the destination station, its associated path set is constructed by the following process (an example with six candidate paths is illustrated in Figure 2):

*Step 1. Generate an ARH path.*

An ARH path is generated by taking ARH service from origin station directly to destination station . We consider only one ARH path for each OD pair which connects station to station within the public road network, such as path 1 in Figure 2. Notably, the travel time of the ARH path can fluctuate with uncertain road conditions.

*Step 2. Generate URT paths.*

First, the *k* shortest path algorithm [30] is adopted to generate at most *k* URT physical paths connecting station to station within the URT network. Then, for each directional URT line along each URT path, both the last train and several non-last trains are considered as possible service trains. For example, in Figure 2, for passengers who aim to travel from station *s1* to station *s9*, two URT paths (i.e. paths 2 and 4) are generated in total if the last train is the only option. But if both the last train and the penultimate train are considered, twelve URT paths (i.e., four URT paths that go through the same directional URT lines as path 2 but use different train services, eight URT paths that go through the same directional URT lines as path 4 but use different train services) can be generated.

*Step 3. Generate joint paths.*

Along each generated URT path, passengers may fail to transfer at each transfer station. Thus, based on each generated URT path, for each transfer station , we generate one joint path along which passengers will take the URT train from station to station and then transfer to the ARH service to station . The number of joint paths generated based on each URT path is equal to the number of transfers contained in the URT path. For example, in Figure 2, one joint path (i.e. path 3) is generated based on URT path 2, and two joint paths (i.e. paths 5 and 6) are generated based on URT path 4. Notably, passengers can actually transfer from URT services to ARH services at any URT station, not just transfer stations. However, along a given URT path, the accessibility of any station located between any two adjacent transfer stations is the same as that of the subsequent transfer station. For example, along URT path 4, the stations located between the transfer stations and have the same accessibility as station . Therefore, in order to control the number of passenger candidate paths, we only consider the transfer stations as representative stations to generate joint paths.

1. Model formulation

Since the regularity of road conditions is not obvious, however, it is convenient to characterize road conditions through scenario enumeration, we propose a two-stage stochastic optimization model to address the last train timetabling problem with ARH service coordination under uncertain road conditions. In the first stage, the model is used to optimize the timetable of each last URT train along each directional URT line. Once the last train timetable has been determined, the feasibility of each train-train transfer and the accessibility of each passenger candidate path are also confirmed in the first stage. Different passenger candidate paths have different travel costs and travel times under different road conditions. Only reachable candidate paths can be selected by passengers. In the second stage, based on the accessibility of each candidate path determined in the first stage, the model is used to decide passenger path selections concerning passenger preferences and uncertain road conditions (i.e., uncertain travel times of ARH services). The two stages interact through the accessibility of passenger candidate paths. The objective function of the model is to minimize the expected total generalized travel cost of all passengers. In the following, we firstly define all the notations in Section 5.1. Then we describe the relevant constraints of the first stage and the second stage in Sections 5.2 and 5.3, respectively. Finally, the objective function of the model is given in Section 5.4.

* 1. Notations

The notations to be used in the model are listed as follows.

|  |  |
| --- | --- |
| ***Sets*** | |
|  | Set of directional URT lines, indexed by or . |
|  | Set of stations on directional URT line , indexed by . , where station and represent the start and terminal stations, respectively. |
|  | Set of all trains on directional URT line , indexed by and . , where and represent the last and penultimate trains, respectively. |
|  | Set of train-train transfers. Each transfer is indexed by , representing that passengers transfer from train to train at station . |
|  | Set of OD pairs, indexed by , where and represent the origin station and the destination station of passengers, respectively. |
|  | Set of pre-generated candidate paths for passengers of OD pair , indexed by . |
|  | Set of transfers contained in path . . |
|  | The entire populations of travel times of ARH services. A sample process of ARH travel time is denoted by . |
| ***Parameters*** | |
|  | The maximum running time from station to station along directional URT line : . |
|  | The minimum running time from station to station along directional URT line : . |
|  | The maximum dwell time at station along directional URT line : . |
|  | The minimum dwell time at station along directional URT line : . |
|  | The minimum safety headway between successive arrivals of trains on directional URT line : . |
|  | The minimum safety headway between successive departures of trains on directional URT line : . |
|  | The minimum safety headway between the departure of the preceding train and the arrival of the following train on directional URT line : . |
|  | The original arrival time of train at station on directional URT line : , . |
|  | The original departure time of train at station on directional URT line : , . |
|  | The allowable latest service closure time for directional URT line : . |
|  | The transfer walking time associated with transfer : . |
|  | The number of passengers of OD pair : . |
|  | The travel time for a passenger on URT service along path : , . |
|  | The realized travel time for a passenger on ARH service along path under scenario : , , . |
|  | The travel cost of URT service for a passenger along path : . |
|  | The travel cost of ARH service for a passenger along path : , . |
|  | The realized generalized travel cost for a passenger of OD pair along path under scenario : , , . |
|  | The realized minimum generalized travel cost among all candidate paths for a passenger of OD pair under scenario : , . |
|  | An auxiliary parameter, which is equal to . |
|  | , a vector of the random variables representing the road conditions under scenario . |
|  | A non-negative number between zero and one, reflecting how knowledgeable a passenger is about his/her candidate paths that he/she will be traveling on. |
|  | The weight of travel time in the generalized travel cost. |
|  | A sufficiently small positive number (less than 1). |
|  | A sufficiently large positive number. |
| ***Decision variables*** | |
|  | Continuous variable indicating the arrival time of train at station on directional URT line : , . |
|  | Continuous variable indicating the departure time of train at station on directional URT line : , . |
|  | Binary variable indicating whether transfer is feasible: . If yes, ; otherwise, . |
|  | Binary variable indicating whether path is destination-reachable: , . If yes, ; otherwise, . |
|  | , which denotes all the decision variables in the first stage. |
|  | Continuous variable indicating the number of passengers choosing path with given sample process : , , . |
|  | Auxiliary variable for linearization:. |
|  | , which denotes all the decision variables in the second stage. |

* 1. The first stage – last train timetabling

The first stage is formulated to determine the timetable of the last train of each directional URT line, the feasibility of each train-train transfer, and the reachability of each passenger candidate path.

The URT network in a city consists of a number of bi-directional URT lines. Each bi-directional URT line consists of two URT directional lines. Trains can only operate along a specific bi-directional URT line, not across different bi-directional URT lines. An example in Figure 3 is given to describe the train operation process along a bi-directional URT line. That is, a rolling stock runs from one terminal station to another terminal station along a specific direction as a train service, and stops at each station along the way for passengers to get on and off. Then the rolling stock turns around at the terminal station, and runs in the opposite direction of the same URT line as another train service. The above operation process is repeated until the service of the rolling stock ends. Additionally, trains do not meet or overtake each other due to the restrictions of facilities. Accordingly, the following constraints (i.e. Eqs. (1)-(8)) are given to describe such operation.

Since trains cannot cross different URT lines, passengers whose origin and destination stations are located on different URT lines need to transfer between different URT lines (i.e. different trains) at the transfer station. The feasibility of each train-train transfer depends on the arrival time of the feeder train and the departure time of the connecting train (as shown by Eq. (9)). Furthermore, the accessibility of each passenger candidate path depends on whether all the transfers it contains are feasible (as shown by Eq. (10)). Therefore, the feasibility of train-train transfers and the accessibility of passenger candidate paths will be determined in the first stage with the determination of the last train timetables.



**Figure 3.** An example of train operation on a bi-directional URT line.

*Constraint 1:* the dwell time of each last train at each station should be within an appropriate range to ensure the quality of passenger service.

|  |  |
| --- | --- |
|  | (1) |

where and are decision variables, representing the arrival and departure times of the last train at station along line , respectively. and are given parameters, representing the minimum and maximum dwell times at station along line .

*Constraint 2:* the running time of each last train between two adjacent stations should be within an appropriate range to ensure the safety of train operation and the quality of passenger service.

|  |  |
| --- | --- |
|  | (2) |

where and are given parameters, representing the minimum and maximum running times from station to station along line .

*Constraint 3*: as illustrated by Figure 4, safety headways (i.e., headway between successive arrivals , headway between successive departures , and headway between departure and arrival ) should be satisfied between the last train and the penultimate train of the same directional URT line.

|  |  |
| --- | --- |
|  | (3) |
|  | (4) |
|  | (5) |

where and indicate the last and penultimate trains along line , respectively.



**Figure 4.** Illustration of safety headways.

*Constraint 4*: the service closure time of each directional URT line should not be later than the latest allowable closure time due to the need for system maintenance and lower operational costs.

|  |  |
| --- | --- |
|  | (6) |

where indicates the terminal station along line . is a given parameter, representing the allowable latest service closure time of line .

*Constraint 5:* the timetable of non-last trains is fixed to the original timetable.

|  |  |
| --- | --- |
|  | (7) |
|  | (8) |

where and represent the original arrival and departure times of train at station along line .

*Constraint 6*: as illustrated in Figure 5, a transfer is feasible only if the departure time of the connecting train is later than the arrival time of the feeder train plus the corresponding transfer walking time of passengers at this transfer station.

|  |  |
| --- | --- |
|  | (9) |

where is a binary decision variable, which is equal to 1 if the transfer is feasible. is a given parameter, representing the transfer walking time of passengers associated with transfer . is a sufficiently large positive value and can be determined by the method proposed by Kang and Meng [3].



**Figure 5.** Illustration of the feasible transfer and the infeasible transfer.

*Constraint 7*: a passenger candidate path is destination-reachable only if all transfers contained in the path are feasible; otherwise, at least one transfer is infeasible.

|  |  |
| --- | --- |
|  | (10) |

where is a binary decision variable, which is equal to 1 if path is destination-reachable. indicates the set of all transfers contained in path . is a sufficiently small positive number (less than 1).

* 1. The second stage – passenger assignment

The second stage is formulated to determine passenger path selections. Passengers can only choose the accessible paths identified in the first stage, and path selections are performed according to the travel time and travel cost of each accessible candidate path. Due to the uncertainty of road conditions, the travel time of each path can be different under different road conditions. Some existing advanced traffic simulation models (e.g., Zeng et al. [31,32] and Qian et al. [33]) help to characterize different road conditions and estimate the corresponding ARH travel times. We model the uncertain road conditions as a finite set of scenarios, denoted by . For each scenario and each path , we introduce as the generalized travel cost along path under scenario :

|  |  |
| --- | --- |
| , | (11) |

where and represent the travel cost of URT service and ARH service when a passenger chooses path , respectively. and respectively represent the travel time of URT service and ARH service when a passenger chooses path . Among them, is scenario-dependent. is the weight of travel time in the generalized travel cost, which can be determined through questionnaire surveys and data mining.

Passengers of the same OD pair may choose different paths according to their preferences, knowledge of the current road conditions, and estimated travel time and travel cost via the ARH platform. Thus, we formulate passengers’ path choice behavior using a multinomial logit choice model, which has been proven to be useful in capturing passengers’ travel behavior and modeling their path choices [34].

*Constraint 8*: passenger path selections. The number of passengers choosing path under scenario can be calculated by Eq. (12). Note that when a candidate path is unreachable (i.e. ), the number of passengers choosing path under any scenario is equal to zero. Recall that the accessibility of each passenger candidate path is determined in the first stage.

|  |  |
| --- | --- |
|  | (12) |

where is a continuous variable, representing the number of passengers choosing path under scenario . is a given parameter, representing the total number of passengers of OD pair . is a given continuous parameter that fluctuates between 0 and 1. The value of reflects how knowledgeable a passenger is about his/her candidate paths that he/she will be traveling on, which can be determined through on-the-spot investigations, questionnaire surveys, and data mining. When he/she is very knowledgeable about his/her candidate paths (i.e., very large ), information such as actual road conditions, last train timetables, transfer feasibility at different stations, accessibility of each candidate path, and rich historical traveling experience are all available to him/her. With the given ARH estimated times, such a passenger has sufficient information to decide the most cost-effective path to reach his/her destination. Conversely, when the value of is very small, this passenger knows very little about his/her candidate paths except for the published train timetable and ARH estimated time. In this case, it is difficult for him/her to judge which candidate paths are destination-reachable or/and which stations in the URT network can be reached only according to the last train timetables. Since this passenger has no other information to support his/her decision-making, knowing the ARH estimated time is also not enough to find the best path as it cannot be compared with other alternative candidate paths. Therefore, each destination-reachable candidate path shares the same probability to be selected. Also, we use to denote the minimum generalized travel cost among all candidate paths for a passenger of OD pair under scenario , which is calculated by:

|  |  |
| --- | --- |
|  | (13) |

In addition, since Eq. (12) is a non-linear constraint (where is a continuous decision variable, and is a binary decision variable), we linearize the equation by the following procedure. First, we introduce an auxiliary parameter to replace , so that Eq. (12) is rewritten as Eq. (14). Since , and are all known parameters, is a known parameter.

|  |  |
| --- | --- |
|  | (14) |

We introduce an auxiliary variable to replace , which is a non-negative continuous variable. The constraints on the value of are established as follows:

|  |  |
| --- | --- |
| , | (15) |
| . | (16) |

Accordingly, Eq. (12) can be rewritten with in a linear manner:

|  |  |
| --- | --- |
| . | (17) |

where .

* 1. Objective

The objective of the two-stage stochastic optimization model is to minimize the expected total generalized travel cost of all passengers.

|  |  |
| --- | --- |
|  | (18) |
|  | (19) |

1. Solution strategy

Considering the stochasticity of our proposed problem formulation, we adopt sample average approximation (SAA) with the support of a genetic algorithm (GA) to solve the two-stage stochastic optimization model in this section.

* 1. Sample Average Approximation

Sample Average Approximation is regarded as an effective solution to address stochastic programming problems. In SAA, the expected value function is approximated by the sample average function , where , …, are random samples of the random vector . A random sample corresponds to one realization of the actual ARH travel times. By increasing the sample size , the solutions obtained by the SAA will converge to the optimal solution of the original two-stage stochastic programming model [35]. Besides, all decision variables in the second stage (i.e., ) become sample-dependent decision variables (i.e.,) in the SAA. Accordingly, the proposed two-stage stochastic optimization model is rewritten as:

**SAA model:**

|  |  |
| --- | --- |
|  | (20) |

subject to

|  |  |
| --- | --- |
| Eq. (1) – (10) | (21) |
| for | (22) |

Eq. (20) is the objective function to minimize the average of total generalized travel costs related to different ARH travel time realizations. Eq. (21) copies the constraints from Eqs. (1) – (10) in the original two-stage stochastic optimization model. Eq. (22) aims to determine the passenger assignment, which consists of replications of Eqs. (11), (13), (15) – (17).

**Table 2.** The number of decision variables and constraints in the SAA model

|  |  |  |
| --- | --- | --- |
|  | # decision variables | # constraints |
| The first stage |  |  |
| Per sample in the second stage |  |  |

This model is a mixed-integer linear programming model and can be solved directly by commercial solvers. However, the scale of the model increases by increasing the number of samples. The sample size should be large enough to ensure that the sample average function is close enough to . In Table 2, we give the number of decision variables and constraints in the SAA model. Among them, the decision variables and constraints in the second stage (associated with passenger assignment) need to be replicated for each sample in the SAA. The number of decision variables and constraints in the second stage depends on the number of passenger candidate paths, which is quite large for large-scale URT networks. Accordingly, for large-scale URT networks, the scale of the SAA model may increase dramatically due to a large number of passenger paths and required samples. Under such circumstances, solving the SAA model directly with commercial solvers may not yield the optimal solution in a reasonable computational time. Alternatively, we develop a GA-related solution to efficiently solve the SAA model.

* 1. Genetic algorithm-based solution

We employ a GA-based solution to solve the SAA model for the following reasons. The biggest difficulty in solving the SAA model is a large number of decision variables and constraints associated with passenger assignment. However, once the train timetables have been determined, the accessibility of each passenger candidate path is determined. Then, the results of passenger assignment under each sample and the value of the objective function (i.e., the average of total generalized travel costs among all samples) can be easily obtained by calculating Eqs. (12) and (20). Therefore, the nature of the decision variables from this study offers an easy way to code them as chromosome representations of GA’s populations.

The solution process of the GA is described in Figure 6, which mainly includes chromosome representation, population initialization, population evaluation, GA operators, termination conditions, and algorithm running parameter settings.



**Figure 6.** The solution process of genetic algorithm.

* + 1. Chromosome representation and population initialization

A chromosome, made up of genes, indicates a solution of the SAA model. As described in Section 6.1, the decision variables of the SAA model include the train timetable (i.e., and ), the feasibility of each transfer (i.e., ), the accessibility of each passenger candidate path (i.e., ), and the number of passengers choosing each path under each sample (i.e., ). Among them, only the decision variables () are chosen as the genes for any chromosome in the GA. The reasons are as follows:

On the one hand, when the transfer feasibility (i.e., ) is determined, the accessibility of each passenger candidate path (i.e., ) and the number of passengers choosing each path under each sample (i.e., ) can be determined according to Eqs. (10) and (12). On the other hand, the transfer feasibility (i.e., ) is determined by the train timetable (i.e., and ) according to Eq. (9). Different train timetables may yield the same transfer feasibility. However, the objective function only includes the decision variables . Although the last train timetable does affect the accessibility of passenger candidate paths (i.e., ), and thus the number of passengers choosing each path under each sample (i.e., ), small differences in the last train timetable, which do not affect the feasibility of transfers, will not affect the value of the objective function. Thus, the binary decision variables (), which represent the feasibility of transfers, are chosen to be the representative for the solution of the SAA model and are used to form the chromosome base in the proposed GA-based solution. In this way, the number of genes in each chromosome is equal to the number of transfers, i.e., . The total number of possible chromosomes in the GA is equal to .

Furthermore, based on a given chromosome (i.e. the value of ), the last train timetable (i.e., and ) can be determined by solving the model consisting of Eqs. (1) – (9). However, it is possible that no feasible train timetable can be obtained based on a given chromosome. The feasibility of each generated chromosome is temporarily ignored during initialization and is modified to be feasible during fitness evaluation (see Section 6.2.2). In GA, the initial population (i.e., the set of chromosomes) is randomly generated, and a total of *N* chromosomes are generated during initialization.

Compared to choosing the continuous decision variables and as the genes, which is commonly used in the existing literature of the last train timetabling problem (such as Chen et al. [15], Kang et al. [5], Nie et al. [10], Wang et al. [18], Yao et al. [20], Zhou et al. [21]), it is easier to converge by choosing the binary decision variables as the genes. This is because the latter can avoid searching for numerous solutions with the same objective value, and the number of possible chromosomes (or solutions) is relatively small.

* + 1. Fitness evaluation
       1. *Chromosome feasibility processing*

Each chromosome is denoted by , where indicates the value of under the chromosome . As mentioned earlier, it is possible that no feasible train timetable can be obtained based on a given chromosome. Specifically, some transfers cannot be feasible at the same time due to train operation restrictions. For example, Kang and Meng [3] indicates that, concerning a pair of transfers between two trains at the same transfer station (e.g., the transfer from the last train of Line -down to the last train of Line -down at station , and the transfer from the last train of Line -down to the last train of Line -down at station ), only one transfer at most can be feasible when the transfer walking time is strictly greater than the train dwell time at this transfer station. Such situations are overlooked when chromosomes are generated, resulting in infeasible chromosomes. Therefore, regarding any chromosome from either initialization or other GA operations (i.e., crossover and mutation), we need to assure its feasibility before calculating its fitness.

To this end, for each generated chromosome, denoted by , we establish the Chromosome Feasibility (CF) Model and the Chromosome Improvement (CI) Model and solve them in turn. Specifically, we first solve the CF model with the chromosome as the input. The solution of the CF model forms a new chromosome, denoted by . Then, we solve the CI model with the chromosome as the input. The solution of the CI model forms another new chromosome, denoted by . Since chromosomes and are both feasible chromosomes, the one with the better fitness will replace the original chromosome in the population to perform the subsequent GA operations (i.e., selection, crossover, and mutation). To further clarify the above, we present a flow chart of the process in Appendix.

**Chromosome Feasibility (CF) Model:**

|  |  |
| --- | --- |
|  | (23) |

subject to

|  |  |
| --- | --- |
| Eq. (1) – (9) | (24) |

**Chromosome Improvement (CI) Model:**

|  |  |
| --- | --- |
|  | (25) |

subject to

|  |  |
| --- | --- |
| Eq. (1) – (9) | (26) |
|  | (27) |

The idea to establish the two models is described as follows. The CF model is developed to make the chromosome feasible on the basis of ensuring the characteristic of the chromosome . The key characteristic of a chromosome is to indicate which transfers are feasible. Thus, the objective of the CF model is to maximize the number of feasible transfers that are also feasible in the chromosome . After the CF model is executed, the CI model is developed to increase the number of feasible transfers as much as possible on the basis of ensuring the original feasible transfers in the chromosome . We do this because in general, the more transfers are feasible, the more passenger paths can be destination-reachable, which may further improve the fitness value or the objective function value. The objective of CI model is to maximize the number of feasible transfers among all transfers. Additional constraints (27) are embedded in the CI model to ensure that transfers that are feasible in the chromosome are still feasible in the new chromosome .

* + - 1. *Fitness function*

The quality of each chromosome is assessed by the fitness. Here, we consider three different fitness functions, given by Eqs. (28), (29), and (30), respectively. Among them, Eq. (28) is a common fitness function used for minimization optimization models. Eq. (29), referred to as the accelerated fitness function, can amplify the difference between any two chromosomes compared to Eq. (28). Eq. (30) takes the idea of simulated annealing into account, where denotes the current temperature and will be decreased with the generations by the cooling factor . When is high, the chromosomes of poor quality are likely to be retained to increase the genetic diversity; when is low, the GA will be accelerated to converge. Besides, if the up-to-now best solution is not improved in several generations, will be updated to the initial temperature to jump out of the local optimum. The application performance of these three fitness functions will be demonstrated and compared in Sections 7.2 and 7.3.1.

|  |  |
| --- | --- |
|  | (28) |
|  | (29) |
|  | (30) |

where is the fitness of the chromosome . is the value of the objective function (i.e., Eq. (20)) under the chromosome . and respectively represent the maximal and minimal objective values among all chromosomes in the current population.

* + 1. GA operators – selection, crossover, and mutation

The GA operators used is similar to the standard GA. The selection operator is used to select a set of *N* chromosomes as the parents for the next generation. We adopt the roulette wheel selection method combined with the elite strategy [15] as the selection operator. The crossover and mutation operators are used to generate *N* child chromosomes of the next generation. The crossover operator is used to combine two parents to generate two child chromosomes with a given probability . The two-point crossover (as shown in Figure 7) is selected as the crossover operator here. The mutation operator is used to provide genetic diversity and escape local optimum. Each chromosome is randomly mutated with a given probability . Since the genes in the chromosome are all binary variables, the basic bit mutation is selected as the mutation operator here.



**Figure 7.** Crossover operation.

* + 1. Termination condition and algorithm running parameter settings

The GA terminates when the maximum number of generations is reached or the best solution does not change after a given number of generations. The basic parameters of the GA include: the population size *N*, the crossover probability , the mutation probability , the number of genes mutated in each chromosome , the maximum number of generations , and the maximum number of generations with no improvement in the objective/fitness value . The value of these parameters may differ with URT networks. The GA should be tested to calibrate its parameters to balance solution quality and computational time.

1. Numerical experiments

This section is designed to illustrate the application of the proposed model and solution strategy based on a real-world URT and road transport network (i.e., Chengdu URT and road network, in China). The algorithms were coded in C# and compiled with Visual Studio 2019. GUROBI 9.1.2 is used for exact solutions and all experiments were carried out on a laptop with AMD Ryzen 7 5800H @ 3.20 GHz and 13.9 GB RAM.

* 1. URT Network description and experiment settings

The Chengdu URT network (April 2019), as depicted in Figure 8, consists of 7 bi-directional URT lines and 156 URT stations. For each URT line, its downward direction is indicated by an arrow next to the line name. We collected the original train timetable and the basic train operation restrictions (i.e., the minimum safety headways, the maximum and minimum train running/dwell times) from Chengdu URT corporation. According to the original train timetable, among all URT lines, the earliest closure time was 23:14:09 (i.e., Line 10-down), and the latest closure time was 00:12:46 (i.e., Line 3-down). In numerical experiments, the closure time of each directional URT line was allowed to be delayed by at most 10 minutes compared with the original timetable.

Based on the Chengdu URT network, we first designed a small-scale case and a large-scale case to benchmark the computational performance of our proposed solution strategy. The difference between the small-scale case and the large-scale case lies in the number of passenger OD pairs considered. For the small-scale case, we picked 6 origin stations (marked by red triangles in Figure 8) and 6 destination stations (marked by green squares in Figure 8), which form 36 OD pairs overall. These OD pairs were selected based on a travel demand analysis of the late-night passengers and actual AFC data (obtained from the Chengdu URT corporation). More specifically, it is observed that the key routes for passengers traveling at late night are from workplaces (e.g., station TFWJ), entertainment venues (e.g., stations TFGC and CDZ), intercity railway stations (e.g., stations HCB and CDDKZ) or airports (e.g., station HZL2) to residential areas (e.g., stations XP, LQY, CDYXY, SLX, WS, and WNC). This observation is further verified by the actual AFC data. Passenger demand of each OD pair is generated randomly within the range from 1 to 20 based on the analysis of the actual passenger demand of the last trains on the Chengdu URT network in April 2019.



**Figure 8.** Map of the Chengdu URT network.

For the large-scale case, we consider all possible OD pairs within the Chengdu URT network. Inspired by existing relevant studies [16], we merged some OD pairs to reduce the problem scale. Specifically, only one URT station is selected between every two adjacent transfer stations or between a terminal station and its adjacent transfer station. This selected station can be seen as a representative of stations in the same area. For example, for a group of stations between station XP (i.e., the terminal station of Line 2-down) and station YPTX (i.e., the adjacent transfer station of station XP), passengers traveling from the same origin to any station of this group will choose the same candidate paths and go through the same transfers. In addition, passengers whose origin and destination stations are located on the same URT line are excluded from our numerical experiments as they are deemed to reach their destinations by URT trains as long as they can catch the last URT train at their origin stations. Consequently, 644 OD pairs in total are considered in the large-scale case. In addition, since every actual station has arriving and departing passengers, when we aggregate several stations into a representative station, the number of arrivals and departures for this representative is taken from the summation of the corresponding numbers for each integrated station.

Passenger candidate paths are generated through the procedure introduced in Section 4, including ARH paths, URT paths, and joint paths. For the small-scale case, a total of 398 passenger candidate paths are generated, including 36 ARH paths, 170 URT paths, and 192 joint paths. For the large-scale case, a total of 6,726 passenger candidate paths are generated, including 644 ARH paths, 3,014 URT paths, and 3,068 joint paths. Along each path, passengers’ travel cost associated with URT trains (i.e., ) is determined by its corresponding travel distance. Such cost information is accessible from the website <https://www.chengdurail.com/index.html>. Passengers’ travel time between different stations with trains (i.e., ) is assumed to be constant and can be obtained from the same website. Passengers’ travel cost with ARH service (i.e., ) is also determined by the travel distance of the passenger spending on road. We assumed that the ARH service always picks the shortest path between a given origin station and a destination station under the guide of Sat Nav. The travel cost of ARH service for each path can be obtained from its online service platforms. Finally, the passenger travel time on ARH service (i.e., ) is a random variable that varies with respect to road conditions. The shortest ARH travel time was obtained from online service platforms. Unless otherwise stated, the actual ARH travel time was considered to be extended from 0 to 30 minutes (with uniform distribution) based on the shortest ARH travel time.

Unless otherwise specified, the weight of travel time in the generalized travel cost was set to 1.0, the coefficient in the multinomial logit choice model was set to 0.8, and the relevant parameters of the GA were set as follows: the population size *N*=50, the crossover probability =0.98, the mutation probability =0.5, the number of genes mutated in each chromosome , the initial temperature = 3000, the cooling factor =0.9, the maximum number of generations =100, and the maximum number of generations with no improvement in the objective/fitness value =30. The above values settings of parameters were determined through multiple executions of GA in order to balance solution quality and computational time.

* 1. The computational performance benchmarking with the small-scale case

Using the small-scale case, the performance of the GA was benchmarked with a commercial solver (i.e., GUROBI). Specifically, based on the small-scale case, a series of SAA models with different sample sizes were developed. For each sample size, we benchmarked the computational time and the objective value gap between GUROBI and the proposed GAs (with three different fitness functions). The detailed results are reported in Table 3.

**Table 3.** Results of the small-scale case.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| # sams | GUROBI solver | | | |  | GA | | | | | | | | |
| # vars | # cons | Time (s) | Opt obj |  | # gens | | |  | Time (s) | | | Best obj | Opt gap |
|  |  |  |  |  |  |  |  |
| 5 | 25,204 | 72,195 | 22 | 28677.90 |  | 32 | 33 | 50 |  | 103 | 122 | 179 | 28677.90 | 0.00% |
| 10 | 49,084 | 140,295 | 66 | 28795.87 |  | 32 | 33 | 50 |  | 111 | 116 | 181 | 28795.87 | 0.00% |
| 20 | 96,844 | 276,495 | 249 | 28760.19 |  | 32 | 33 | 50 |  | 116 | 120 | 179 | 28760.19 | 0.00% |
| 40 | 192,364 | 548,895 | 1,169 | 28812.86 |  | 32 | 33 | 50 |  | 113 | 128 | 180 | 28812.86 | 0.00% |
| 60 | 287,884 | 821,295 | 2,814 | 28816.56 |  | 32 | 33 | 50 |  | 113 | 123 | 184 | 28816.56 | 0.00% |
| 80 | 383,404 | 1,093,695 | 5,525 | 28823.54 |  | 32 | 33 | 50 |  | 118 | 124 | 181 | 28823.54 | 0.00% |
| 100 | 478,924 | 1,366,095 | 12,828 | 28812.80 |  | 32 | 33 | 50 |  | 114 | 124 | 184 | 28812.80 | 0.00% |
| **#sams** – number of samples; **#vars** – number of decision variables; **#cons** – number of constraints;  **#gens** – number of generations; | | | | | | | | | | | | | | |

As can be seen from Table 3, a linear increase in sample size can result in the exponential growth of the computational time for using the solver directly. When the sample size reached 100 samples, the solver took up to 3.5 hours to obtain the optimal solution.

Comparatively, the use of a GA-based solution can dramatically reduce the computational time without compromising solution quality. In particular, even for the largest sample size, our proposed GA solution can solve the problem within 4 minutes regardless of which fitness function was employed. In addition, the results also imply that the computational time of the GA-based solution increased very minorly despite a 20-fold increase in the number of samples (i.e. from 5 samples to 100 samples). The reason for that is, the computational time of the GA-based solution is irrelevant to the sample size, but mainly depends on the number of generations and the population size of each generation. The most time-consuming step in the GA-based solution is to solve the CF model and CI model to ensure the feasibility of each chromosome.

Recall that is the simplest fitness function, is the accelerated fitness function, and is the one considering simulated annealing. In contrast, the number of generations when using or was less than when using in the small-scale case. The performance of these three fitness functions in terms of solution quality and computational time will be further compared in the next section.

* 1. The performance on the large-scale case

Using the large-scale case, we analyzed the performance of the GA-based solution with different fitness functions in Section 7.3.1. We then analyzed the effectiveness of the embedded CI model in Section 7.3.2.

* + 1. The benchmarking of the three fitness functions

To further test the performance of the GA-based solution, we generated a series of large-scale instances with different sample sizes. Notably, GUROBI was excluded from the following experiments as its direct use was not capable of handling the large-scale case even with only 5 samples. When considering 5 samples, the number of decision variables was 425,268, and the number of constraints was 1,202,223. The optimality gap was 4.66% after the solver ran for 10 hours. The GA-based solution with three different fitness functions was applied to solve these instances respectively. The results are reported in Table 4. The best solution obtained from the three fitness functions is highlighted in bold. In addition, we also benchmarked the economic performance between the original train timetable with the best solution (i.e., the one in bold) obtained from our proposed GA-based solutions as well.

Results in Table 4 show that the GA-based solution is also capable of handling large-scale cases within a reasonable timeframe (i.e., no more than 6 minutes and 70 generations). The best solutions obtained from our proposed GA-based solutions were able to yield more than a 5% cost reduction compared with the use of the original timetable.

Moreover, results in Table 4 also show that the use of (i.e., the accelerated fitness function) outperformed the other two fitness functions as it returns the best objective values. Furthermore, the number of generations from our proposed GA-based solution with was more stable than using the other two functions with respect to different sample sizes. Therefore, it is selected as the solution for further numerical experiments.

**Table 4.** Results of the large-scale case under different fitness functions.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| # sams |  | | |  |  | | |  |  | | |  | Original | Obj impro |
| Time (s) | # gens | Best obj (\* 105) |  | Time (s) | # gens | Best obj (\* 105) |  | Time (s) | # gens | Best obj (\* 105) |  | Obj  (\* 105) |
| 5 | 186 | 37 | 24.71 |  | 270 | 51 | **24.44** |  | 343 | 67 | 24.71 |  | 25.84 | 5.41% |
| 10 | 176 | 35 | 24.69 |  | 289 | 51 | **24.41** |  | 231 | 43 | 24.69 |  | 25.80 | 5.40% |
| 20 | 216 | 42 | 24.71 |  | 311 | 51 | **24.43** |  | 288 | 52 | 24.47 |  | 25.83 | 5.39% |
| 40 | 249 | 45 | 24.68 |  | 326 | 51 | **24.41** |  | 310 | 52 | 24.44 |  | 25.80 | 5.40% |
| 60 | 221 | 36 | 24.68 |  | 342 | 51 | **24.41** |  | 260 | 45 | 24.68 |  | 25.80 | 5.40% |
| 80 | 244 | 37 | 24.68 |  | 336 | 51 | **24.41** |  | 290 | 47 | 24.44 |  | 25.80 | 5.38% |
| 100 | 261 | 40 | 24.68 |  | 353 | 51 | **24.40** |  | 309 | 48 | 24.68 |  | 25.79 | 5.38% |
| **#sams** – number of samples; **#gens** – number of generations; **Original** – the objective value under the original timetable; **Obj impro** – the percentage improvement in the best objective function value resulting from GA-based solution, compared to the original timetable. | | | | | | | | | | | | | | |

* + 1. The performance of CI model

Recall that the purpose of the CI model is to further improve the fitness value (or the objective value) of a given chromosome (or a solution). To verify the effectiveness of the CI model, we employed the same instances as in Section 7.3.1. For each instance, the GA-based solution was applied with and without the CI model to solve the SAA formulation. The accelerated fitness function was also used for implementing the GA-based solution. The results are reported in Table 5, where “Obj impro” refers to the improvement in the objective value with the CI model compared to without the CI model.

**Table 5.** Comparison with and without CI model.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| # samples | Without CI model | | |  | With CI model | | | Obj impro |
| Time (s) | # gens | Best obj (\* 105) |  | Time (s) | # gens | Best obj (\* 105) |
| 5 | 117 | 38 | 24.47 |  | 270 | 51 | 24.44 | 0.10% |
| 10 | 115 | 38 | 24.44 |  | 289 | 51 | 24.41 | 0.13% |
| 20 | 132 | 38 | 24.46 |  | 311 | 51 | 24.43 | 0.10% |
| 40 | 137 | 38 | 24.44 |  | 328 | 51 | 24.41 | 0.12% |
| 60 | 130 | 38 | 24.44 |  | 342 | 51 | 24.41 | 0.12% |
| 80 | 137 | 38 | 24.44 |  | 336 | 51 | 24.41 | 0.12% |
| 100 | 142 | 38 | 24.43 |  | 353 | 51 | 24.40 | 0.12% |
| **#gens** – number of generations; **Obj impro** – the percentage improvement in the best objective function value resulting from the use of CI model. | | | | | | | | |

As indicated by the results, the use of the CI model was able to further bring the cost down by 0.1% on average across all sample sizes. However, it is also noted that the use of the CI model resulted in a nearly three-fold increase in computational time over the non-CI model solution, suggesting that the advantage of the CI model may only be significant when the problem scale and sample size are relatively large.

* 1. The benefits of considering uncertain road conditions

In this section, upon the large-scale case with 100 samples, we demonstrate the benefits of considering the uncertainty of road conditions. To this end, we adopted the model proposed by Ning et al. [24] for last-train timetable optimization, in which the travel time and travel cost of ARH services were set to the minimum among all considered scenarios. The obtained results were compared with those obtained considering uncertain road conditions, as shown in Table 6.

**Table 6.** Results under different values of .

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Uncertainty | Travel time (\*103 h) | | |  | Travel cost (\*104 RMB) | | |  | # passengers | | | # feasible transfers |
| Total | ARH | URT |  | Total | ARH | URT |  | ARH path | URT path | Joint path |
| Y | 26.68 | 13.32 | 13.36 |  | 86.71 | 74.68 | 12.03 |  | 4857 | 8696 | 15216 | 83 |
| N | 27.72 | 14.89 | 12.83 |  | 94.00 | 82.43 | 11.57 |  | 4,915 | 7,981 | 15,873 | 78 |
| **Y:** the uncertainty of road conditions **is** considered; **N:** the uncertainty of road conditions **is not** considered. | | | | | | | | | | | | |

By comparison, if the uncertainty of road conditions is not considered, the feasibility of transfers between different last trains under the optimized train timetable cannot meet the actual needs of passengers. This results in higher total travel time and total travel costs for passengers, and reduced usage of URT services.

* 1. Sensitivity analysis

In this section, upon the large-scale case with 100 samples, we perform a series of sensitivity analyses to evaluate the impacts of parameters and on the overall optimization.

* + 1. The impacts of

Recall that the value of reflects the level of passengers’ traffic knowledge. Particularly, it indicates how much unknown information that a passenger can harness to find the best path. In between 0 to 1, the higher the value is, the higher possibility this passenger can harness all available information (i.e., visible information such as ARH journey estimation and published train timetable, and invisible information such as personal historical traveling experience) to choose the most cost-effective paths. Conversely, when a passenger has very little knowledge about the overall traffic of a certain OD pair (e.g., ), only visible information will be used for path choice. Namely, despite the ARH platform offering travel time estimation, it is still hard to know the difference amongst different candidate paths as this passenger has no other knowledge to compare and make decisions. To understand how the value of affects the overall operation and coordination between URT and ARH, a series of experiments were carried out with respect to different values of (i.e., 1.0, 0.8, 0.6, 0.4, 0.2, 0.0). The results are shown in Figure 9 and Table 7.



**Figure 9.** Results under different values of .

As Figure 9 above indicated, while decreased, both the total passenger travel time and the total passenger travel cost increased. Particularly, a group of fully non-knowledgeable passengers spent nearly 1,140 hours more in travel times and 169,000 RMB more in travel costs than full-knowledgeable passengers. This implies that, in order to further optimize the performance of the last train timetable, URT managers can further consider solutions such as broadcasting or advertising the real-time information of the whole transport network (e.g., estimated transfer time at each station, transferability of subsequent stations, any road congestion, etc.) as comprehensive as possible. Therefore, passengers are informed to be knowledgeable and rational, which makes coordination between URT and ARH more effective. In addition, this can also lead to an increase in the use of URT services. As shown in Table 7, the greater the number of knowledgeable passengers (i.e., the value of increased from 0 to 1), the more passengers preferred URT services (i.e., the number of passengers choosing URT paths increased from 5,433 to 9,590, and the travel costs for URT services increased from 106,300 RMB to 124,300 RMB).

**Table 7.** Results under different values of .

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Travel time (\*103 h) | | |  | Travel cost (\*104 RMB) | | |  | # passengers | | | # feasible transfers |
| Total | ARH | URT |  | Total | ARH | URT |  | ARH path | URT path | Joint path |
| 1.0 | 26.50 | 12.54 | 13.96 |  | 82.73 | 70.30 | 12.43 |  | 4398 | 9590 | 14780 | 81 |
| 0.8 | 26.68 | 13.32 | 13.36 |  | 86.71 | 74.68 | 12.03 |  | 4857 | 8696 | 15216 | 83 |
| 0.6 | 26.91 | 13.98 | 12.92 |  | 89.92 | 78.22 | 11.71 |  | 5257 | 7951 | 15560 | 83 |
| 0.4 | 27.13 | 14.71 | 12.43 |  | 93.28 | 81.93 | 11.35 |  | 5704 | 7128 | 15936 | 86 |
| 0.2 | 27.38 | 15.46 | 11.91 |  | 96.73 | 85.75 | 10.97 |  | 6197 | 6287 | 16284 | 108 |
| 0.0 | 27.64 | 16.20 | 11.44 |  | 99.66 | 89.03 | 10.63 |  | 6604 | 5433 | 16731 | 110 |

In addition, as shown in Table 7, as decreased (i.e. the passenger’s knowledge of the candidate path he/she will travel on decreased), the number of feasible transfers used by passengers in the URT network increased instead. This is because passengers have less information about candidate paths, and the spatial distribution of passenger path selection is more scattered. This indicates that as decreased, the usage of URT services decreased, but the number of transfers that need to be maintained as feasible in actual operation increased (i.e., less revenue for URT companies, but greater operating pressure for URT operators).

* + 1. The impacts of under different road conditions

Recall that is the weight of travel time in the cost measure, which reflects the importance of travel time in terms of travel cost for late-night passengers. With a large value of , passengers are more sensitive to the total duration of their journey. To understand how affects the overall coordination and optimization, we solved the problem with different values of (i.e., 1.0, 2.0, 3.0, 4.0, 5.0) based on the same group of realized scenarios. In addition, experiments in this section further consider the impact of different road conditions. Namely, the road is running by normal traffic, where the travel time of ARH services is at most 30 minutes delay than the shortest travel time (with uniform distribution), and the other type is running by congested traffic flow, which has at least 30 to 60 minutes delay than the shortest travel time (with uniform distribution) with ARH services. The results are shown in Table 8.

**Table 8.** Results under different values of and different road condition.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Road condition |  | Travel time (\*103 h) | | |  | Travel cost (\*104 RMB) | | |  | # passengers | | | # feasible transfers |
| Total | ARH | URT |  | Total | ARH | URT |  | ARH path | URT path | Joint path |
| Normal | 1.0 | 26.68 | 13.32 | 13.36 |  | 86.71 | 74.68 | 12.03 |  | 4,857 | 8,696 | 15,216 | 83 |
| 2.0 | 26.59 | 14.13 | 12.46 |  | 92.24 | 80.81 | 11.43 |  | 5,507 | 7,669 | 15,592 | 82 |
| 3.0 | 26.56 | 14.42 | 12.14 |  | 94.15 | 82.95 | 11.20 |  | 5,789 | 7,334 | 15,646 | 82 |
| 4.0 | 26.55 | 14.57 | 11.98 |  | 95.16 | 84.08 | 11.08 |  | 5,942 | 7,158 | 15,668 | 81 |
| 5.0 | 26.48 | 14.86 | 11.62 |  | 96.84 | 85.99 | 10.85 |  | 6,212 | 6,935 | 15,621 | 79 |
| Congested | 1.0 | 35.34 | 21.14 | 14.20 |  | 81.32 | 68.70 | 12.62 |  | 4,215 | 10,777 | 13,776 | 117 |
| 2.0 | 35.56 | 22.33 | 13.22 |  | 86.75 | 74.79 | 11.96 |  | 4,954 | 10,031 | 13,783 | 115 |
| 3.0 | 35.63 | 22.72 | 12.91 |  | 89.11 | 77.34 | 11.76 |  | 5,125 | 9,815 | 13,828 | 110 |
| 4.0 | 35.67 | 22.91 | 12.77 |  | 89.95 | 78.29 | 11.66 |  | 5,245 | 9,692 | 13,831 | 108 |
| 5.0 | 35.70 | 23.02 | 12.68 |  | 90.48 | 78.88 | 11.60 |  | 5,320 | 9,617 | 13,831 | 100 |

Under all road conditions, it is observed that, due to the fast travel time (on average) offered by ARH services, the number of passengers choosing ARH paths increased when travel time became more critical to their journeys (i.e. as increased). Such an increase also appeared in the number of passengers choosing joint paths. Meanwhile, the usage of URT services decreased. This led to a decrease in the number of feasible transfers used by passengers in the URT network, which means less pressure for URT operators in actual operation. Differently, when passengers are not that sensitive to travel times, the cheaper service offered by URT attracts more passengers. At the same time, it also brings greater operational pressure to URT operators, because the number of transfers that need to be maintained as feasible in actual operation increases.

The above arguments still hold, when road condition varies. Differently, a more congested road will further make URT service more attractable as the speed advantage of ARH fades and passengers prefer a more reliable transportation mode to guarantee at least a cheaper transport can be purchased to complete their journeys. The increased use of URT services has also resulted in an increase in the number of feasible transfers used by passengers, which requires additional attention from URT operators.

In addition, the results in Table 8 also show that the number of passengers choosing joint paths in normal traffic was larger than in congested traffic, which indicates that the coordination between URT service and ARH service is more demanding in less congested road conditions.

1. Conclusion

In this study, we’ve optimized the coordination between ARH and URT services through a last-train timetabling design. To capture the road traffic uncertainties, we proposed a two-stage stochastic optimization model and considered a multinomial logit choice model to describe passengers’ traveling behavior when both URT and ARH services are available. To address the computational complexity of this model, we further adopted the SAA strategy to handle uncertainties and used both a commercial solver and a GA-based solution to solve the problem. Through the results of our experiments, the GA-based solution outperforms the direct use of the commercial solver regarding its computational performance. Thanks to the feasibility procedure throughout the chromosome generation, the GA-based solution also demonstrates high quality in its solutions when benchmarked with the results from a commercial solver. Based on a realistic transport network of a medium-sized Chinese city, the practicability of our solution is further articulated and some key findings are obtained:

1) It is worth offering additional support to last-train timetable optimization with respect to different types of passengers by URT managers. In particular, when passengers are fairly new to the URT network (e.g., at the beginning of the operation of new URT lines), URT operators should maintain as many transfers as feasible because passengers’ path choices would be more scattered. More traffic live information should also be provided to passengers to assist their travel decisions. As passengers become more familiar with the transportation conditions and their path choices could be more concentrated, the number of transfers that need to be maintained as feasible could be reduced, while the passenger demand that URT can meet would increase.

2) It is important to understand passengers’ preferences and average road conditions during the last-train operation period for obtaining an effective last-train timetable. As our experimental results suggest, the design of the last train timetable should focus more on the ability to coordinate with ARH services when passengers are more time-sensitive and/or when road conditions are less congested. The number of taxis required at each station should be provided to ARH despatcher as early as possible (e.g., when passengers catch the last trains at their origin stations) to ensure that passengers have available vehicles and reduce passengers’ waiting time for transfers.

Following the findings above, several further research opportunities are inspired. *First*, the current research setting assumes that the latest allowable service closure time is known, which largely limits the optimization space of the last train timetable. The decision on the latest allowable service closure is influenced by the rolling stock deadhead routing plan and night maintenance plan [36]. Therefore, it would be interesting to see how such coordinated last-train timetabling incorporates a cost-effective rolling stock deadhead routing plan. *Second*, the current coordinated last-train timetabling does not include the use of bus transport. Due to the increasing research interest in Mobility as a Service (MaaS) in the transportation field, e.g., Kim et al. [37] explored tourists’ preference for the combination of travel modes under the MaaS environment, it is also worth seeing how the research topic from this paper could be further extended to the MaaS context. *Third*, our current research only considers travel time uncertainties from ARH services, but more uncertainties are also observed during last-train operation period such as dynamic passenger demand or train delay. Therefore, it can enhance the practicability of the proposed models by considering more uncertain elements in the future. *Fourth*, since the computational efficiency of the proposed model and algorithm is related to the number of OD pairs considered, it would be meaningful to explore how to extract the representative OD pairs from all OD pairs, so as to reduce the number of OD pairs considered without sacrificing the quality of the solution. *Fifth*, as mentioned earlier, passengers from remote areas might not have access to available ARH services. In this case, the Park and Ride (P+R) mode would be a better travel option than ride-hailing, which however places more requirements on the robustness of last train timetables to ensure that passengers can reach designated transfer parking lots even if unexpected disruptions and disturbances could occur in actual train operations. Therefore, it would be beneficial to explore the robust optimization of last train timetables with considering passengers’ preferences for ride-hailing and P+R.

CRediT authorship contribution statement

**Jia Ning:** Conceptualization, Methodology, Software, Data analysis, Writing – original draft. **Xinjie Xing:** Conceptualization, Methodology, Data analysis, Writing – original draft. **Yadong Wang:** Conceptualization, Methodology, Writing – reviewing & editing. **Yu Yao:** Data collection, Methodology, Software, Writing – reviewing. **Liujiang Kang:** Conceptualization, Methodology, Writing – reviewing. **Qiyuan Peng:** Conceptualization, Data collection, Resources, Supervision, Funding acquisition, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This paper is supported by the National Key Research and Development Program of China (No. 2022YFB4300502), the National Natural Science Foundation of China (72301065), the China Postdoctoral Science Foundation (2023M730923), and the Natural Science Foundation of Jiangsu Province (BK20230964). The authors want to thank anonymous reviewers for their insightful comments on this paper.

Appendix

*A.1 Execution flowchart of CF model and CI model*



**Figure A. 1.** The flow chart of the execution process of the CI and CF models.

Reference

[1] J. Ning, Q. Peng, Y. Zhu, Y. Jiang, O. Anker, A Bi-objective optimization model for the last train timetabling problem, J. Rail Transp. Plan. Manag. 23 (2022) 100333. https://doi.org/10.1016/j.jrtpm.2022.100333.

[2] X. Guo, J. Wu, H. Sun, X. Yang, J.G. Jin, D.Z.W. Wang, Scheduling synchronization in urban rail transit networks: Trade-offs between transfer passenger and last train operation, Transp. Res. Part A Policy Pract. 138 (2020) 463–490. https://doi.org/10.1016/j.tra.2020.06.008.

[3] L. Kang, Q. Meng, Two-phase decomposition method for the last train departure time choice in subway networks, Transp. Res. Part B Methodol. 104 (2017) 568–582. https://doi.org/10.1016/j.trb.2017.05.001.

[4] L. Kang, H. Sun, J. Wu, Z. Gao, Last train station-skipping, transfer-accessible and energy-efficient scheduling in subway networks, Energy. 206 (2020) 118127. https://doi.org/10.1016/j.energy.2020.118127.

[5] L. Kang, J. Wu, H. Sun, X. Zhu, Z. Gao, A case study on the coordination of last trains for the Beijing subway network, Transp. Res. Part B Methodol. 72 (2015) 112–127. https://doi.org/10.1016/j.trb.2014.09.003.

[6] L. Kang, J. Wu, H. Sun, X. Zhu, B. Wang, A practical model for last train rescheduling with train delay in urban railway transit networks, Omega (United Kingdom). 50 (2015) 29–42. https://doi.org/10.1016/j.omega.2014.07.005.

[7] L. Kang, X. Zhu, H. Sun, J. Wu, Z. Gao, B. Hu, Last train timetabling optimization and bus bridging service management in urban railway transit networks, Omega (United Kingdom). 84 (2019) 31–44. https://doi.org/10.1016/j.omega.2018.04.003.

[8] L. Kang, X. Zhu, J. Wu, H. Sun, S. Siriya, T. Kanokvate, Departure Time Optimization of Last Trains in Subway Networks: Mean-Variance Model and GSA Algorithm, J. Comput. Civ. Eng. 29 (2015) 04014081. https://doi.org/10.1061/(asce)cp.1943-5487.0000407.

[9] L. Kang, X. Zhu, Strategic timetable scheduling for last trains in urban railway transit networks, Appl. Math. Model. 45 (2017) 209–225. https://doi.org/10.1016/j.apm.2016.12.016.

[10] W. Nie, H. Li, N. Xiao, H. Yang, Z. Jiang, N. Buhigiro, Modeling and solving the last-shift period train scheduling problem in subway networks, Phys. A Stat. Mech. Its Appl. 569 (2021) 125775. https://doi.org/10.1016/j.physa.2021.125775.

[11] C. Wang, X. Meng, M. Guo, H. Li, Z. Hou, An integrated energy-efficient and transfer-accessible model for the last train timetabling problem, Phys. A Stat. Mech. Its Appl. 588 (2022) 126575. https://doi.org/https://doi.org/10.1016/j.physa.2021.126575.

[12] D. Yu, H. Xu, W. Chen, J. Yao, Synchronized optimization of last trains’ timetables in mass rail transit networks based on extra dwell time, J. Phys. Conf. Ser. 1176 (2019). https://doi.org/10.1088/1742-6596/1176/5/052034.

[13] Q. Zhang, X. Li, T. Yan, L. Lu, Y. Shi, Last train timetabling optimization for minimizing passenger transfer failures in urban rail transit networks: A time period based approach, Phys. A Stat. Mech. Its Appl. 605 (2022) 128071. https://doi.org/https://doi.org/10.1016/j.physa.2022.128071.

[14] F. Zhou, Y. Li, R. Xu, Demand-Driven Transfer Coordination of Last Trains in Metro Network, Transp. Res. Rec. 2672 (2018) 316–326. https://doi.org/10.1177/0361198118773185.

[15] Y. Chen, B. Mao, Y. Bai, T.K. Ho, Z. Li, Timetable synchronization of last trains for urban rail networks with maximum accessibility, Transp. Res. Part C Emerg. Technol. 99 (2019) 110–129. https://doi.org/10.1016/j.trc.2019.01.003.

[16] Y. Zhou, Y. Wang, H. Yang, X. Yan, Last train scheduling for maximizing passenger destination reachability in urban rail transit networks, Transp. Res. Part B Methodol. 129 (2019) 79–95. https://doi.org/10.1016/j.trb.2019.09.006.

[17] S. Long, L. Meng, J. Miao, X. Hong, F. Corman, Synchronizing Last Trains of Urban Rail Transit System to Better Serve Passengers from Late Night Trains of High-Speed Railway Lines, Networks and Spatial Economics, 2020. https://doi.org/10.1007/s11067-019-09487-0.

[18] F. Wang, R. Xu, X. Song, P. Wang, Collaborative optimization of last-train timetables for metro network to increase service time for passengers, Comput. Oper. Res. 151 (2023) 106091. https://doi.org/10.1016/j.cor.2022.106091.

[19] L. Yang, Z. Di, M.M. Dessouky, Z. Gao, J. Shi, Collaborative optimization of last-train timetables with accessibility: A space-time network design based approach, Transp. Res. Part C Emerg. Technol. 114 (2020) 572–597. https://doi.org/10.1016/j.trc.2020.02.022.

[20] Y. Yao, X. Zhu, H. Shi, P. Shang, Last train timetable optimization considering detour routing strategy in an urban rail transit network, Meas. Control (United Kingdom). 52 (2019) 1461–1479. https://doi.org/10.1177/0020294019877480.

[21] W. Zhou, L. Deng, M. Xie, X. Yang, Coordination optimization of the first and last trains’ departure time on urban rail transit network, Adv. Mech. Eng. 2013 (2013). https://doi.org/10.1155/2013/848292.

[22] K. Huang, J. Wu, F. Liao, H. Sun, F. He, Z. Gao, Incorporating multimodal coordination into timetabling optimization of the last trains in an urban railway network, Transp. Res. Part C Emerg. Technol. 124 (2021) 102889. https://doi.org/10.1016/j.trc.2020.102889.

[23] K. Huang, J. Wu, H. Sun, X. Yang, Z. Gao, X. Feng, Timetable synchronization optimization in a subway–bus network, Phys. A Stat. Mech. Its Appl. 608 (2022) 128273. https://doi.org/https://doi.org/10.1016/j.physa.2022.128273.

[24] J. Ning, Q. Peng, Y. Zhu, X. Xing, O.A. Nielsen, Bi-objective optimization of last-train timetabling with multimodal coordination in urban transportation, Transp. Res. Part C Emerg. Technol. 154 (2023) 104260. https://doi.org/10.1016/j.trc.2023.104260.

[25] K. Yang, Y. Lu, L. Yang, Z. Gao, Distributionally robust last-train coordination planning problem with dwell time adjustment strategy, Appl. Math. Model. 91 (2021) 1154–1174. https://doi.org/10.1016/j.apm.2020.10.035.

[26] S. Yang, K. Yang, Z. Gao, L. Yang, J. Shi, Last-Train Timetabling under Transfer Demand Uncertainty: Mean-Variance Model and Heuristic Solution, J. Adv. Transp. 2017 (2017). https://doi.org/10.1155/2017/5095021.

[27] S. Yang, K. Yang, L. Yang, Z. Gao, MILP formulations and a TS algorithm for reliable last train timetabling with uncertain transfer flows, J. Oper. Res. Soc. 69 (2018) 1318–1334. https://doi.org/10.1080/01605682.2017.1392406.

[28] Y. Chen, B. Mao, Y. Bai, T.K. Ho, Z. Li, Optimal coordination of last trains for maximum transfer accessibility with heterogeneous walking time, J. Adv. Transp. 2019 (2019). https://doi.org/10.1155/2019/9692024.

[29] W. Li, Q. Luo, Q. Cai, Coordination of Last Train Transfers Using Potential Passenger Demand from Public Transport Modes, IEEE Access. 7 (2019) 126037–126050. https://doi.org/10.1109/ACCESS.2019.2939401.

[30] D. Eppstein, Finding the k shortest paths, SIAM J. Comput. 28 (1998) 652–673. https://doi.org/https://doi.org/10.1137/S0097539795290477.

[31] J. Zeng, Y. Qian, F. Yin, L. Zhu, D. Xu, A multi-value cellular automata model for multi-lane traffic flow under lagrange coordinate, Comput. Math. Organ. Theory. 28 (2022) 178–192. https://doi.org/10.1007/s10588-021-09345-w.

[32] J. Zeng, Y. Qian, J. Li, Y. Zhang, D. Xu, Congestion and energy consumption of heterogeneous traffic flow mixed with intelligent connected vehicles and platoons, Phys. A Stat. Mech. Its Appl. 609 (2023) 128331. https://doi.org/https://doi.org/10.1016/j.physa.2022.128331.

[33] Y. Qian, J. Zeng, N. Wang, J. Zhang, B. Wang, A traffic flow model considering influence of car-following and its echo characteristics, Nonlinear Dyn. 89 (2017) 1099–1109. https://doi.org/10.1007/s11071-017-3502-5.

[34] M. Ben-Akiva, M. Bierlaire, Discrete Choice Models with Applications to Departure Time and Route Choice, in: Handb. Transp. Sci., 2006: pp. 7–37. https://doi.org/10.1007/0-306-48058-1\_2.

[35] J.X. Dong, C.Y. Lee, D.P. Song, Joint service capacity planning and dynamic container routing in shipping network with uncertain demands, Transp. Res. Part B Methodol. 78 (2015) 404–421. https://doi.org/10.1016/j.trb.2015.05.005.

[36] D. Wang, A. D’Ariano, J. Zhao, Q. Zhong, Q. Peng, Integrated rolling stock deadhead routing and timetabling in urban rail transit lines, Eur. J. Oper. Res. 298 (2022) 526–559. https://doi.org/10.1016/j.ejor.2021.05.053.

[37] E.J. Kim, Y. Kim, S. Jang, D.K. Kim, Tourists’ preference on the combination of travel modes under Mobility-as-a-Service environment, Transp. Res. Part A Policy Pract. 150 (2021) 236–255. https://doi.org/10.1016/j.tra.2021.06.016.