Hindawi
Journal of Advanced Transportation
Volume 2018, Article ID 4530787, 18 pages
https://doi.org/10.1155/2018/4530787

Research Article

Optimizing High-Speed Railroad Timetable with Passenger and Station Service Demands: A Case Study in the Wuhan-Guangzhou Corridor

Jin Wang, Leishan Zhou, Yixiang Yue, Jinjin Tang, and Zixi Bai

School of Traffic and Transportation, Beijing Jiaotong University, Beijing 100044, China

Correspondence should be addressed to Leishan Zhou; lshzhou@bjtu.edu.cn

Received 22 May 2018; Revised 26 July 2018; Accepted 4 September 2018; Published 24 September 2018

Academic Editor: Samiul Hasan

This paper aims to optimize high-speed railroad timetables for a corridor. We propose an integer programming model using a time-space network-based approach to consider passenger service demands, train scheduling, and station service demands simultaneously. A modified branch-and-price algorithm is used for the computation. This algorithm solves the linear relaxation of all nodes in a branch-and-bound tree using a column generation algorithm to derive a lower-bound value (LB) and derive an upper-bound value (UB) using a rapid branching strategy. The optimal solution is derived by iteratively updating the upper- and lower-bound values. Three acceleration strategies, namely, initial solution iteration, delayed constraints, and column removal, were designed to accelerate the computation. The effectiveness and efficiency of the proposed model and algorithm were tested using Wuhan-Guangzhou high-speed railroad data. The results show that the proposed model and algorithm can quickly reduce the defined cost function by 38.2% and improve the average travel speed by 10.7 km/h, which indicates that our proposed model and algorithm can effectively improve the quality of a constructed train timetable and the travel efficiency for passengers.

1. Introduction

A train timetable is a technical document that describes the operation of trains on certain railroad sections and the trains' departure and arrival times or passing times at stations. With the rapid development of high-speed railroads in China and various other countries during the last decade, the train timetabling problem (TTP) for high-speed railroads has become a new challenge for railway operators. The TTP for high-speed railroads differs from the traditional TTP in several ways:

(1) The mismatch between transportation demand and capacity causes the full use of the transportation capacity of railroads and the minimization of travel times of trains to be the main goals for the traditional TTP. In general, the TTP for high-speed railroads focuses on the maximal satisfaction of passenger travel demands and the improvement in customer service quality. Due to the high speed and high density of the high-speed railroad, passengers may have more choices of when to travel. Thus, the passenger flow of high-speed railroads shows distinct volatility in different periods. To satisfy the volatility characteristics of high-speed railroad passengers, we set a feasible departure time range for each train at its starting station according to the train service plan, as mentioned in Section 2.3.

(2) Passenger trains and freight trains run on traditional railroads, where the passenger trains have higher priority than freight trains. Thus, passenger trains may overtake freight trains at some stations. Medium- and high-speed passenger trains run on high-speed railroads, where medium-speed trains and high-speed trains have the same priority. Thus, the number of times overtaking occurs on high-speed railroads may be less than that of traditional railroads. We propose a punishment coefficient to control the number of occurrences on high-speed railroads in Section 3.1.

(3) The maintenance window for high-speed railroads is generally set at night, and maintenance requires a considerable length of time. The skylight for traditional railroads is more flexible and requires a shorter length of time.

The TTP has received considerable attention in recent decades as a fundamental aspect of railroad transportation management. Many researchers have explored train scheduling. Frank [1] was one of the earliest researchers to conduct a mathematical analysis of train timetables. He began to explore train scheduling for two-way railroad operations on a single track in 1966. Szpigel [2] converted the single-track TTP into a job scheduling problem and proposed a mixed-integer programming model, in which minimizing the total travel time was defined as the goal, which was subject to interval time constraints related to train overtaking or crossing. A branch-and-bound algorithm was employed for the computation of this model; however, it could be applied to only small-scale computations. Serafini and Ukovich [3] proposed the periodic event scheduling problem (PESP) in 1989, after which Schrijver and Steenbeek [4] applied the PESP to solve the TTP and pioneered research on periodic timetable optimization problems [5–11]. Periodic timetables help passengers memorize accurate departure times and effectively reduce the scale of the computation. Currently, periodic timetabling is the preferred choice of many railroad enterprises. Brännlund et al. [12] proposed a 0-1 integer programming model based on the discretization of time, the goal of maximizing the benefits for all trains, and the need to satisfy track capacity constraints. The Lagrange relaxation algorithm was employed for the computation of this model. Caprara et al. [13] explored periodic timetables based on graph theory and constructed an integer programming model for which the Lagrange relaxation algorithm was also used for computation. This model can be effectively applied to large-scale computations. Zhou and Zhong [14] explored the single-track TTP for conditions of limited track resources with the goal of minimizing the total travel time with safety headway constraints. Cacchiani et al. [15] proposed a time-space network-based optimization model for the TTP using a full timetable for a train as a variable. A column generation algorithm was utilized for the computation of this model. He et al. [16] explored the high-speed railroad TTP based on a time-space network model and the branch-and-price algorithm. They implemented a punishment value to reduce the deviations of train start times and the number of occurrences of train overtaking. They also implemented a strategy of increasing the amount of stopping at stations to increase the possibility of train overtaking but they did not consider the strategy of reducing the amount of stopping at stations to shorten passenger waiting times. These studies have yielded useful methods for the optimization of the TTP. However, due to the hierarchical optimization of traditional transportation management processes, passenger demands were not directly considered in these studies.

Another subset of previous studies has focused on passenger demands. Ceder [17] proposed an optimization framework for constructing bus timetables using passenger loading data. This framework can be used to synchronize the departure times of vehicles under dynamic passenger demands. Peeters and Kroon [18] employed the branch-and-price algorithm to solve the rolling stock scheduling problem for a given train timetable and passenger seat demands. Zhou et al. [19] constructed a bilevel programming model to optimize a passenger train operation plan and diagram with the goal of maximizing the profits of the railroad enterprise. Passengers can determine whether to travel by train and can select their transfer schemes based on ticket prices, departure and arrival times, switching times, and congestion charges. The railroad enterprise can improve its train operation plan and diagram based on passenger demands. Kaspi and Raviv [20] proposed a model for comprehensively optimizing a train operation plan and diagram with the goal of minimizing the train operation costs and passenger travel times based on time-varying passenger demands. This model did not have any train capacity constraints; this assumption reduced the computational complexity but yielded results that were inconsistent with the actual situation. Canca et al. [21] proposed a nonlinear integer programming model with the goal of minimizing passenger waiting times and the operational costs for the railroad enterprise. In this model, the departure and arrival times were determined based on dynamic passenger demands. Wang et al. [22] proposed an event-driven model that included departure events, arrival events, and events that correspond to changes in passenger arrival rates. They also constructed a nonlinear programming model with the multiple goals of minimizing the total passenger travel time and optimizing the energy consumption based on passenger transfer behavior in a railroad network. High-speed railroad operations enable highly dense train schedules with the ability to satisfy time-varying passenger demands. Niu et al. [23] proposed a nonlinear mixed-integer programming model based on time-varying origin-destination (OD) passenger demand matrices and explored the high-speed railroad TTP with the goal of minimizing passenger waiting times. Yue et al. [24] proposed a mathematical model for optimizing high-speed railroad timetables to simultaneously consider both passenger service demands and train scheduling. Lagrangian relaxation and a column generation algorithm were employed for computation. They verified the effectiveness of their model and algorithm via a case study of the Beijing-Shanghai high-speed railroad.

Stations are important nodes in a high-speed railroad transportation network; they are the locations of events, such as passenger boarding, alighting, and transfer. Therefore, some researchers have studied the simultaneous optimization of passenger and station service demands with the main goal of optimizing the train stop schedule plan. Suh et al. [25] implemented a skip-stop strategy for subway service planning based on OD matrix information, distances between stations, headways, and maximum link speeds. Deng et al. [26] proposed a bilevel programming model for train stop schedule planning based on the passenger transfer demands, the level of stations at which trains stop, the operational capacities of the train stations, and other factors. The objective function of the upper programming model included the generalized minimum travel cost and the number of train stops, whereas the lower programming model was a passenger flow assignment model with multiclass user equilibrium based on the train stop schedule. Zheng et al. [27] constructed a 0-1 integer programming model that considered skip-stop patterns, with the goal of minimizing the total passenger travel time. The tabu search algorithm was employed for computation. Li et

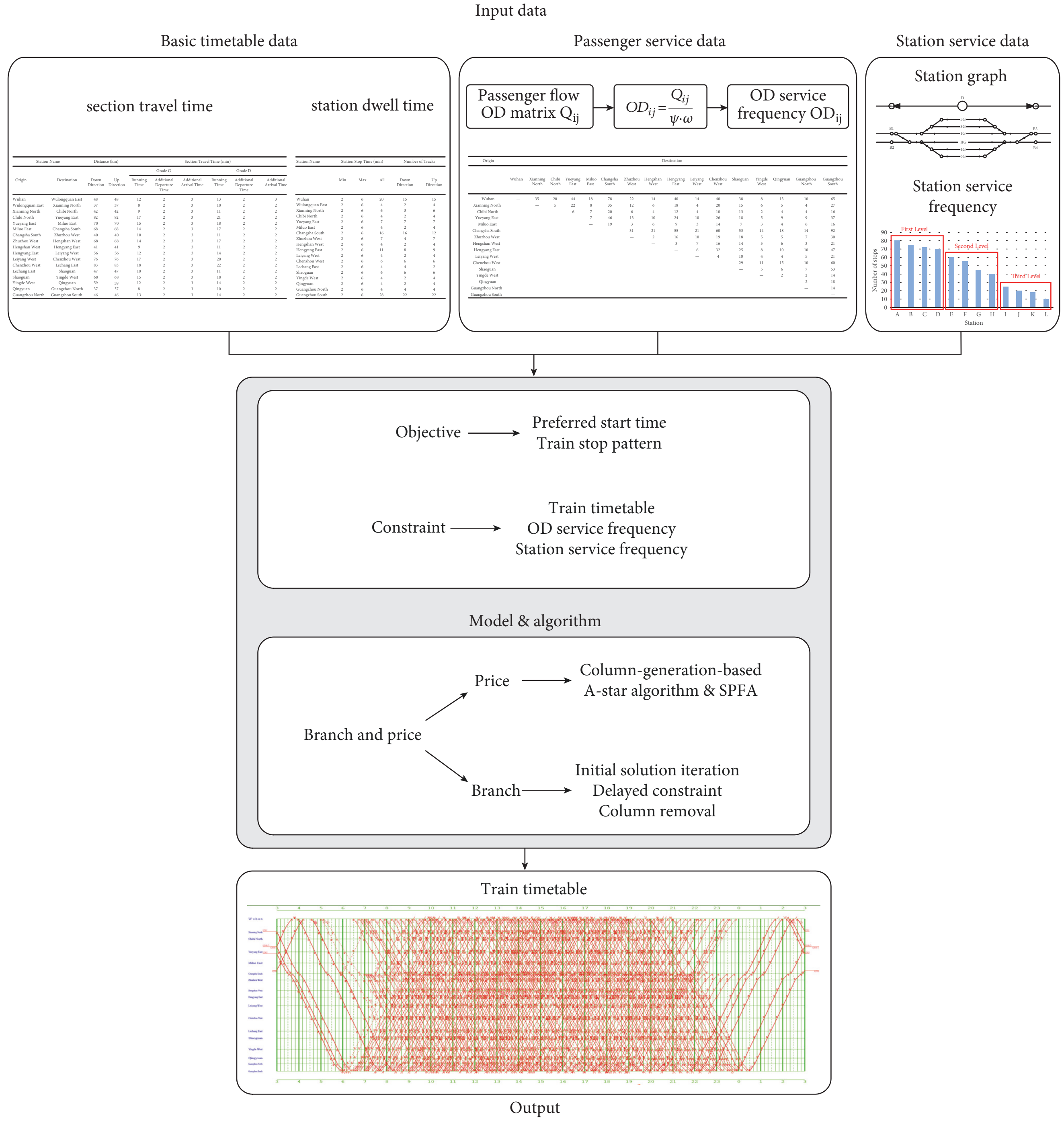


Figure 1: Overview of the proposed high-speed railroad train timetable optimization method.

al. [28] analyzed problems encountered with conventional train stop schedules and proposed a high-speed railroad stop schedule optimization model, in which the main constraints were related to the node service frequency, the interstation service accessibility, and the number of one-train stops, and the goal of optimization was to minimize the total number of stops at stations. They also quantitatively analyzed the effects of various factors on the construction of train stop schedule plans. Current research is primarily focused on the station stop schedules and does not consider the simultaneous optimization of train timetables.

We propose a new high-speed railroad train timetable optimization method using a time-space network-based approach and a modified branch-and-price algorithm to simultaneously consider passenger service demands, train scheduling, and station service demands. Figure 1 presents an overview of the method. The input data include the timetable parameters, passenger service data, and station service data.

First, the timetable parameters, such as the section travel times and dwell times at stations, are acquired from the existing diagram. Second, the service frequency of each station OD pair is inferred from the average daily passenger flow data provided by the passenger ticketing department. Last, the number of sidetracks at each station is determined based on the station graph. The stations are divided into nodes of different grades in accordance with the size, geographic location, and passenger flow data of each station. The service frequency of each station is determined according to its node grade. We propose a new integer programming model with constraints related to OD service frequency, station service frequency, and various operation time standards, with the goal of minimizing the deviation from the predefined train start times, the total number of train stops, and the dwell time at stations. The details regarding the modified branch-and-price algorithm and the three acceleration strategies developed for this algorithm are discussed in Section 4 of this paper. The output is a near-global-optimal train timetable.

The four main contributions of our study can be summarized as follows:

(1) We propose a new time-space network method to represent a train stop schedule plan and the comprehensive optimization of the train timetable. We use an associated labeling method to address the corresponding challenge of unconnected arc segments.

(2) We propose a new integer programming model that simultaneously considers passenger service demands, train scheduling, and station service demands.

(3) To calculate the pricing problem (PP), we employ a new method based on the A-star (A$*$) algorithm and the Shortest-Path Faster Algorithm (SPFA) to search for the shortest path in the unconnected and dynamic time-space network.

(4) We propose a modified branch-and-price algorithm and design three acceleration strategies for this algorithm: initial solution iteration, delayed constraints, and column removal. These algorithms and strategies can be used to quickly solve a very-large-scale TTP.

The remainder of this paper is organized as follows: In Section 2, we elaborate on the research problem and introduce the construction of the time-space network. In Section 3, we describe the new integer programming model. In Section 4, we discuss the extended branch-and-price algorithm and the three acceleration strategies. Section 5 presents a verification of the effectiveness of the proposed algorithm and model using data from the Wuhan-Guangzhou high-speed railroad. In Section 6, we discuss our conclusions and offer suggestions for future studies.

2. Problem Statement

This paper considers the high-speed TTP along one train direction of a two-direction railroad line under passenger demands, where the stations are sequentially numbered as $1, 2, \cdots, n_S$. Let $S = \{1, 2, \cdots, n_S\}$ be the set of stations. The set $N = \{1, 2, \cdots, n_N\}$ of trains travel along the line from the predetermined starting stations to the predetermined ending stations according to the train service plan. Let $t \in T$ be the index of times, where $T = \{0, 1, \cdots, n_T\}$ is the planning time horizon with equal one-minute intervals.

2.1. OD Service Frequency. The OD service frequency, which is denoted by $OD_{i,i'}$, is the minimum number of direct trains that serve passengers who travel between the stations of an OD pair. The relationship between the OD service frequency and the passenger flow data can be expressed by the following equation:

$$OD_{i,i'} = \frac{Q_{i,i'}}{\psi_j \cdot \omega_{i,i'}}, \tag{1}$$

where $Q_{i,i'}$ represents the average number of passengers who arrive at station $i \in S$ to travel to station $i' \in S : i' \neq i$ daily on the line, according to the passenger ticketing department; ψ_j is the capacity of train $j \in N$, i.e., the maximum number of passengers accommodated by train j; and $\omega_{i,i'}$ is the train seat occupation rate between station i and station i'.

Figure 2 lists the OD service frequencies of a line. Light red shading indicates OD sections with lower service frequencies. These sections have poor travel access, and passengers have few travel options. Yellow shading indicates OD sections with zero service frequency. Passengers on these OD sections must switch stations. As shown in Figure 2, the OD service frequencies related to station K are low; this situation is attributed to the low grade of the station and its low passenger traffic. A low OD service frequency also causes low attractiveness to passengers from sections related to station K, which inhibits an increase in passenger traffic. Therefore, a reasonable OD service frequency is an important condition for guaranteeing accessible services among stations and the satisfaction of passenger travel demands.

2.2. Station Service Frequency. Station service frequency is the number of trains that stop at and start from a station during a certain period [28]. In actual calculations, the grade of a station is determined based on various factors, such as political, economic, and cultural factors of the city in which the station is located and its attractiveness to passenger traffic, infrastructure, and operational capability. The service frequency of a station is determined in accordance with its grade.

2.3. Predefined Start Times. In theory, the start time of a train can have any value within a possible start time range. In practice, reasonable start time ranges can ensure that the train timetable exhibits reasonable performance in terms of the evenness of the train distribution and the satisfaction of the time-varying passenger demands. Suitable start time ranges can also tighten the feasible space of the resulting optimization models and reduce the search time. The start time range for a train can be determined based on the experience of the planners or acquired from existing timetables [23].

2.4. Description of a Train Timetable in the Form of a Time-Space Network. A description of a train timetable in the form of a time-space network can accurately reflect various

	A	B	C	D	E	F	G	H	I	J	K	L	M
A	-	26	46	9	7	10	10	25	6	13	2	4	26
B	10	-	58	8	12	10	11	30	6	11	2	6	32
C	33	46	-	25	31	18	24	64	14	29	3	11	69
D	10	6	28	-	12	11	15	33	7	18	1	3	26
E	6	9	29	11	-	7	13	29	6	14	0	4	26
F	6	10	21	8	7	-	9	20	4	9	3	2	17
G	10	13	30	12	13	7	-	26	6	13	1	5	22
H	16	24	65	34	29	18	29	-	18	35	5	14	69
I	6	8	20	8	8	8	8	20	-	11	3	4	16
J	7	11	31	20	15	4	14	39	10	-	3	6	25
K	3	0	7	3	5	1	3	10	2	5	-	4	6
L	4	6	17	8	7	6	6	19	6	6	5	-	17
M	16	24	70	28	21	19	24	68	20	27	10	23	-

Figure 2: OD service frequencies of a train line.

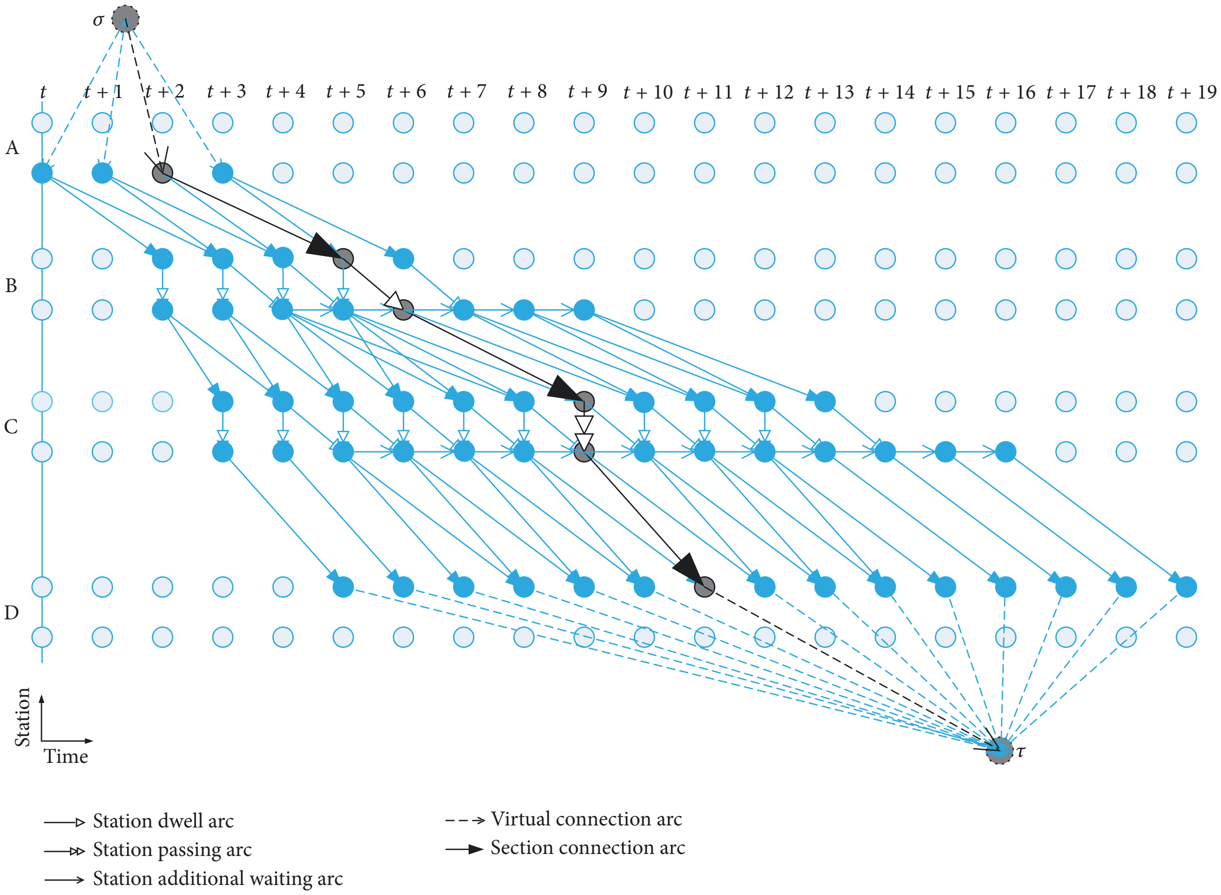


Figure 3: Illustration of a time-space network.

events and their occurrence times during train operations [16]. A time-space network is the combination of the physical network and the time horizon.

We use Figure 3 to demonstrate the time-space network formulation, in which the physical route of train j consists of four nodes/stations and three links/sections. The horizontal axis corresponds to discrete time, where the time horizon is represented as $[t, t + 19]$ with equal one-minute intervals. The vertical axis corresponds to space and represents stations and sections. The time-space vertex represents the arrival or departure state of train j at a physical node at the current time, and the time-space arc represents the movement of train j in stations or sections with the entering and leaving times.

The time-space network starts from the virtual departure node σ. Different virtual departure arcs are generated based on the feasible start time range. We define the section connection arcs to imply the section travel times of train j. When train j passes by a station, we define the station passing arcs to imply it. When train j stops at a station, additional starting and stopping times and dwell times are incurred.

Additional starting and stopping times are calculated in accordance with the stop statuses of train j at the two adjacent stations and added to the section connection arcs. The dwell times at stations are required to satisfy certain maximum and minimum dwell time requirements. The purpose of imposing a maximum dwell time is to tighten the feasible space of the optimization model and produce a more reasonable train timetable. The minimum dwell time is the minimum time required for passenger boarding, alighting, transfer, and other operations. We define the station dwell arcs to imply the minimum dwell times of train j at stations. Any additional dwell time beyond the minimum dwell time is represented by one or more additional waiting arcs. The time duration of each additional waiting arc is fixed to one minute in this paper. The network ends at the virtual arrival node τ. Thus, the path from σ to τ in the time-space network corresponds to a feasible timetable for train j. Each train j has multiple paths that constitute the path set $P_j \subset P$, where P is the set of all paths.

For example, the path $p \in P_j$, as shown by the black solid lines in Figure 3, starts from station A at time $t + 2$. Traveling with the section connection arc, which is subject to the section pure travel time and additional starting and stopping times, train j arrives at station B at time $t + 5$. After dwelling for one time interval, train j departs from station B at time $t + 6$. Train j passes by station C at time $t + 9$ after traveling with the section connection arc between station B and station C, which is subject to the section pure travel time and additional starting times. Likewise, train j reaches the destination at time $t+11$. In this manner, the train scheduling problem can be transformed into a routing problem in the time-space network.

We redesign the train stop schedule plan when optimizing the train timetable, unlike in the traditional TTP, in which the train stop pattern is predetermined. Thus, both station passing arcs and station dwell arcs are generated for the train path at every station. During the optimization process, the train can choose to either stop at or pass by a station, depending on demand. For stations at which trains must stop for technical reasons, only station dwell arcs and additional waiting arcs are generated. The existence of additional starting and stopping times and the variable stop pattern produce unconnected arcs in the time-space network. To address this issue, we add an associated label $(\beta_\alpha^{sto}, \beta_\alpha^{pst})$ to each arc $\alpha \in A$, where A is the set of arcs. When α is a section arc, β_α^{sto} indicates whether α stops at the ending station of this section, where a value of 1 represents yes and a value of 0 represents no, and β_α^{pst} indicates whether α stops at the starting station of that section. When α is a station arc, β_α^{sto} indicates whether α stops at this station, and β_α^{pst} indicates whether α stops at the preceding adjacent station.

As shown in Figure 4, neither C nor D is a must-stop station. The red path passes by station C. If the red path chooses to pass by station D (as on path p), the section operation arc is calculated based on the pure section travel time. The associated label is (0,0). If the red path chooses to stop at station D (as on path p'), the section operation arc is calculated using the pure section travel time and the additional stopping time at station D. The associated label is (1,0). The blue path stops at station C. If the blue path chooses to pass by station D (as on path q), the section operation arc is calculated using the pure section travel time and the additional starting time at station C. The associated label is (0,1). If the blue path chooses to stop at station D (as on path q'), the section operation arc is calculated using the pure section travel time, the additional starting time at station C, and the additional stopping time at station D. The associated label is (1,1). When selecting a time-space path, we choose the next arc from the subsequent connected arcs for which the β_α^{pst} label is equal to the β_α^{sto} label of the current arc.

3. Model Construction

3.1. Objective Function. We transfer the costs of the time-space nodes to the adjacent arcs. Thus, the cost of a time-space path is equal to the total cost of the included arcs. We define the objective function with the goal of minimizing this cost.

Let td_j be the predefined preferred departure time of train j from its starting station. tdr_p is the actual departure time from the starting station on path $p.\delta$ is the half-width of the permitted departure time range at the starting station. td_j^E and td_j^L are the earliest departure time and latest departure time of train j from its starting station. As noted in Section 2.3, the preferred start times of the trains are predefined. The permitted variations of the departure time fall in the range $[td_j{-}\delta, td_j{+}\delta]$. The cost of a virtual departure arc in this range is zero. Paths that lie outside this range are allowed but the punishment coefficient, c_1^{pen}/min, is added to achieve maximum control of the deviation of the departure time. The cost of the deviation is calculated and added to the cost of the virtual departure arc. In the example shown in Figure 5, the actual departure time of path $p \in P_j$ falls outside the allowed fluctuation range, and the cost of the corresponding virtual departure arc is $(tdr_p - td_j - \delta) \cdot c_1^{pen}$.

The stopping of a train at a station will cause an increase in the passenger travel time because it generates a dwell time and additional starting and stopping times. Our goal is to reduce the number of stops and the dwell time. The number of stops is associated with the punishment coefficient c_2^{pen}/num. The punishment coefficient for the dwell time c_3^{pen}/min includes the punishment for both the station dwell arc and any corresponding additional waiting arcs.

A punishment of c_4^{pen}/min is applied for additional waiting arcs to reduce but not preclude the occurrence of train overtaking.

The costs of section arcs, station passing arcs, and virtual arrival arcs are 0.

The cost c^σ of a virtual departure arc is

$$c^\sigma = \begin{cases} \left(\left|tdr_p - td_j\right| - \delta\right) \cdot c_1^{pen} & \left|tdr_p - td_j\right| > \delta \\ 0 & else \end{cases} \quad (2)$$

$$\forall j, p \in P_j.$$

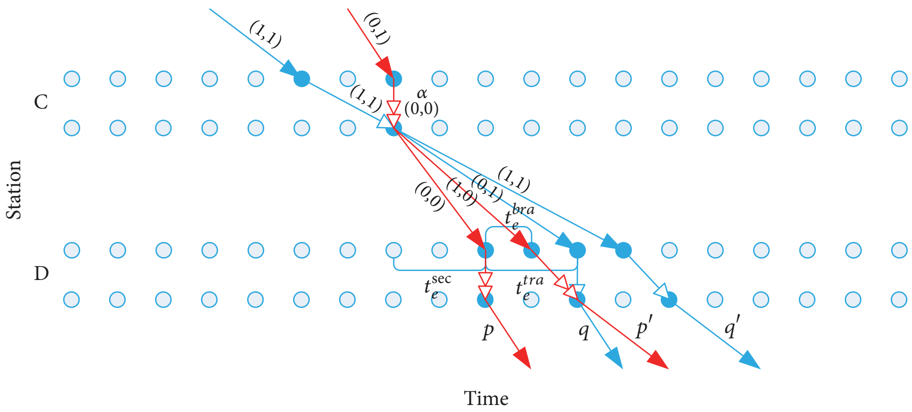


Figure 4: Illustration of unconnected arcs.

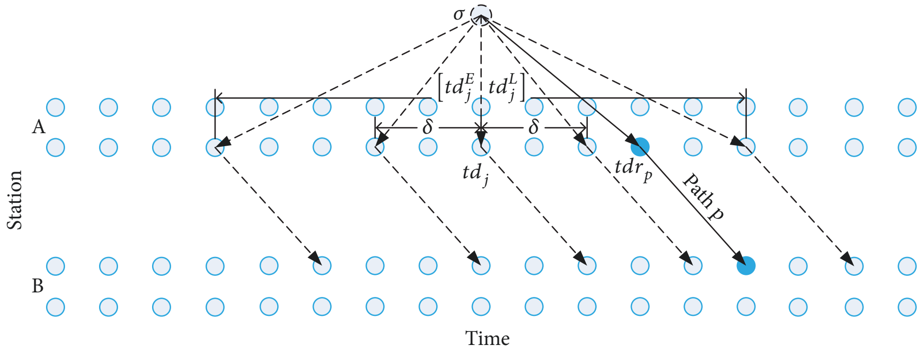


Figure 5: Illustration of the cost calculation for virtual departure arcs.

A^{sto} is the station dwell arc set. The cost c_α^{sto} of a station dwell arc $\alpha(\forall \alpha \in A^{sto})$ is

$$c_\alpha^{sto} = c_2^{pen} + \left(t_\alpha^{end} - t_\alpha^{beg}\right) \cdot c_3^{pen}, \tag{3}$$

where t_α^{beg} and t_α^{end} are the start time and end time of arc α.

A^{wai} is the station additional waiting arc set. The cost c_α^{wai} of an additional waiting arc $\alpha(\forall \alpha \in A^{wai})$ is

$$c_\alpha^{wai} = \left(t_\alpha^{end} - t_\alpha^{beg}\right) \cdot \left(c_3^{pen} + c_4^{pen}\right). \tag{4}$$

The cost c_p of a time-space path p is

$$c_p = c^\sigma + \sum_{\alpha \in A_p^{sto}} c_\alpha^{sto} + \sum_{\alpha \in A_p^{wai}} c_\alpha^{wai}. \tag{5}$$

The objective function is represented as

$$\min \sum_{j \in N} \sum_{p \in P_j} c_p x_p, \tag{6}$$

where the decision variable x_p indicates whether a certain path p appears in the optimal solution. It is equal to 1 if and only if path p is selected in the optimal solution and is equal to 0 otherwise.

3.2. Constraints. The network flow constraint,

$$\sum_{p \in P_j} x_p = 1 \quad \forall j, \tag{7}$$

represents the selection of only one path by each train.

The arrival safety headway constraint,

$$\sum_{p:\beta_\alpha^p = 1} x_p + \sum_{p':\beta_{\alpha'}^{p'} = 1} x_{p'} \le 1 \\ \forall e, \forall \alpha, \alpha' \in A_e^{\sec} : 0 \le t_{\alpha'}^{end} - t_\alpha^{end} < t_{d(e)}^{arr}, \tag{8}$$

represents a constraint on the safety headway for consecutive arrivals of trains that travel in the same direction. $e \in E$ is the index of sections, where E is the set of sections. $\alpha' \in A_e^{\sec}$ is the index of arcs, where $A_e^{\sec}$ is the section arc set. $o(e)$ and $d(e)$ are the starting station and ending station of section e. β_α^p is the binary parameter, which is equal to 1 if arc α belongs to path p and is equal to 0 otherwise. $t_{d(e)}^{arr}$ is the safety headway for trains that arrive at station $d(e)$.

The departure safety headway constraint,

$$\sum_{p:\beta_\alpha^p = 1} x_p + \sum_{p':\beta_{\alpha'}^{p'} = 1} x_{p'} \le 1 \\ \forall e, \forall \alpha, \alpha' \in A_e^{\sec} : 0 \le t_{\alpha'}^{beg} - t_\alpha^{beg} < t_{o(e)}^{dep}, \tag{9}$$

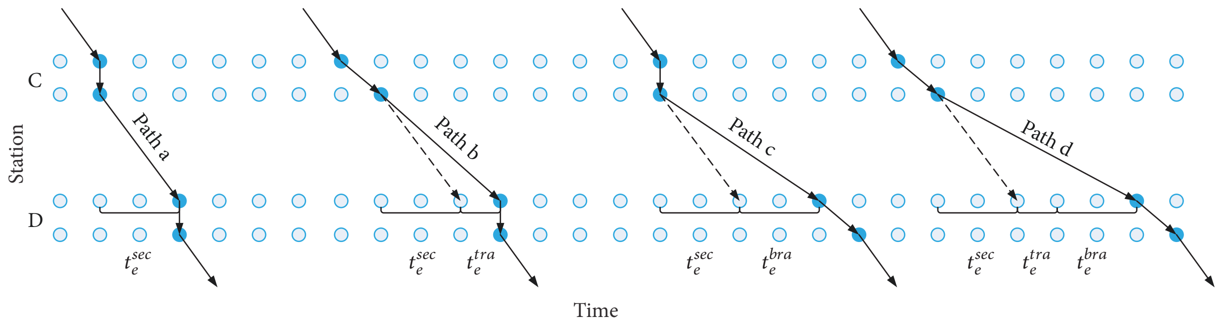


FIGURE 6: Illustration of the calculation of additional starting and stopping times.

represents a constraint on the safety headway for consecutive departures of trains that travel in the same direction. $t_{o(e)}^{dep}$ is the safety headway for trains that depart from station $o(e)$.

The overtaking constraint,

$$\sum_{p:\beta_\alpha^p=1} x_p + \sum_{p':\beta_{\alpha'}^{p'}=1} x_{p'} \leq 1$$
$$\forall e, \forall \alpha, \alpha' \in A_e^{\sec} : \left(t_{\alpha'}^{beg} - t_\alpha^{beg}\right) \cdot \left(t_{\alpha'}^{end} - t_\alpha^{end}\right) \leq 0, \tag{10}$$

represents the constraint that trains are not allowed to overtake other trains that travel in the same direction on a section.

The station capacity constraint is

$$\sum_{\alpha \in A_i^{dum}: t_\alpha^{beg} \leq t, t_\alpha^{end} \geq t} \; \sum_{p:\beta_\alpha^p=1} x_p \leq num_i \quad \forall i, t, \tag{11}$$

where num_i is the number of sidetracks at station i and A_i^{dum} is a set of virtual station arcs, including station passing arcs, station dwell arcs, and additional waiting arcs. A_i^{dum} is used to represent all possible operations of a train with respect to station i. The start time of an arc is the time at which the train enters the station from the previous section, and the end time of the arc is the time at which the train leaves the station and enters the next section.

The OD service frequency constraint,

$$\sum_{p:\tau_p^{i,i'}=1} x_p \geq OD_{i,i'} \quad \forall i, i', \tag{12}$$

represents the constraint that the number of trains that serve a given station OD pair should satisfy the passenger service demand of this pair. $\tau_p^{i,i'}$ is the coupled-stop index, which is equal to 1 if path p stops at both station i and station i' and is equal to 0 otherwise.

The station service frequency constraint,

$$\sum_{p:\mu_p^i=1} x_p \geq SF_i \quad \forall i, \tag{13}$$

represents the constraint that the number of trains that stop and start at a given station during a certain period should satisfy the service frequency requirement of the node to which this station belongs. SF_i is the station service frequency of station i, which corresponds to the station grade, and μ_p^i is the stop index, which is equal to 1 if path p stops at station i and is equal to 0 otherwise.

The departure time constraint is

$$td_j^E \leq tdr_p \leq td_j^L \quad \forall j, p \in P_j. \tag{14}$$

The station dwell time constraint is

$$t_i^{\min} \leq t_i^{sto} \leq t_i^{\max} \quad \forall i, p : \mu_p^i = 1, \tag{15}$$

where $t_i^{\max}$ and $t_i^{\min}$ are the maximum dwell time and minimum dwell time of trains at station i, and t_i^{sto} is the dwell time at station i on path p.

The section travel time constraint is

$$t_\alpha^{end} = t_\alpha^{beg} + t_e^{\sec} + \mu_{p:\beta_\alpha^p=1}^{o(e)} t_e^{tra} + \mu_{p:\beta_\alpha^p=1}^{d(e)} t_e^{bra}$$
$$\forall e, \forall \alpha \in A_e^{\sec}, \tag{16}$$

where $t_e^{\sec}$ is the pure section travel time of section e, and t_e^{tra} and t_e^{bra} are the additional starting time and stopping time of section e.

As shown in Figure 6, during the generation of the section arcs, the additional starting time and stopping time depend on the stopping statuses of the train at the two adjacent stations. These times are determined based on the results of the traction calculations.

The station travel time constraint is

$$t_\alpha^{end} = \begin{cases} t_\alpha^{beg} + t_i^{sto} & \mu_p^i = 1 \\ t_\alpha^{beg} & \mu_p^i = 0 \end{cases} \quad \forall i, p. \tag{17}$$

In the time-space network, the start time and end time of a station passing arc are equivalent. The end time of a station dwell arc is the sum of the start time and the minimum dwell time at this station. Any additional dwell time beyond the minimum dwell time is represented by one or more additional waiting arcs.

The decision variables are

$$x_p = \{0, 1\} \quad \forall p. \tag{18}$$

Model M1 (a generic model with passenger service demands and station service demands) consists of the objective function (6), which is subject to constraints (7)-(18). Constraints (14)-(17) are used to construct the time-space network. By limiting the departure time range at the starting station [(14)] and the maximum and minimum station dwell times [(15)], redundant time-space nodes and arcs can be effectively reduced. By defining the station dwell arcs and additional waiting arcs to imply the station dwell times [(17)], the number of station arcs can be effectively reduced. Thus, the complexity of the model and the scale of the time-space network are manageable. Model M1 is an integer programming model with a linear objective function and linear constraints. We apply an extended branch-and-price algorithm to solve this model.

4. Extended Branch-and-Price Algorithm

As the numbers of stations and trains increase, the computational scale of model M1 also increases. The problem becomes an integer linear programming problem on an extremely large scale. The number of generated feasible train paths, i.e., the number of decision variables, exceeds the number of constraints. Therefore, we propose a modified branch-and-price algorithm to solve this problem.

The branch-and-price algorithm employs a column generation algorithm to solve the linear relaxation of the nodes in the entire branch-and-bound tree, derives a reasonable lower-bound value for the original problem, and uses the branch-and-bound algorithm to solve the integer programming problem. The underlying idea of the column generation algorithm is to reduce the enumeration of all columns to only one set of feasible paths for the original problem in terms of a large-scale linear problem whose scale is considerably smaller than that of the original problem. A restricted master problem (RMP) is generated. Dual variables are derived by solving the RMP. Better paths are generated based on the PP and are added to the RMP. The algorithm iteratively alternates between the PP and the RMP until the best solution to the linear programming problem has been derived.

4.1. Restricted Master Problem. Model M1 is an integer linear programming model. In particular, (14)-(17) are used to construct the time-space network and are not applied to the RMP. The integer decision variables in (18) are relaxed to continuous variables, and the subset P_0 is selected from the path set P.

$$0 \leq x_p \leq 1 \quad \forall p \in P_0 \subset P \tag{19}$$

The RMP is defined as follows:

$$\text{RMP: Equations } (6)\text{-}(13) \text{ and } (19). \tag{20}$$

4.2. Pricing Problem. Solving the PP is a key step of the branch-and-price algorithm. The purpose is to seek the feasible solution that can reduce the objective function, by adding the column with the negative reduced cost to the RMP. The reduced cost of any column can be regarded as the price of this column.

γ_j, $\pi_{e,o,\alpha}$, $\beta_{e,d,\alpha}$, $\xi_{e,\alpha}$, $\lambda_{i,t}$, $\eta_{i,i'}$, and θ_i are dual variables that correspond to (7), (8), (9), (10), (11), (12), and (13), respectively. They represent shadow prices that are associated with time-space nodes, arcs, and other resources. For a certain train path $p \in P_j$, the reduced cost is

$$\begin{aligned} \varphi_p = {} & c_p - \gamma_j - \sum_e \sum_{\alpha \in A_e^{\mathrm{sec}}} \pi_{e,o,\alpha} - \sum_e \sum_{\alpha \in A_e^{\mathrm{sec}}} \beta_{e,d,\alpha} \\ & - \sum_e \sum_{\alpha \in A_e^{\mathrm{sec}}} \xi_{e,\alpha} - \sum_i \sum_t \lambda_{i,t} - \sum_i \sum_{i'} \eta_{i,i'} - \sum_i \theta_i. \end{aligned} \tag{21}$$

If $\varphi_p \geq 0$, then all train paths optimally satisfy the original problem, and the optimal solution to the linear programming problem is derived. If $\varphi_p < 0$, further optimization is required. Considering the characteristics of the time-space network, the PP can be transformed into a shortest-path problem. The reduced cost of an entire time-space path can be calculated based on the costs and weights of the arcs that belong to this path. The weights of the arcs are updated using the dual variables derived from the RMP. The path with the negative reduced cost constitutes a new column to be added to the RMP [16]. As shown in (21), the weights of arcs can be negative; in general, the problem can be solved using the Bellman-Ford algorithm or the SPFA.

4.3. Calculation of the Shortest Path. In general, once the weights of all time-space arcs have been determined, the PP can be solved using a shortest-path algorithm. Due to (12), a conventional shortest-path algorithm is not applicable in this case. Equation (12) is an OD service frequency constraint that is related to any pair of different stop stations on a train path. Dual variables cannot be directly employed to update the arc weights; instead, we identify the stopping statuses along the entire path and then update the weights of the path using the dual variables that correspond to each pair of different stop stations. Due to the large scale of the time-space network, a search of all paths would be excessively time consuming and significantly affect the efficiency of the algorithm. Therefore, we propose a new method based on the A$*$ algorithm and the reversed SPFA.

The A$*$ algorithm can be regarded as a heuristic algorithm in which the estimation function $f^*(m) = g(m) + h^*(m)$ is used to conduct a network search. This algorithm has been widely applied in searches for an optimal path. In the time-space network, $f^*(m)$ is the estimated cost of a path that starts from the virtual departure node σ, passes through the node m, and arrives at the virtual arrival node τ. $g(m)$ is the actual cost of the path from σ to m. $h^*(m)$ is the estimated cost of the best path from m to τ. $h(m)$ is the actual cost of the best path from m to τ. The key to the A$*$ algorithm is to select an appropriate estimation function. Since $g(m)$ is the actual cost of the path from the virtual departure node to the node m along the selected path, the problem involves the selection of an appropriate heuristic function $h^*(m)$ to achieve a quick and exhaustive search.

If the estimation function satisfies the admissibility condition [i.e., if the number of nodes subsequent to each node in the network is limited and $h^*(m) \leq h(m)$ is satisfied for each node m], then the best path for the original problem can be derived via the A$*$ algorithm [29]. Therefore, we reverse the original time-space network. Then, $h^*(m)$ can be derived as the cost of the shortest path from the virtual arrival node τ to a random node m, as calculated via the SPFA. If (12) is not considered, then $h^*(m) = h(m)$, and the search for the best path will be conducted exactly along the shortest path. All trains are assumed to stop at all stations after node m. We construct pairs from each station i' after node m and from each stop station i before node m and add the corresponding dual variables $\eta_{i,i'}$ to $h^*(m)$. The $h^*(m)$ derived in this manner must satisfy $h^*(m) \le h(m)$. As noted in Section 2.4, the number of nodes subsequent to each node in the time-space network is limited, and, thus, the admissibility condition is satisfied. Therefore, the A$*$ algorithm can be used to derive the shortest path for the original problem.

The main procedure for solving the PP is as follows:

Step 1. The weights of the time-space arcs are updated using the dual variables derived from the RMP. The set $G^{\min}$ is used to store the calculated shortest path for each train. G_j represents the set of time-space paths for train j. A status priority queue $\{(m, g(m), f^*(m))\}$ is constructed. $cnt(m)$ denotes the number of times node m leaves the queue. $S(m)$ denotes the set of stop stations prior to node m. $G^{\min} = \varnothing$ and $j = 1$.

Step 2. ${G_j}'$ is derived through a reversal of G_j; in other words, all time-space arcs in G_j are reversed. In addition, $\sigma' = \tau$ and $\tau' = \sigma$. The shortest path from node σ' in ${G_j}'$ to any node $m' = m$ is calculated using the SPFA. The result can also be regarded as the cost of the shortest path from node m in G_j to τ. This cost is recorded as $h'(m)$. The calculation proceeds to Step 3.

Step 3. All items are cleared from the status priority queue. $cnt(\sigma) = 0$, $cnt(\tau) = 0$, $S(\sigma) = \varnothing$, and $S(\tau) = \varnothing$. The initial status $(\sigma, 0, h^*(\sigma))$ is added to the queue.

Step 4. The status priority queue is sorted in increasing order of the estimated cost $f^*(m)$. The lowest status $(u, g(u), f^*(u))$ is removed from the queue. $cnt(u) = cnt(u)+1$. If the node u is equal to τ and $cnt(u) = 1$, then the shortest path for train j is the output, and the calculation proceeds to Step 6; otherwise, it proceeds to Step 5.

Step 5. The statuses of all nodes subsequent to node u are calculated using a status transfer function and are added to the status priority queue. v represents an adjacent node subsequent to node u . $w[u][v]$ represents the weights of the arcs of the adjacent nodes (u, v). The status transfer function is

$$\begin{aligned}&(u, g(u), f*(u))\\&\longrightarrow \big(v, g(u) + w[u][v] + g'(v), g(u) + w[u][v]\\&\quad + g'(v) + h^*(v)\big).\end{aligned} \tag{22}$$

$g'(v)$ is derived from the set $S(u)$ and the stop status of the current arc (u, v). If the current arc is a section arc and the associated train stops at the ending station of the section, then we select each station k from $S(u)$ to form an OD pair with v. The corresponding dual variable $\eta_{k,v}$ is added to $g'(v)$, and $S(v) = S(u)+v$; otherwise, $g'(v) = 0$ and $S(v) = S(u)$. The train is assumed to stop at all stations after node v. Each station i from the set $S(v)$ and each station i' from the stations after node v form an OD pair, and the dual variable $\eta_{i,i'}$, which corresponds to this pair, is added to the $h'(v)$ calculated in Step 2. The result $[h^*(v)]$ is the estimated cost of the shortest path.

If $cnt(v) > 1$, then the calculated status is not added to the status priority queue. The calculation proceeds to Step 4.

Step 6. The shortest path for train j is added to $G^{\min}$. $j = j + 1$. If $j > n$, the calculation proceeds to Step 7; otherwise, it proceeds to Step 2.

Step 7. All train paths in $G^{\min}$ are ordered according to their costs. The path with the lowest cost is added to the RMP. The calculation ends.

4.4. Rapid Branching Strategy. We use a rapid branching strategy, in which only one branch is taken. The selected path, which is represented by a fractional variable, will directly appear in the next iteration. The corresponding train will choose and is allowed only to choose this path. The beam search algorithm is used for branching. The derived fractional variables are ordered from greatest to least according to their corresponding coefficients in the objective function. At each branching, only a certain number of promising variables are selected; this number is referred to as the beam width. This method has been demonstrated to be effective for a similar integer programming problem [30].

Figure 7 illustrates the workflow of the extended branch-and-price algorithm, where ε is the termination parameter.

4.5. Acceleration Strategies

4.5.1. Initial Solution Iteration. In the branch-and-price algorithm, new columns generated by solving the PP are iteratively added to the path set considered in the RMP to improve the objective value and to achieve the optimal result. As the iterations proceed, the scale of the path set considered in the RMP increases. An excessively large problem scale will affect the efficiency of the algorithm. Thus, strategies are necessary to limit the increase in the scale of the path set. In general, a reasonable initial feasible solution facilitates rapid convergence of the algorithm.

For these reasons, we designed an initial solution iteration strategy. As shown by the dotted box in Figure 7, once an integer solution is derived in the calculation process and the numerical difference between the corresponding objective function value and the current UB exceeds a certain threshold, the integer solution will be used as a new initial

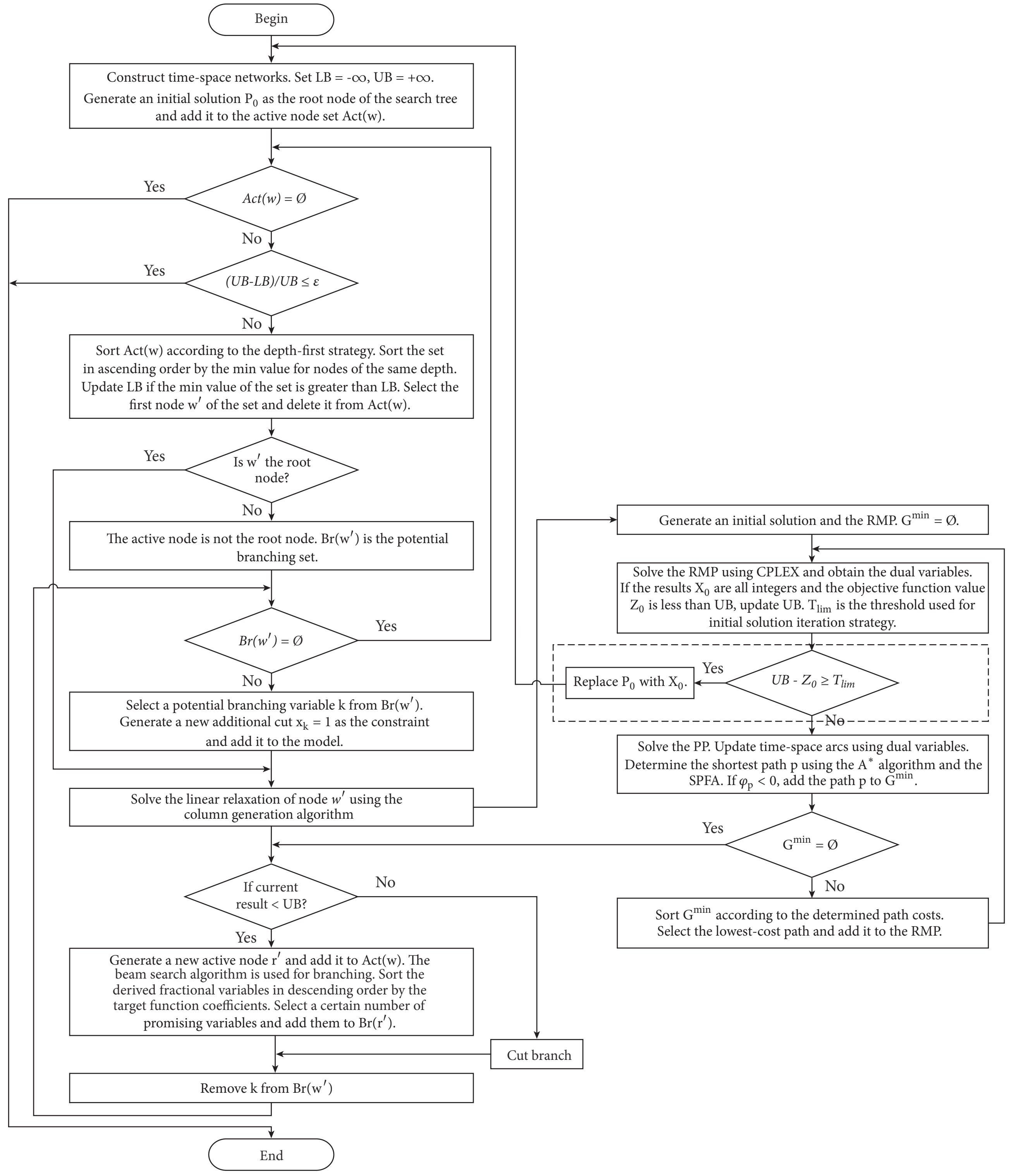

Begin
Construct time-space networks. Set LB = -∞, UB = +∞. Generate an initial solution P_0 as the root node of the search tree and add it to the active node set Act(w).
Yes
$Act(w) = \emptyset$
No
Yes
$(UB\text{-}LB)/UB \leq \varepsilon$
No
Sort Act(w) according to the depth-first strategy. Sort the set in ascending order by the min value for nodes of the same depth. Update LB if the min value of the set is greater than LB. Select the first node w′ of the set and delete it from Act(w).
Yes
Is w′ the root node?
No
The active node is not the root node. Br(w′) is the potential branching set.
Yes
$Br(w') = \emptyset$
No
Select a potential branching variable k from Br(w′). Generate a new additional cut $x_k = 1$ as the constraint and add it to the model.
Solve the linear relaxation of node w' using the column generation algorithm
If current result < UB?
No
Yes
Generate a new active node r′ and add it to Act(w). The beam search algorithm is used for branching. Sort the derived fractional variables in descending order by the target function coefficients. Select a certain number of promising variables and add them to Br(r′).
Cut branch
Remove k from Br(w′)
End
Generate an initial solution and the RMP. $G^{min} = \emptyset$.
Solve the RMP using CPLEX and obtain the dual variables. If the results X_0 are all integers and the objective function value Z_0 is less than UB, update UB. T_{lim} is the threshold used for initial solution iteration strategy.
Replace P_0 with X_0.
Yes
$UB - Z_0 \geq T_{lim}$
No
Solve the PP. Update time-space arcs using dual variables. Determine the shortest path p using the A* algorithm and the SPFA. If $\varphi_p < 0$, add the path p to G^{min}.
Yes
$G^{min} = \emptyset$
No
Sort G^{min} according to the determined path costs. Select the lowest-cost path and add it to the RMP.

FIGURE 7: Workflow of the extended branch-and-price algorithm.

feasible solution to restart the execution of the entire branch-and-price algorithm. The purpose of specifying this threshold is to avoid consuming excessive time when frequently solving the linear relaxation of the root node. This initial solution iteration strategy improves the calculation efficiency and effectively reduces the processing scale.

4.5.2. Delayed Constraints.

4.5.2. Delayed Constraints. The delayed constraint strategy is specifically related to (8)-(11). In (8), as shown in Figure 8, (α, p) represents the section arc a and its corresponding path p. If path p exists in the current path set of the RMP, the time intervals between arriving at station C on path p and all adjacent paths must be greater than or equal to t_C^{arr}. The

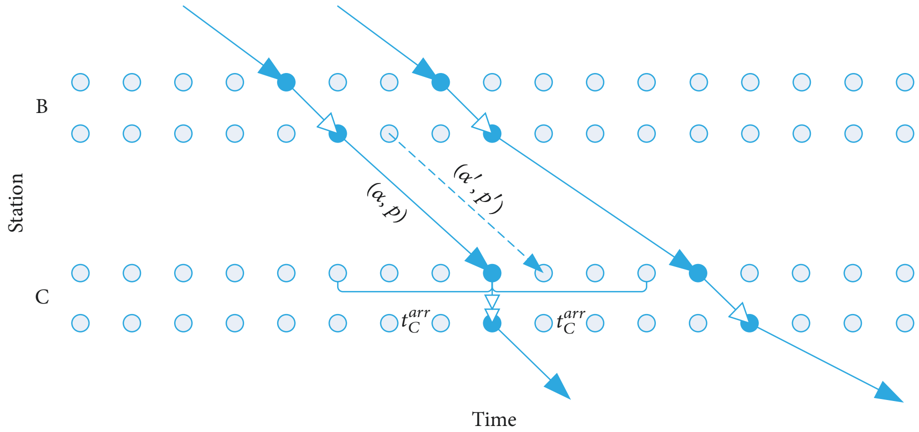


Figure 8: Illustration of the delayed constraint strategy.

dual variable of path p related to (8) is $\pi_{e,o,\alpha} = 0$. However, the weight of another arc (such as α') at station C in the time range $[t_{\alpha}^{arr}, t_{\alpha}^{arr} + t_C^{arr})$ is not affected by path p. If path p' is the shortest path selected by solving the PP and is added to the RMP, then path p and path p' must satisfy the constraint $x_p + x_{p'} \leq 1$ according to (8), and the dual variable of path p related to (8) may not be zero. In the calculation, the value of path p' may be less than 1 or even 0. In this case, the added path does not serve its ideal function in reducing the value of the objective function. Similarly, a time-space arc at station C in the time range $(t_{\alpha}^{arr} - t_C^{arr}, t_{\alpha}^{arr}]$ that is selected in the next iteration can conflict with path p. This constraint is referred to as a delayed constraint; the effect of a constraint in the current iteration can be shifted to the next iteration. To address these delayed constraints, we add a delay weight to each time-space arc. The value of the delay weight is determined based on the average value of the paths over several recent iterations (e.g., seven iterations) in the RMP. When the time-space network is updated using dual variables, the delay weight attributes of the paths are added to all selected arcs at station C in the time range $(t_{\alpha}^{arr} - t_C^{arr}, t_{\alpha}^{arr} + t_C^{arr})$. Delayed constraints related to (9)-(11) are handled in a manner similar to that applied for (8).

Applying the delayed constraint strategy to (8)-(11) effectively improves the efficiency of the branch-and-price algorithm in solving the linear relaxation of the nodes.

4.5.3. Column Removal. Due to the rapid branching strategy, in which a selected branching path is treated as a constraint that is added to the model, the train associated with a selected branching path must select only this path. All other paths related to this train in the current path set can be deleted. Any paths in the path set that have operational conflicts with the branching path [(8)-(11)] can be deleted. This train can be skipped in the calculation of the PP. When the arc weights are updated, the weights of all arcs that conflict with the branching path are set to $+\infty$. Thus, these conflicting arcs are avoided in the calculation of the shortest path. This column removal strategy can effectively control the processing scale and improve the efficiency of the algorithm.

5. Case Study

The effectiveness of the proposed model and algorithm was demonstrated using train data from the Wuhan-Guangzhou railroad. The algorithm was implemented using a Windows 7 computer with an Intel Xeon E3 3.30-GHz CPU and 8 GB of RAM. The programming tools were Microsoft Visual Studio 2010 with the C# language and ILOG CPLEX 12.7.1.0.

The Wuhan-Guangzhou high-speed railroad is an abbreviated name for the Wuhan-Guangzhou section of the Beijing-Guangzhou-Shenzhen-Hong Kong high-speed railroad. The length of the railroad is 968 km (601 miles), and the route has a total of 18 stations. The Wulongquan East station is an overtaking station that does not provide passenger transportation. The Lechang East station officially opened in May 2017 and was treated as a "passed-by" station in this experiment. All trains that run on this route are classified as Grade G and operate at a speed of 300 km/h. The operation of Grade D trains, with a speed of 250 km/h, was officially suspended on January 10, 2016. Due to the time designated for maintenance and testing, the actual passenger service time ranges from 6:30 AM to 0:00 AM. Additional infrastructure information is available on the website: https://en.wikipedia.org/wiki/Wuhan-Guangzhou_high-speed_railway.

Table 1 lists the station names and the number of tracks at each station on the Wuhan-Guangzhou railroad. Table 2 lists the section lengths, pure section travel times, and additional starting and stopping times.

To explicitly show the train timetabling results, the calculation was performed using trains that travel in the downward direction as an example.

Table 3 lists the predefined earliest and latest departure times for the trains at their starting stations. Table 4 lists the OD service frequencies inferred from the daily average passenger traffic data provided by the ticketing department. The grades of the station nodes and the corresponding service frequencies determined based on the station size, properties, infrastructure, railroad hub position, city population, economic data, and traffic data are shown in Table 5. All data are provided by the State Key Laboratory of Rail Traffic Control

Table 1: Station names and number of tracks.

Station Name	Station ID	Number of Tracks		
		Downward direction	Upward direction	All
Wuhan	1	15	15	20
Wulongquan East	2	2	4	4
Xianning North	3	3	6	6
Chibi North	4	2	4	4
Yueyang East	5	7	7	7
Miluo East	6	2	4	4
Changsha South	7	16	12	16
Zhuzhou West	8	4	7	7
Hengshan West	9	2	4	4
Hengyang East	10	8	9	11
Leiyang West	11	2	4	4
Chenzhou West	12	6	6	6
Lechang East	13	4	2	4
Shaoguan	14	6	6	6
Yingde West	15	2	4	4
Qingyuan	16	2	4	4
Guangzhou North	17	4	4	4
Guangzhou South	18	22	22	28

Table 2: Section lengths and section travel times.

Adjacent Station Pair	Distance (km)	Section Travel Time (min)		
		Running Time	Additional Starting Time	Additional Stopping Time
1-2	48	12	2	3
2-3	37	8	2	3
3-4	42	9	2	3
4-5	82	17	2	3
5-6	70	15	2	3
6-7	68	14	2	3
7-8	40	10	2	3
8-9	68	14	2	3
9-10	41	9	2	3
10-11	56	12	2	3
11-12	76	17	2	3
12-13	83	18	2	3
13-14	47	10	2	3
14-15	68	15	2	3
15-16	59	12	2	3
16-17	37	8	2	3
17-18	46	13	2	3

and Safety (Beijing Jiaotong University, Beijing, China). The values of the parameters n_S, n_N, n_T, δ, ε, c_1^{pen}, c_2^{pen}, c_3^{pen}, and c_4^{pen} were set to 18, 107, 1440, 12 min, 0.055, 1.5, 1.5, 1.2, and 0.5, respectively. The values of $t_i^{\min}$, $t_i^{\max}$, t_i^{dep}, and t_i^{arr} for trains at all stations were set to 2 min, 7 min, 2 min, and 3 min, respectively.

Based on these data and parameters, calculations were performed using model M1 and the extended branch-and-price algorithm. To achieve a balance between efficiency and accuracy, we applied a termination rule in the form of (23). When the difference between the UB and the LB was within the allowable range, the calculation was terminated. An approximately globally optimal solution was achieved. We performed 30 sets of experiments, and the average computation time of the extended branch-and-price algorithm was 199 s.

$$\frac{(UB - LB)}{UB} \leq \varepsilon. \tag{23}$$

Table 3: Earliest and latest departure times at starting stations.

Train	Starting Station	Earliest Time	Latest Time	Train	Starting Station	Earliest Time	Latest Time	Train	Starting Station	Earliest Time	Latest Time	Train	Starting Station	Earliest Time	Latest Time
1	1	15:27	16:27	28	1	15:58	16:58	55	1	07:44	08:44	82	7	07:00	08:00
2	1	17:33	18:33	29	1	13:05	14:05	56	1	07:55	08:55	83	7	07:25	08:25
3	1	17:45	18:45	30	1	10:50	11:50	57	1	08:08	09:08	84	7	09:37	10:37
4	1	12:39	13:39	31	1	16:11	17:11	58	1	08:23	09:23	85	7	10:28	11:28
5	1	13:57	14:57	32	1	12:21	13:21	59	1	08:45	09:45	86	7	11:25	12:25
6	1	16:23	17:23	33	1	15:02	16:02	60	1	09:14	10:14	87	7	15:40	16:40
7	1	08:30	09:30	34	1	16:54	17:54	61	1	09:48	10:48	88	7	18:35	19:35
8	1	13:50	14:50	35	1	15:39	16:39	62	1	10:30	11:30	89	7	18:57	19:57
9	1	12:52	13:52	36	1	16:40	17:40	63	1	10:39	11:39	90	7	07:00	08:00
10	1	09:29	10:29	37	1	17:17	18:17	64	1	11:31	12:31	91	7	07:00	08:00
11	1	13:22	14:22	38	1	18:16	19:16	65	1	11:55	12:55	92	7	07:10	08:10
12	1	14:13	15:13	39	1	11:05	12:05	66	1	12:14	13:14	93	7	07:40	08:40
13	1	11:49	12:49	40	1	19:53	20:53	67	1	13:29	14:29	94	7	09:00	10:00
14	1	09:43	10:43	41	1	07:00	08:00	68	1	14:37	15:37	95	7	11:49	12:49
15	1	07:00	08:00	42	1	09:00	10:00	69	1	14:50	15:50	96	7	12:28	13:28
16	1	14:55	15:55	43	1	10:23	11:23	70	1	15:18	16:18	97	7	14:49	15:49
17	1	12:08	13:08	44	1	11:38	12:38	71	1	16:47	17:47	98	7	17:24	18:24
18	1	19:20	20:20	45	1	13:12	14:12	72	1	17:04	18:04	99	7	19:23	20:23
19	1	20:30	21:30	46	1	14:07	15:07	73	1	17:26	18:26	100	7	20:30	21:30
20	1	12:32	13:32	47	1	15:06	16:06	74	1	17:54	18:54	101	7	20:36	21:36
21	1	12:59	13:59	48	1	16:28	17:28	75	1	18:05	19:05	102	7	20:36	21:36
22	7	07:00	08:00	49	1	11:12	12:12	76	1	18:55	19:55	103	7	20:36	21:36
23	7	14:00	15:00	50	1	07:00	08:00	77	1	18:43	19:43	104	5	10:35	11:35
24	1	10:10	11:10	51	1	07:00	08:00	78	1	18:30	19:30	105	5	16:45	17:45
25	1	15:48	16:48	52	1	07:00	08:00	79	1	19:02	20:02	106	1	07:00	08:00
26	1	18:24	19:24	53	1	07:12	08:12	80	7	14:30	15:30	107	7	12:53	13:53
27	1	09:36	10:36	54	1	07:26	08:26	81	7	07:00	08:00				

TABLE 4: OD service frequencies of sections that operate in the downward direction.

Stations	1	3	4	5	6	7	8	9	10	11	12	14	15	16	17	18
1	—	14	11	18	9	36	14	7	17	8	21	20	5	6	10	32
3		—	4	7	4	14	7	2	9	3	10	9	3	3	4	12
4			—	4	5	11	6	2	7	4	7	7	2	3	4	10
5				—	4	18	7	5	9	4	13	11	5	5	6	16
6					—	8	3	3	3	3	7	7	2	2	3	9
7						—	16	14	21	13	28	28	7	9	14	43
8							—	2	9	8	11	13	4	5	7	16
9								—	3	7	9	12	4	5	3	14
10									—	5	16	12	7	7	8	21
11										—	4	12	4	4	5	13
12											—	16	7	8	10	28
14												—	4	6	7	28
15													—	2	2	7
16														—	2	9
17															—	14
18																—

TABLE 5: Classification of node grades and corresponding station service frequencies.

Level	Station ID									Service Frequency
1	7	18	1	12						60
2	14	10	5	3	8					30
3	9	11	4	6	16	15	17	2	13	14

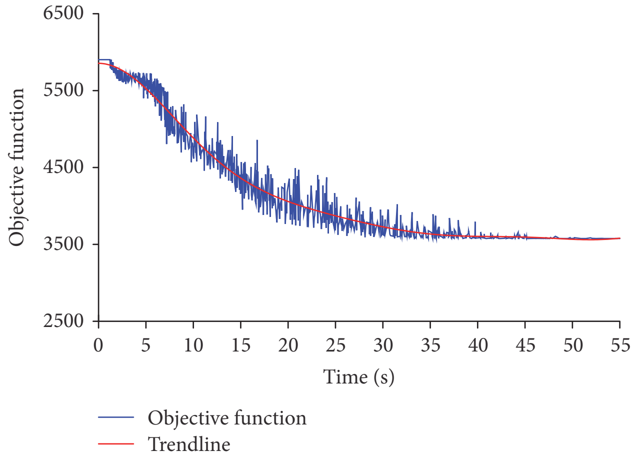


FIGURE 9: Variation in the objective function value of the root node over time.

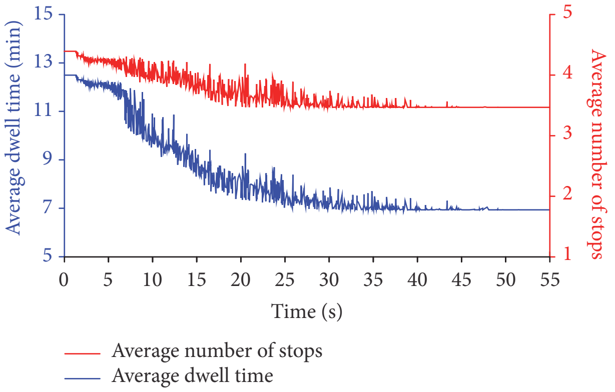


FIGURE 10: Variation in the parameters of the root node over time.

As shown in Figure 9, during the process of solving the linear relaxation of the root node using the column generation algorithm, the value of the objective function initially decreases rapidly and then gradually tends toward stability as the calculation time increases.

As Figure 10 similarly illustrates, as the calculation time increases, both the average number of stops and the average dwell time decrease and gradually tend toward stability.

Figure 11 shows a comparison of the predefined and actual numbers of train departures during different hour-long periods. As shown in the figure, a reasonable selection of values for the parameters δ and c_1^{pen} can ensure that the majority of trains depart within the predefined departure time ranges.

Table 6 presents a comparison between the real train timetable and the optimized train timetable in terms of the average travel speed, the average number of stops at stations, and the average station dwell time. The real case is the real train timetable for the Wuhan-Guangzhou high-speed railroad in February 2016 (refer to Dataset S1 in the Supplementary Materials), which was also employed as the initial solution for the optimal case. The timetable specified the operations of 107 downward-direction trains (from 6:30 AM to 0:00 AM). The minimum station dwell time was 2 min, and the maximum dwell time was 28 min. The average

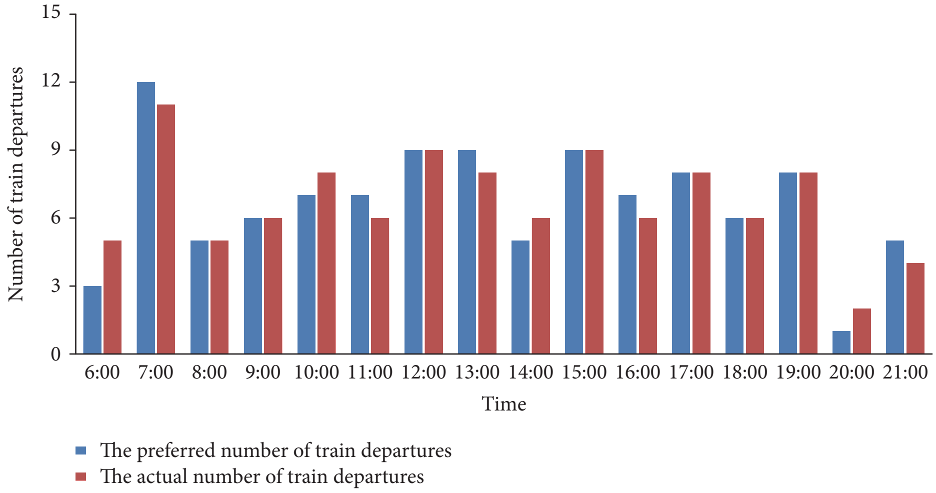


Figure 11: Preferred and actual numbers of train departures for each hour.

Table 6: Comparison between the real train timetable and the optimized train timetable.

Parameters	Real Case	Optimal Case	Change	Percent Change (%)
Number of trains	107	107	0	0
Average travel speed (km/h)	223.4	234.1	10.7	4.8
Average number of stops	4.4	3.5	-0.9	-20.3
Average dwell time (min)	12.5	7.1	-5	-43.2
Overtaking	40	0	-40	-100.0
Max dwell time (min)	28	6	-22	-78.6
Total cost of trains	5902.5	3645.7	-2257	-38.2
Average cost of trains	55.2	34.1	-21	-38.2
UB	—	3645.7	—	—
LB	—	3576	—	—
Computation time (s)	—	199.0	—	—
Gap	—	1.9%	—	—

travel speed was 223.4 km/h. The optimal case corresponds to the optimized train timetable, in which the number of trains, train types, and paths are the same as those in the real case. The gap value was calculated as follows:

$$\text{Gap} = \frac{(UB - LB)}{UB} \times 100\%. \tag{24}$$

As shown in Table 6, the average number of stops at stations and the average dwell time are reduced by 20.3% and 43.2%, respectively, in the optimized train timetable. The average travel speed is also significantly increased, and the total defined cost function of the trains is reduced by 38.2%. These results indicate that our proposed model and algorithm can effectively improve the quality of a constructed train timetable and the travel efficiency for passengers. Due to the more reasonable design of the train stop schedule plan, the number of occurrences of overtaking is reduced from 40 to 0, and the maximum dwell time is reduced by 78.6%. These improvements effectively reduce the workload for station operations and improve the operational capability of the station infrastructure. The total computation time of the optimal case was 199 s, and the gap value was 1.9%. The results demonstrate the efficiency of the proposed algorithm. We also tested the efficiency of the algorithm without the three acceleration strategies. In this case, the gap value was 50% when the calculation ran for 6 min. This result demonstrates that the three proposed acceleration strategies are effective in accelerating the algorithm. Figure 12 shows the train timetable from 6:30 AM to 0:00 AM based on the optimal case.

6. Conclusions

In this study, we developed a high-speed railroad train timetable optimization model based on a time-space network representation. The goal of the model is to minimize the deviations of the train start times, the total number of train stops, and the dwell time at stations, which are subject to the constraints related to the OD service frequency, train scheduling, and station service frequency. The train stop schedule plan is redesigned during the process of optimizing the train timetable. An experiment using data from the

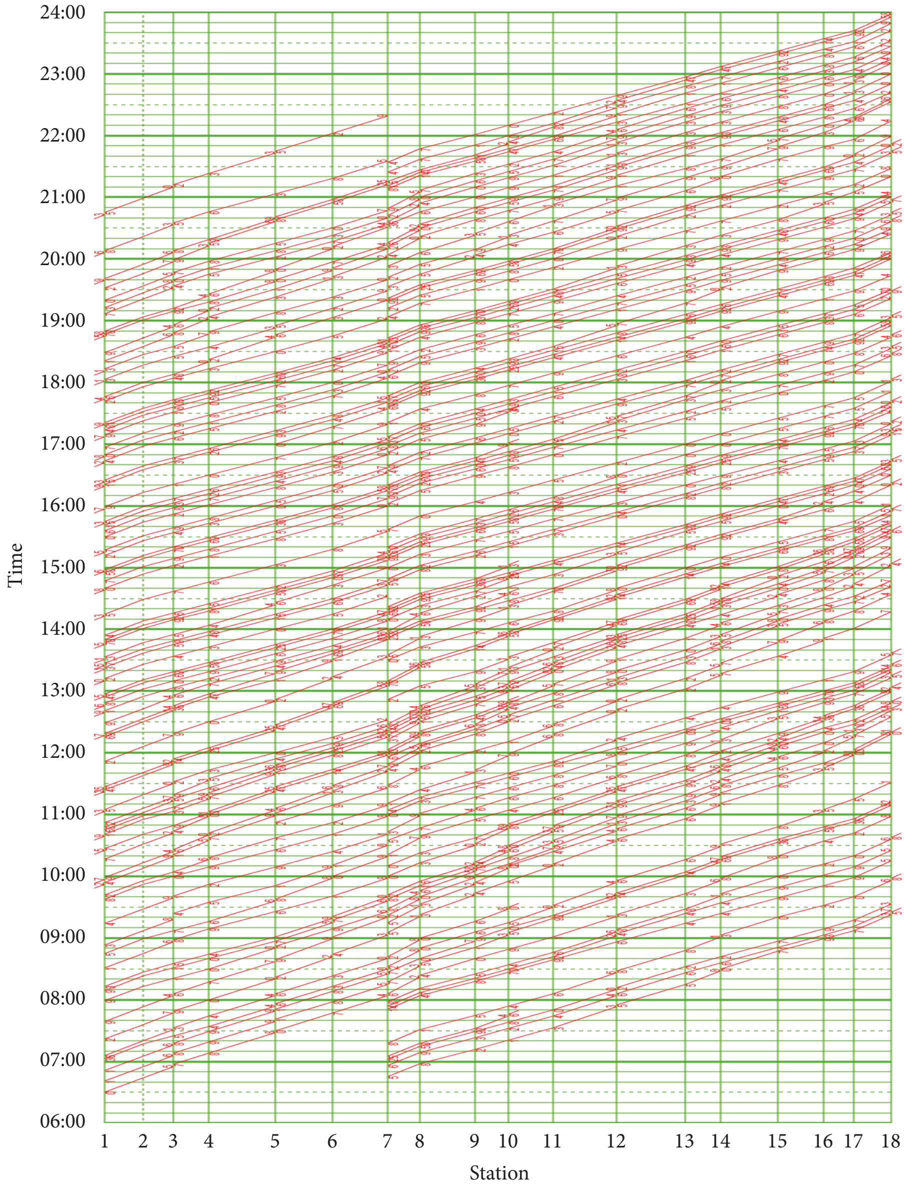


Figure 12: Optimized train timetable for the optimal case.

Wuhan-Guangzhou high-speed railroad demonstrated that the proposed model and algorithm can effectively reduce the number of stops at stations and the station dwell time to improve the quality of the constructed train timetables and the travel efficiency for passengers. First, we discussed the construction of a time-space network to represent a high-speed railroad environment. We used an associated labeling method to handle unconnected arcs. Second, we employed an extended branch-and-price algorithm to calculate the model. We also proposed a new method based on the A∗ algorithm and the SPFA to search for the shortest path in the unconnected and dynamic time-space network when solving the PP. Last, we designed three acceleration strategies: initial solution iteration, delayed constraints, and column removal. Our experiment demonstrated that these three strategies are effective in accelerating the algorithm.

Our future study will pursue three research directions. First, only constraints concerning the limited number of sidetracks at a station were considered when optimizing the train timetable; constraints related to routing schedules and track use at stations were not considered. Thus, addition work is needed to achieve the simultaneous optimization of the high-speed railroad train timetable and the station operation plan. Second, we plan to explore the impact of the OD/station service frequency on passenger demands. Last, we will perform sensitivity analyses to assess the effects of the punishment coefficients on the objective function.

Data Availability

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

Conflicts of Interest

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

Acknowledgments

This study was supported by National Key R&D Program of China (2018YFB1201402) and Technology Research and Development Program of China railway (2017X009-E). We thank Mingkai Bi and Yu Zhou for their constructive comments.

Supplementary Materials

The supplementary material of this study consists of a dataset that describes the real train timetable for the Wuhan-Guangzhou high-speed railroad in February 2016. *(Supplementary Materials)*

References

[1] O. Frank, "Two-Way Traffic on a Single Line of Railway," *Operations Research*, vol. 14, no. 5, pp. 801–811, 1966.

[2] B. Szpigel, "Optimal train scheduling on a single line railway," 1973.

[3] P. Serafini and W. Ukovich, "A mathematical model for periodic scheduling problems," *SIAM Journal on Discrete Mathematics*, vol. 2, no. 4, pp. 550–581, 1989.

[4] A. Schrijver and A. Steenbeek, "Dienstregelingontwikkeling voor Nederlandse Spoorwegen NS Rapport Fase 1," *Centrum voor Wiskunde en Informatica (Oktober 1993)*, 1993.

[5] M. A. Odijk, "A constraint generation algorithm for the construction of periodic railway timetables," *Transportation Research Part B: Methodological*, vol. 30, no. 6, pp. 455–464, 1996.

[6] K. Nachtigall and S. Voget, "A genetic algorithm approach to periodic railway synchronization," *Computers & Operations Research*, vol. 23, no. 5, pp. 453–463, 1996.

[7] C. Liebchen, "Symmetry for periodic railway timetables," *Electronic Notes in Theoretical Computer Science*, vol. 92, pp. 34–51, 2004.

[8] C. Liebchen, "Periodic timetable optimization in public transport," *Operations Research Proceedings 2006*, pp. 29–36, 2007.

[9] G. Caimi, M. Fuchsberger, M. Laumanns, and K. Schüpbach, "Periodic railway timetabling with event flexibility," *Networks*, vol. 57, no. 1, pp. 3–18, 2011.

[10] J.-H. Zhong, M. Shen, J. Zhang, H. S.-H. Chung, Y.-H. Shi, and Y. Li, "A differential evolution algorithm with dual populations for solving periodic railway timetable scheduling problem," *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 4, pp. 512–527, 2013.

[11] D. Sparing and R. M. P. Goverde, "A cycle time optimization model for generating stable periodic railway timetables," *Transportation Research Part B: Methodological*, vol. 98, pp. 198–223, 2017.

[12] U. Brännlund, P. O. Lindberg, A. Nõu, and J.-E. Nilsson, "Railway timetabling using Lagrangian relaxation," *Transportation Science*, vol. 32, no. 4, pp. 358–369, 1998.

[13] A. Caprara, M. Fischetti, and P. Toth, "Modeling and solving the train timetabling problem," *Operations Research*, vol. 50, no. 5, pp. 851–861, 2002.

[14] X. Zhou and M. Zhong, "Single-track train timetabling with guaranteed optimality: branch-and-bound algorithms with enhanced lower bounds," *Transportation Research Part B: Methodological*, vol. 41, no. 3, pp. 320–341, 2007.

[15] V. Cacchiani, A. Caprara, and P. Toth, "A column generation approach to train timetabling on a corridor," *4OR*, vol. 6, no. 2, pp. 125–142, 2008.

[16] B. He, R. Song, S. He, and Y. Xu, "High-speed rail train timetabling problem: A time-space network based method with an improved branch-and-price algorithm," *Mathematical Problems in Engineering*, vol. 2014, Article ID 641562, 15 pages, 2014.

[17] A. Ceder, "Methods for creating bus timetables," *Transportation Research Part A: General*, vol. 21, no. 1, pp. 59–83, 1987.

[18] M. Peeters and L. Kroon, "Circulation of railway rolling stock: a branch-and-price approach," *Computers & Operations Research*, vol. 35, no. 2, pp. 538–556, 2008.

[19] W.-I. Zhou, F. Shi, Y. Chen et al., "Method of integrated optimization of train operation plan and diagram for network of dedicated passenger lines," *Journal of the China Railway Society*, vol. 2, 2011.

[20] M. Kaspi and T. Raviv, "Service-oriented line planning and timetabling for passenger trains," *Transportation Science*, vol. 47, no. 3, pp. 295–311, 2013.

[21] D. Canca, E. Barrena, E. Algaba, and A. Zarzo, "Design and analysis of demand-adapted railway timetables," *Journal of Advanced Transportation*, vol. 48, no. 2, pp. 119–137, 2014.

[22] Y. Wang, T. Tang, B. Ning, T. J. J. van den Boom, and B. De Schutter, "Passenger-demands-oriented train scheduling for an urban rail transit network," *Transportation Research Part C: Emerging Technologies*, vol. 60, pp. 1–23, 2015.

[23] H. Niu, X. Zhou, and R. Gao, "Train scheduling for minimizing passenger waiting time with time-dependent demand and skip-stop patterns: nonlinear integer programming models with linear constraints," *Transportation Research Part B: Methodological*, vol. 76, pp. 117–135, 2015.

[24] Y. Yue, S. Wang, L. Zhou, L. Tong, and M. R. Saat, "Optimizing train stopping patterns and schedules for high-speed passenger rail corridors," *Transportation Research Part C: Emerging Technologies*, vol. 63, pp. 126–146, 2016.

[25] W. Suh, K.-S. Chon, and S.-M. Rhee, "Effect of Skip-Stop Policy on a Korean Subway System," *Transportation Research Record Journal of the Transportation Research Board*, vol. 1793, no. 1, pp. 33–39, 2002.

[26] L.-B. Deng, F. Shi, and W.-I. Zhou, "Stop Schedule Plan Optimization for Passenger Train," *China Railway Science*, vol. 30, no. 4, pp. 102–107, 2009.

[27] L. Zheng, R. Song, S. W. He et al., "Optimization model and algorithm of skip-stop strategy for urban rail transit," *Journal of the China Railway Society*, vol. 31, no. 6, pp. 1–8, 2009.

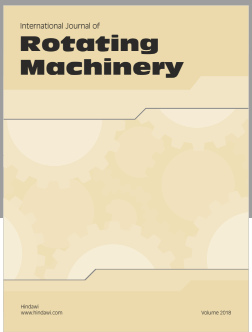
[28] D. Li, B. Han, X. Li et al., "High-speed railway stopping schedule optimization model based on node service," *Journal of the China Railway Society*, vol. 35, no. 6, pp. 1–5, 2013.

[29] N. J. Nilsson, "Principles of artificial intelligence," *Pattern Analysis & Machine Intelligence IEEE Transactions on*, vol. PAMI-3, no. 1, 112 pages, 1980.

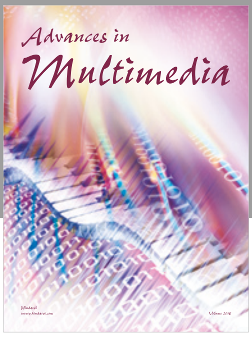
[30] T. Schlechte, "Railway Track Allocation," *Operations Research Proceedings*, vol. 203, no. 2, pp. 15–20, 2012.

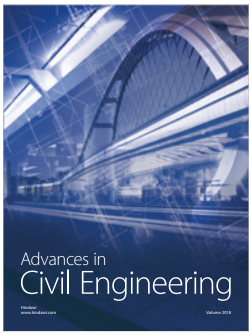
Journal of
Engineering

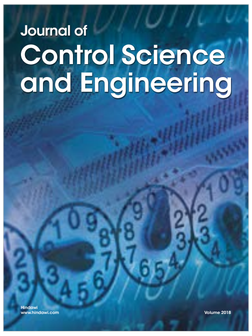
The Scientific
World Journal


International Journal of
Rotating
Machinery

Journal of
Sensors

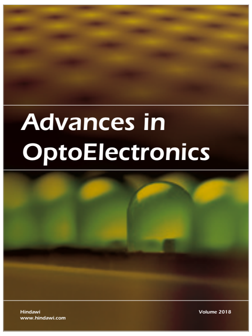

Advances in
Multimedia


Advances in
Civil Engineering


Journal of
Control Science
and Engineering

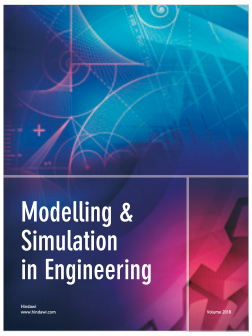
Journal of
Robotics

Journal of
Electrical and Computer
Engineering

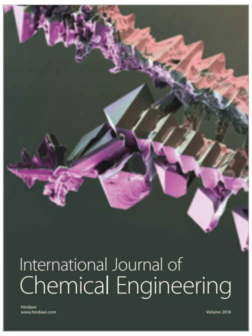

Advances in
OptoElectronics

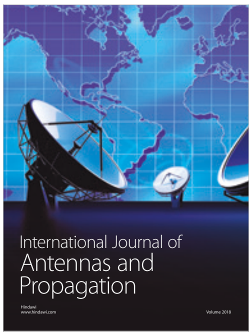
VLSI Design

International Journal of
Navigation and
Observation

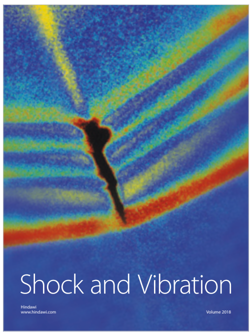

Modelling &
Simulation
in Engineering

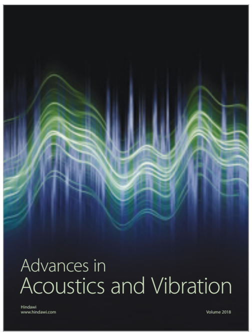
International Journal of
Aerospace
Engineering


International Journal of
Chemical Engineering


International Journal of
Antennas and
Propagation

Active and Passive
Electronic Components


Shock and Vibration


Advances in
Acoustics and Vibration