Research Article

High-Speed Rail Train Timetabling Problem: A Time-Space Network Based Method with an Improved Branch-and-Price Algorithm

Bisheng He, Rui Song, Shiwei He, and Yue Xu

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

Correspondence should be addressed to Rui Song; rsong@bjtu.edu.cn

A time-space network based optimization method is designed for high-speed rail train timetabling problem to improve the service level of the high-speed rail. The general time-space path cost is presented which considers both the train travel time and the high-speed rail operation requirements: (1) service frequency requirement; (2) stopping plan adjustment; and (3) priority of train types. Train timetabling problem based on time-space path aims to minimize the total general time-space path cost of all trains. An improved branch-and-price algorithm is applied to solve the large scale integer programming problem. When dealing with the algorithm, a rapid branching and node selection for branch-and-price tree and a heuristic train time-space path generation for column generation are adopted to speed up the algorithm computation time. The computational results of a set of experiments on China's high-speed rail system are presented with the discussions about the model validation, the effectiveness of the general time-space path cost, and the improved branch-and-price algorithm.

1. Introduction

The train timetabling problem (TTP) plays a critical role in the organization and operation of the railway system. TTP for the high-speed rail (HSR), which is different from conventional lines on operational requirements, brings a tremendous challenge for the operation company. For example, the frequency of high-speed train from Wuhan to Guangzhou in China is one hundred pairs for weekdays, compared to only ten pairs before the Wuhan-Guangzhou HSR was operated [1]. It can be observed that the scale of the problem is tenfold or more than before. Meanwhile, as there are more trains available in a day, passengers could design their trips on their need freely. The ridership fluctuates at each hour in a day referring to [2]. Hence, HSR service frequency at different hours in a day should accommodate passenger volumes like that in the urban transit system. Furthermore, high frequency service of peak hour brings a high infrastructure capacity utilization ratio. If the service plans, especially the stopping plans, are implemented rigidly, it may lead to trains' extra delay. An adjustment of stopping plan of trains within

an acceptable range could be adopted to make the system more efficient.

TTP schedules the movement of trains on tracks in the railway system, and our research focuses on the offline timetable optimization problem. It should be noted that the relevant optimization problems of TTP like the train scheduling problem, the train dispatching problem, and the track allocation problem will also be reviewed in the following. TTP has attracted many researches in these decades, which are usually classified into periodic train timetabling problem and nonperiodic timetabling problem according to the operation mode.

Periodic timetable is applied in some European countries like Germany and The Netherlands, due to the fact that it is easy to be used, understood, and operated. After Serafini and Ukovich introduced the concept of the Periodic Event Scheduling Problem [3], the method became a well-studied model for the periodic timetabling problem, such as Goverde [4], Peeters [5], Liebchen and Peeters [6], and Kroon et al. [7]. However, in other countries and regions, periodic train timetabling is not widely used by the railway companies.

Schlechte held the view that it was less efficient with the more and more helpful real-time information system [8]. Furthermore, he thought that the periodic timetable cannot accommodate the fluctuations of passenger volumes in a day. The period timetable does not suit the train timetabling problem of HSR.

A wide range of studies has been devoted to the nonperiodic train timetable problem. The heuristic methods were addressed in a number of researches because of a short time for acquiring a feasible solution. However, solution quality of the heuristic approach cannot be guaranteed. Cai et al. proposed a local optimality criterion to resolve the meet and pass event in a single-track railway line. The conflict resolving measures assigned trains according to the traffic prediction [9, 10]. Higgins et al. formulated the train scheduling problem as a nonlinear mixed integer program and solved by the metaheuristics algorithms, including local search, genetic algorithms, and hybrid techniques [11, 12]. Lee and Chen presented a four-step heuristic procedure to determine train pathing and train timetabling [13].

A sequence optimization method inspired by the job shop problem is a commonly used method to TTP. The trains are considered as the tasks and the tracks are considered as the machines. This type of method can solve the train timetabling problem with the metaheuristics algorithms [14] or heuristic rules [15]. However, it cannot improve the solution quality.

Integer programming based formulation can guarantee the solution quality with less speed than the heuristic approaches. As the off-line timetabling problem is less sensitive to the time requirement, integer programming based formulations are a more popular approach for nonperiodic train timetabling. These approaches usually adopted the time-space network or the likely network to describe the trains' movement on the tracks, and designed an exact algorithm to obtain a more optimal solution. Brännlund et al. suggested an integer programming approach to determine a profit maximizing schedule. The profit was given by the departure time minus a per minute cost for unnecessary waiting along the track. The problem was solved with a Lagrangian relaxation approach in which track capacity constraints are relaxed. They assigned a price to each track and separated the problem into dynamic programs [16]. Caprara et al. proposed a linear integer programming model based on a graph theoretic representation of the TTP, considering the track capacity and operational constraints. Their model also aimed to maximize the total train profit and the profit determined by the (nonnegative) difference between the running times in the actual and ideal timetables. A heuristic Lagrange relaxation algorithm was designed in their method [17, 18]. The same model based on a graph theoretic was considered in Cacchiani's study where the authors proposed heuristic column generation algorithms for the train timetabling problem on a corridor, and their method was only proved to be effective only on a corridor of railway and uncertainty in the railway network [19]. Zhou and Zhong proposed a method that aimed to minimize the total train travel time. The problem was formulated as a generalized

resource-constrained project scheduling problem. A branch and bound method was adopted to obtain the optimal schedules. A beam search heuristic method and a Lagrangian relaxation technique were applied to speed up the update of upper bound [20]. Li et al. constructed a train routing model combined with a train scheduling problem in which the objective function was to minimize the total delay of all trains in the railway network. Their method focused on how to determine train route choice. When the routes of all trains were determined, the model was transformed into a train scheduling problem [21]. A directed hyper-graph formulation was addressed by Harrod to model multi-commodity flows or activity paths through constrained resources [22]. A time-space network was presented in his research, and the model was solved by ILOG Cplex. Lusby et al. thought of the number of possible routings and made it a very complicated problem and presented a set-packing model with a resource-based constraint to the train routing problem. They proposed a route time-space network, which described the movement of trains in the junctions with different speeds [23]. Schlechte considered the real world track allocation problem for the railway system. The time expanded train scheduling digraph was proposed to build coupling mathematical models by micro-macro transformations. Their model proposed the penalty for exceeding minimum travel time and additional stops. Then, a rapid branching, column generation technique, and the bundle method were combined to tackle large scale track allocation instances [8].

By comparing the above methods to the train timetabling problem, our study applies the time-space network to formulate the mathematical model. Note that minimizing the total train travel time and maximizing the total train profits are the commonly used objective functions in the previous train timetabling researches. However, it cannot handle the HSR requirements like the service frequency in different hours, stopping plan adjustment, and train priority. Furthermore, although branch-and-price is a promising method for the train timetabling problem [8, 23], the long computational time to get an integer solution should be improved.

Our study makes the following specific contributions. First, a new time-space network generation method is presented to describe trains' movement in stations in more detail and facilitate to adjust the train stopping plan and formulate the constraints in the model. Second, the general time-space path cost is proposed to consider the new situation of HSR. Minimizing the total train general time-space path cost aims to make the system more efficient by abiding the requirement of service frequency in different hours, stopping plan adjustments and train priority. Third, an improved branch-and-price (IBAP) algorithm integrating with column generation, heuristic time-space path generation, and rapid branch strategy is designed, which is suitable for handling large scale problems and speeds up the solving processes.

This paper is organized as follows. After presenting the necessary notation and describing the problems in detail of TTP and time-space network in Section 2, the mathematical model is rigorously formulated. In Section 3, an improved branch-and-price algorithm to solve the TPP is proposed.

A set of practical examples based on China's HSR are provided in Section 4 to demonstrate the validity of the proposed method. Finally, Section 5 summarizes our contributions and outlines the future research issues.

2. Model Formulation

2.1. Notation

Sets

R: Set of trains, indexed by r,

T: Set of time interval, indexed by τ,

S: Set of stations, indexed by s,

E: Set of sections, indexed by e,

$E_{s,\text{out}}$: Set of section outflow from station s, indexed by e,

$E_{s,\text{in}}$: Set of section inflow to station s, indexed by e,

N: Set of time-space nodes, indexed by n,

$N_{s,i}$: Set of time-space nodes at station s remarked as inbound, indexed by n,

$N_{s,o}$: Set of time-space nodes at station s remarked as outbound, indexed by n,

N_τ: Set of time-space nodes of which $\tau_n = \tau$, indexed by n,

A: Set of time-space arcs, indexed by a,

A_r: Set of time-space arcs of train r, indexed by a,

$A_{s,\text{pa}}$: Set of passing time-space arcs occurring at station s, indexed by a, and $A_{S,\text{pa}} = \bigcup_{s \in S} A_{s,\text{pa}}$,

$A_{s,\text{de}}$: Set of dwell time-space arcs occurring at station s, indexed by a, and $A_{S,\text{de}} = \bigcup_{s \in S} A_{s,\text{de}}$,

$A_{s,\text{wa}}$: Set of additional waiting time-space arcs occurring at station s, indexed by a, and $A_{S,\text{wa}} = \bigcup_{s \in S} A_{s,\text{wa}}$,

A_e: Set of time-space arcs occurring at section e, indexed by a, and $A_E = \bigcup_{e \in E} A_e$. Note that, $A_{r,e}$ represents the set of time-space arcs of train r occurring at section e. The same to $A_{r,s}$, $A_{r,s,\text{pa}}$, $A_{r,s,\text{de}}$, $A_{r,s,\text{wa}}$,

A_Φ: Set of dummy time-space arcs, indexed by a,

MW: Set of maintenance windows, indexed by mw,

P: Set of time-space paths, indexed by p,

P_r: Set of time-space paths of train r, indexed by p,

P_s: Set of time-space paths through station s, indexed by p,

$P_{s,\text{de}}$: Set of time-space paths through station s with a dwell time-space arc, indexed by p,

$P_{s,\text{wa}}$: Set of time-space paths through station s with an additional waiting time-space arc, indexed by p,

$P_{s,\text{pa}}$: Set of time-space paths through station s with a passing time-space arc, indexed by p.

Parameters

$t_{\text{mw},b}$: Beginning time of maintenance widow mw,

$t_{\text{mw},d}$: Ending time of maintenance widow mw,

$n_{a,b}$: Source time-space node of time-space arc a,

$n_{a,d}$: Sink time-space node of time-space arc a,

$t_{a,b}$: Departure time of time-space arc a,

$s_{a,b}$: Departure station of time-space arc a,

$t_{a,d}$: Arrive time of time-space arc a,

$s_{a,d}$: Arrive station of time-space arc a,

t_a: Running time of time-space arc a,

c_a: General cost of time-space arc a,

c_p: General cost of time-space path p,

$t_{p,e,b}$: Arrive time of section e for time-space path p,

$t_{p,e,d}$: Leave time of section e for time-space path p,

$t_{p,s,b}$: Arrive time of station s for time-space path p,

$t_{p,s,d}$: Departure time of station s for time-space path p,

r_p: The train which time-space path p belongs to,

d_r: Direction of train r. The direction is generally distinguished by up-bound and down-bound, like the west bound and east bound in American railway system,

d_p: Direction of time-space path p,

Φ_{sou}: Source dummy node,

Φ_{sin}: Sink dummy node,

$\beta_{a,p}$: 0-1 parameters with $\beta_{a,p} = 1$ if time-space arc a in time-space path p; otherwise, $\beta_{a,p} = 0$.

Variables

x_p: a 0-1 variable with $x_p = 1$ if train r_p is scheduled by time-space path p; otherwise, $x_p = 0$.

2.2. Train Timetable Based on Time-Space Network.
A train timetable is a document setting out information on service time to assist passengers to plan a trip and listing the time when a train is scheduled to arrive at and depart from specified locations. The train timetabling problem for HSR is an off-line optimization problem in our study. It does not need to schedule trains on each track. Therefore, physical railway network $G = (E, S)$ illustrated in Figure 1(a) is applied to describe the infrastructure information. Sections E are aggregated by tracks linked stations and stations S are aggregated by tracks in the station areas. Due to the fact that G is an aggregated network, the minimum headway time at stations or on sections is able to cope with relevant railway safety systems [8].

Time-space network is a familiar method for the slot allocation problem considering the spatial and temporal requirements. Time-space network shows the position of individual train in time and in space. It is very useful for understanding and describing the operation of trains. In time-space network models, time horizon T is discretized into small interval τ. τ is a trade-off between the model size and the accuracy. If τ is too long, such as 1 hour, it cannot precisely describe the movement of HSR trains. However, if τ is too short, such as 1 second or 5 seconds, it will
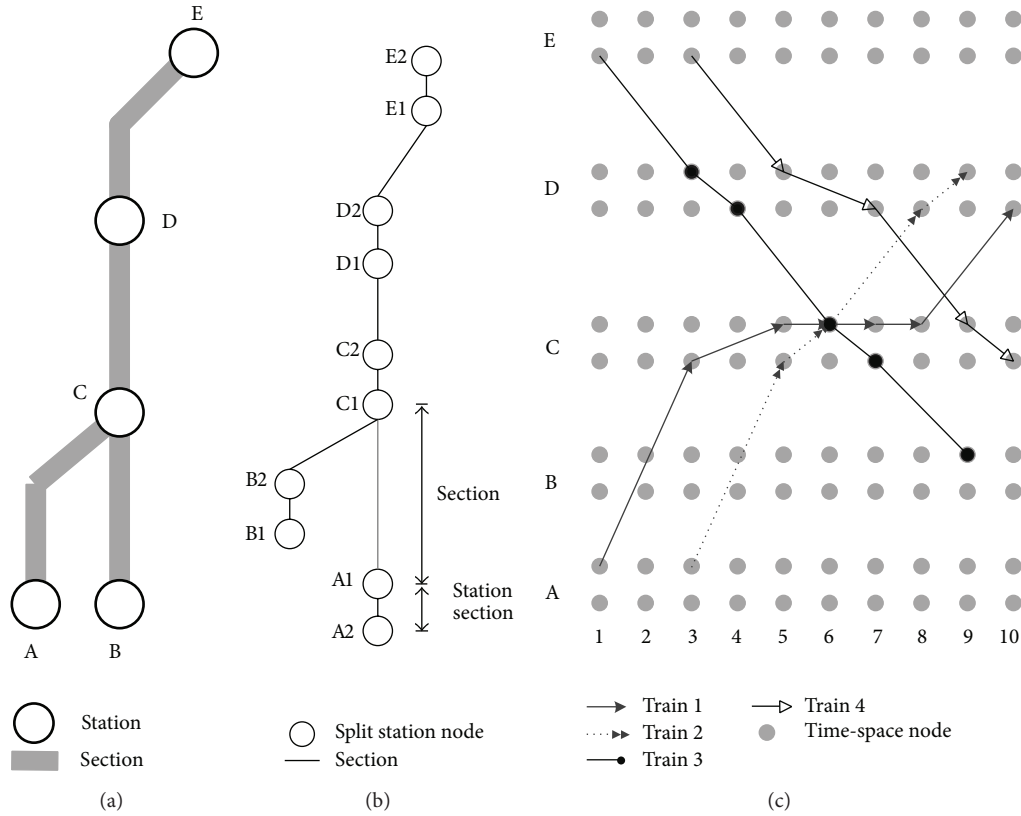
FIGURE 1: An illustration for the macroscopic railway network and the time-space network.

bring dramatically increased copied nodes, leading to a long computation time for such size problem. In our study, τ is set as 1 min so that the size of the problem is in an acceptable range and the movement of the trains can meet the need of real operational requirement at the same time.

It does not permit trains to stop on sections; that is, there are only time-space running arcs on sections which are adopted to describe the movement of trains. Train activities are more complicated in stations than on sections. Stations are split into two types of time-space node to specify the trains activities in more detail and are depicted in the Figure 1(c). To distinguish the time-space nodes in the same station, the inbound and outbound are remarked according to the down-bound direction without any other meanings. Train activities in stations can be divided into three types. The passing time-space arc inside a station specifies the train passing the station without stopping, while the dwell time-space arc inside stations is used to ensure the minimum dwell time at the station. Note that dwell time for trains dwelling at a station should be no less than a necessary time to load/unload the passenger to the trains and consider the additional arrival/departure times for the train movement. The additional waiting time-space arc is adopted to describe the stopping actives beyond the minimum dwell time.

Furthermore, if the stopping plan can be adjusted, especially in the peak hour, the total system may be more efficient. Adding and deleting stopping stations are the two adjustment measures. If train r stops at station s, it may bring a change for

other trains to overtake train r at station s which may decrease the delay for other trains. Moreover, adding stopping stations only brings little additional delay for the trains without disturbing the passengers' travel plan, while deleting stop stations may cause some passengers to be unable to board or get off the trains. Therefore, only adding stopping station will be applied in our study. However, along the passing stations of a train, not all the stations need to be changed. Therefore, the passing stations are divided in fixed stations and adjusting stations for each train, and the adjusting stations do not belong to the stopping plan stations. For the fixing station, if a train is scheduled through it without stopping, only the passing time-space arcs are generated. Otherwise, if the train is scheduled to stop at it, the dwell time-space arcs as well as the additional waiting time-space arcs are generated. For the adjusting stations, the passing, dwell, and additional waiting time-space arcs will be generated. As depicted in Figure 2, a train is scheduled to stop at station C, and the stopping plan at station D can be adjusted.

As the speed, stop plan, origin station, and destination station of each train are different, it should generate the time-space network $G_r^T = (N, A_r)$ with $A_r = A_{r,E} \cup A_{r,S,\text{pa}} \cup A_{r,S,\text{de}} \cup A_{r,S,\text{wa}} \cup A_{r,\Phi}$ for each train r. The total time-space network is $G^T = \bigcup_{r \in R} G_r^T$ with $A = \bigcup_{r \in R} A_r$. Figure 1 illustrates four trains in the same time-space network, and Figure 2 shows a time-space network for a train. Note that, a dummy sink and a source node are set to facilitate the flow conversation constraints.
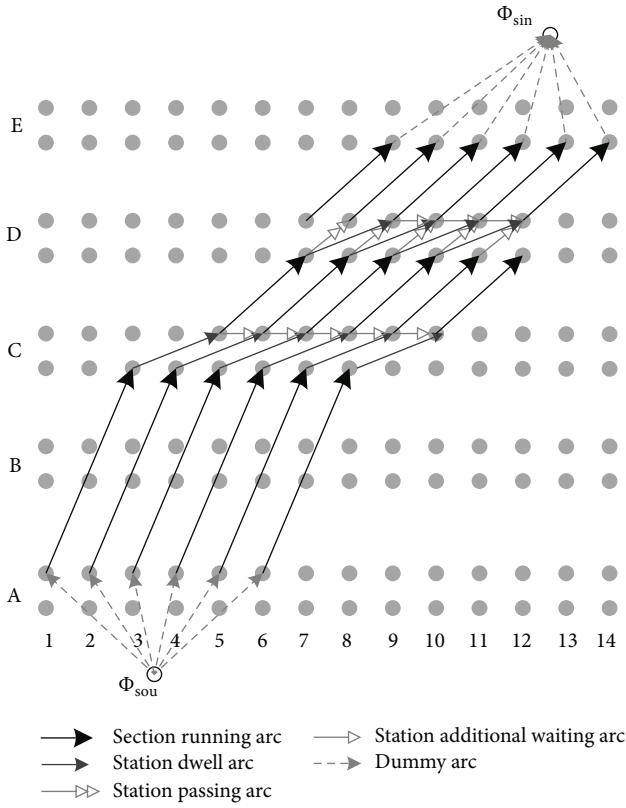
FIGURE 2: A time-space network for a train.

2.3. Optimization Objective Function. Many railway train timetabling methods addressed that the optimization objective in their approaches was to minimize the total train travel time. Nonetheless, it may be insufficient to ensure that each train is scheduled in a reasonable time-space path. For example, in Figures 3(a) and 3(b), the total travel time of train 1 and train 2 is equal. Comparing the two plans, train 1 stops at station D with 5 unit time in (a), 3 unit time in (b), and train 2 with 1 unit in (a), 3 in (b). Supposing that the situation in Figure 3(a) appears repeatedly, train 1 will always take the plan 1. It will increase a huge additional waiting time for train 1, but the total travel time of the train timetable does not change. Compared with other trains, it is obvious that train 1 lacks competitiveness due to too much waiting time. Therefore, the objective function should be improved to avoid the overtaking between same speed trains.

Moreover, there are different operational conditions and requirements between HSR and the conventional lines. Note that the train frequency in different hour should be set as diverse values to satisfy the passenger travel fluctuations in a day. Despite the fact that the stopping plan can be adjusted, that situation should occur as less as possible. In addition, trains are classified into different types with different priorities. The higher level trains have the higher priory than the lower ones. The high level trains are suggested not to be overtaken by the lower level trains. Given these reasons, a general time-space path cost is presented to acquire a high service level of timetable. Note that all the above requirements are like the soft constraints in the mathematical models. Hence, these requirements are transformed as the penalty functions. If the conditions on the variables are not satisfied, these are penalized in the objective function.

The running area in time-space network is introduced firstly. Supposing that a train departs at time τ from station A, the fastest time-space path for the train by its stop plan can be acquired. Hence, given a start time at the origin station and an end time at the destination for a train, a running area can be obtained like the gray areas shown in Figure 3(c).

Then, the general time-space path cost of the above requirements is introduced below.

(1) Service Frequency Requirement. For each train, there is a designed departure hour. Due to the operational rules or constraints, trains may not depart at designed departure hour and depart at the next hours actually. That is defined as the actual departure hour. The service frequency requirement is adopted to make trains depart at the designed departure hour as soon as possible. Three parameters relating to a train to generate three types of running areas in time-space network as shown in Figure 3 are defined: the designed hour beginning time $t_{r,e}$, the designed hour ending time $t_{r,l}$, and the reasonable departure time $t_{r,o}$. According to the three designed times, three fastest time-space paths can be acquired with the designed hour beginning time $t_{r,s,e}$, designed hour ending time $t_{r,s,l}$, and reasonable departure time $t_{r,s,o}$ in the passing stations of train r. The running area determined by the designed hour beginning time and reasonable departure time is defined as Area I in time-space network, which is not penalized. Train r traversing in Area I is considered as the desired state. The running area determined by the reasonable departure time and the designed hour ending time is defined as Area II in time-space network, which is penalized as $c_{r,II}$/min on the dwell, additional waiting and dummy source time-space arcs of Area II. Both Area I and Area II are adopted to enforce each train to depart in its designed departure hour. Area II is adopted to make trains arrive as early as possible and will make the system more efficient. The running area later than the designed hour ending time is called Area III in time-space network, which is penalized as $c_{r,III}$/min on the dwell, additional waiting and dummy source time-space arcs of Area III. The penalty value of $c_{r,III}$ is higher than $c_{r,II}$ because the trains are not expected to depart later than the designed hour. Therefore, different running areas in time-space network ensure the trains depart in the designed hour as soon as possible to satisfy the service frequency requirements in each hour. Note that the value of designed hour beginning time $t_{r,e}$ is set to the beginning time of the hour, and the designed hour ending time $t_{r,l}$ is set to ending time of the hour. Furthermore, the reasonable departure time $t_{r,o}$ is set to a time within the designed hour. The time is usually set to 30 min in an hour. It can enforce the trains to depart as early as possible in the designed departure hour.

(2) Stopping Plan Adjustment. As discussed before, adding stopping stations will be the measure to adjust stopping plan. However, adding stopping station will bring the trains with additional stopping time. Therefore, such measures should
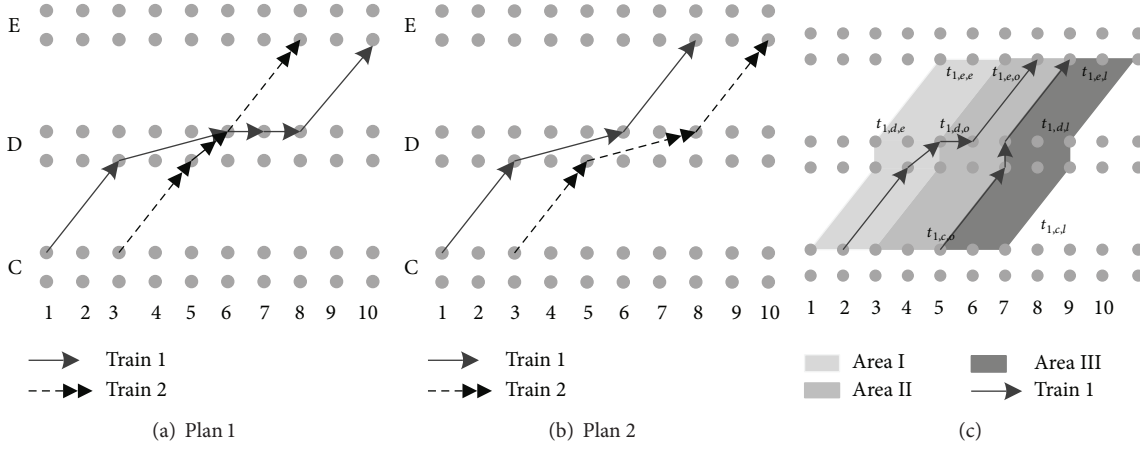
FIGURE 3: Two-train scheduling plan with same travel time and the illustration for time-space areas.

be penalized to avoid too many stopping plan adjustment occurring. Adding a stopping station is penalized $c_{r,a}$/min on the dwell and additional waiting time-space arc of the adjustment station.

(3) Priority of Train Types. Due to the priority of train types, the penalty for the higher level train is higher than the lower level train. Trains are classified into 300 km/h trains and 200 km/h trains according to their speeds. The additional waiting time-space arc for each train r by its type penalizes as $c_{r,300}$/min, $c_{r,200}$/min. The overtaking between the same speed trains should be avoided. Hence, additional waiting time-space arc is penalized c_{same}/min to make sure that the cost of additional waiting time-space arc is higher than the dwell time-space arc. That will make the cost of Plan 1 higher than Plan 2 and the overtaking will not occur between the same speed trains. Note that if the cost dwell time-space arc of one train is higher than the additional waiting time-space arc, the overtaking will also occur. That is because a train in Area III has a higher priority to pass.

The general cost of time-space arc c_a is calculated by accumulating the running time of time-space arc t_a with the penalty cost according to the conditions discussed before. The general cost of time path c_p can be expressed as

$$c_p = \sum_{e \in E} \sum_{a \in A_{r,e}} \beta_{a,p} c_a x_a + \sum_{s \in S} \sum_{a \in A_{r,s}} \beta_{a,p} c_a x_a. \tag{1}$$

2.4. Mathematical Model. Train timetabling problem based on time-space path (TTPTP) combined with optimization objectives and constraints is designed. TTPTP is a path-based network flow formulation.

(1) The Objective Function Is to Minimize the General Cost for All Time-Space Paths.

$$\sum_{r \in R} \sum_{p \in P_r} c_p x_p. \tag{2}$$

Except part of the requirements like the service frequency requirement, stopping plan adjustment, and priority

of train types, the constrains or business requirements for HSR train timetabling also include the safety requirements, maintenance time windows, station capacity requirement, minimum dwell time at station, and section running time. The time-space running arc guarantees section running time requirement. The safety requirements are realized by means of minimum headway time; refer to the research [8]. The minimum headway time means the occupational time for a train traversing on a section or station to gap the train between the other trains. In time-space network, the occupational time-space area can realize the minimum headway time as shown in Figure 4. Owing to that the railway system is a network in our study, not a corridor, the safety requirement includes minimum headway constraints on departure time at stations, minimum headway constraints on arrival time at stations, and minimum headway constraints on sections. Constraints of TTPTP are introduced in the following.

(2) Flow Conservation Constraints. For each train, there is only one time-space path to be utilized

$$\sum_{p \in P_r} x_p = 1, \quad r \in R, \tag{3}$$

where P_r is a set of all time-space paths for the train r.

(3) Station Capacity Constraints. This station capacity is realized by the number of parallel tracks num_s in station s. As there are so many parallel routes at throat section in the HSR stations, it can be seen that trains could route in or out of the station when the amount of trains holding in the station is less than num_s at time τ. Hence, the constraints are expressed as

$$\sum_{r \in R} \sum_{p \in P_{s,\tau}} x_p \leq \text{num}_s, \quad \forall s \in S, \ \tau \in T, \tag{4}$$

where $P_{s,\tau} = \{p \in P_{s,\text{de}} \cup P_{s,\text{wa}} \mid t_{p,s,b} \leq \tau, t_{p,s,d} \geq \tau\}$ represents the set of the time-space paths using the parallel tracks at time τ. Adding stopping stations will add a time-space path to $P_{s,\tau}$.
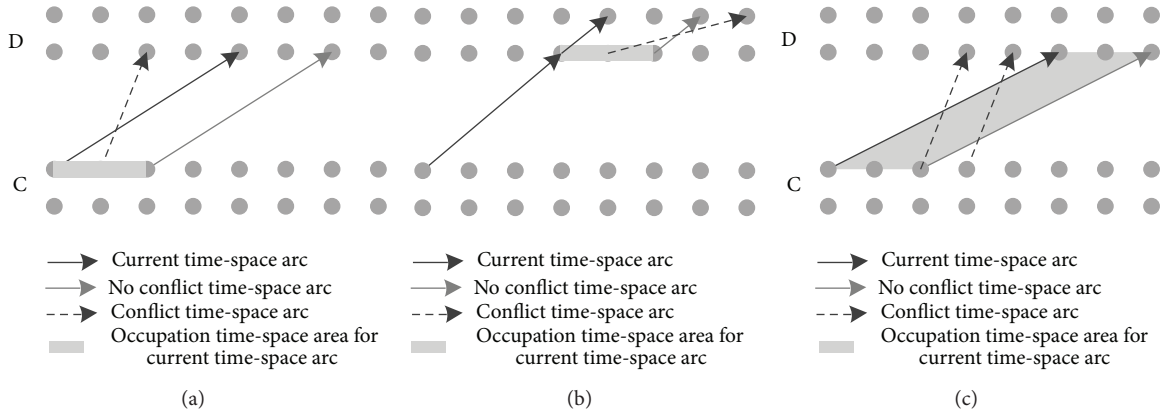
FIGURE 4: The illustration for the three types of headway constraints in time-space network.

Hence, Constraint (4) handles the changes by the stopping plan adjustment.

(4) Minimum Headway Constraints on Departure Time at Stations. These constraints enforce a minimum headway time $t_{s,\text{dep}}$ between the successive departure trains. The shadow area shown in Figure 4(a) is the occupational area for the first departure train and is prohibited for the running time-space arcs of other trains departing from the station.

$$x_p + \sum_{p' \in P_{s,e,p}} x_{p'} \le 1, \quad \forall s \in S, \ e \in E_{s,\text{out}}, \ p \in P_e, \quad (5)$$

where $P_{s,e,p} = \{p' \in P : d_{p'} = d_p, r_{p'} \ne r_p, t_{p,e,b} \le t_{p',e,b} \le t_{p,e,b} + t_{s,\text{dep}}\}$ is the set of time-space path which uses occupational time-space area and has conflicts with time-space path p.

(5) Minimum Headway Constraints on Arrival Time at Station. When a train arrives at a station, there is a minimum headway time $t_{s,\text{ari}}$ for the successive arrival trains. As illustrated in Figure 4(b), the occupational time-space area is prohibited for the passing time-space arcs and dwell time-space arcs of other trains inside station. Note that the station additional waiting time-space arc can use the area because the trains, which hold on the parallel tracks at the station and do not conflict with the arrival trains:

$$x_p + \sum_{p' \in P_{s,p}} x_{p'} \le 1, \quad \forall s \in S, \ p \in P_s, \quad (6)$$

where $P_{s,p} = \{p' \in P : d_{p'} = d_p, r_{p'} \ne r_p, a \in A_{s,\text{pa}} \cup A_{s,\text{de}}, \beta_{a,p'} = 1, t_{p,s,b} \le t_{a,b} \le t_{p,s,b} + t_{s,\text{ari}}\}$ express the set of time-space paths which uses occupational time-space area and has conflicts with time-space path p.

Note that both minimum headway constraints on arrival and departure time at stations ensure that one train can be overtaken by other trains at a station with the safety requirements.

(6) Minimum Headway Constraints on Sections. These constraints obtain the minimum headway time t_e between the

successive running trains on the section and the occupational time-space area is shown in Figure 4(c):

$$x_p + \sum_{p' \in P_{e,p}} x_{p'} \le 1, \quad \forall e \in E, \ p \in P_e, \quad (7)$$

where $P_{e,p} = \{p' \in P : d_{p'} = d_p, r_{p'} \ne r_p, t_{p,e,b} \le t_{p',e,b} \le t_{p,e,d} + t_e, t_{p',e,d} \le t_{p,e,d} + t_e\}$ is the set of time-space path that uses occupational time-space area and has conflicts with path p at section e.

(7) Maintenance Window Constraints. When a maintenance tack is going, the sections and stations of the maintenance task should be forbidden for any train to move in for safety consideration. As Figure 5(a) illustrated

$$\sum_{p \in P_{tw}} x_p = 0, \quad \forall tw \in TW, \quad (8)$$

where $P_{tw} = \{p \in P : e \in E_{tw}, t_{p,e,b} \ge t_{tw,b}, t_{p,e,d} \le t_{tw,d}\} \cup \{p \in P : s \in S_{tw}, \beta_{a,p} = 1, a \in A_{s,\text{pa}} \cup A_{s,\text{de}}, t_{a,b} \ge t_{tw,b}, t_{a,d} \le t_{tw,d}\}$ represent the set of time-space paths which use the maintenance time windows time-space area.

(8) Minimum Dwell Time at Stations Constraints. If a train, which passes a station without stopping by its stopping plan, has to stop at the station. The stopping time is enforced to be no less than the minimum dwell time. Therefore, it cannot use the additional waiting time-space arc in the station directly and must select the dwell time-space arc to obtain that requirement. As Figure 5(b) depicted

$$\sum_{p \in P_s} x_p = 0, \quad \forall s \in S, \quad (9)$$

where $P_s = \{p \in P : a \in A_{s,\text{pa}}, a' \in A_{s,\text{wa}}, \beta_{a,p} = \beta_{a',p} = 1, t_{a,d} = t_{a',b}\}$ collect all the time-space paths in which a passing time-space arc directly connects an additional time-space arc within one time-space path.

(9) Decision Variables. The decision variable is a 0-1 variable for each train:
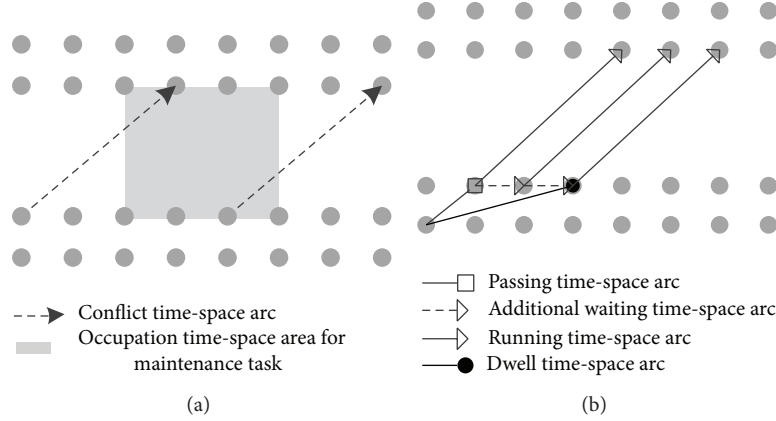
$$x_p = \{0, 1\}, \quad \forall p \in P. \quad (10)$$

FIGURE 5: The illustration for the maintenance window constraints and minimum dwell time at stations constraints in time-space network.

TABLE 1: The parameters for IBAP.

V: the set of nodes generated in branch-and-price tree, indexed by v	$X(v)$: the set of un-split x_p a node v
$\overline{V}$: the set of nodes unsearched in V, indexed by v	Ω: the set of time-space path, and $\Omega \subseteq P$
$Co(v)$: the set of all the constraints in a node v	$G_{r,\text{dual}}^T$: the time-space network with the value of dual variables
$UB(v)$: upper bound of a node v	η_p: reduce cost of time-space path p
$LB(v)$: lower bound of a node v	$LB = \min_{v \in V} LB(v)$: is the lower bound of branch-and-price tree
	$UB = \min_{v \in V} UB(v)$: is the upper bound of branch-and-price tree

Obviously, there are huge time-space arcs and nodes when time-space network is applied to TTP. Thus, the number of time-space path, which consists of time-space arc and node, is far beyond what we can enumerate. In the next section, a diligently designed algorithm is presented to solve the large-scale problem.

3. Improved Branch-and-Price Algorithm

The train timetabling optimization is a large-scale linear programming problem. At present, the commonly used applicable algorithm for that problem is branch-and-price (BAP) algorithm [24], which is combined by column generation and branch-and-bound (BAB) algorithm. BAB is implemented to obtain the integer solution, while column generation algorithm is used to solve the nodes in BAB search tree. Inspired by BAP, an improved branch-and-price (IBAP) is designed with some measures to speed up the solving process: (1) a rapid branching and node selection for branch-and-price tree; (2) a heuristic train time-space path generation for column generation.

3.1. Improved Branch-and-Price Algorithm. IBAP is a hybrid algorithm, which includes many producers. The flow chart of IBAP shown in Figure 6 can help to understand it in an overall view. Table 1 gives the parameters of IBAP. The detailed processes are introduced in Figure 6.

3.2. Branch-and-Price Search Tree. Branch-and-price search tree is similar to Branch-and-Bound tree. It is the framework

of the algorithm and is used to get the integer solution by adding the new cuts iteratively. Each node v in BAP search tree is an original TTPTP problem with new cuts. When $\overline{V}$ has no node v to solve, it means the search tree is implicit enumeration. Then, the algorithm terminates.

A rapid branching strategy just adds a specific type of cut into the nodes. A selected node is generally divided into left and right nodes with the cuts of $x_p = 0$ and $x_p = 1$ as x_p is a 0-1 variable. Note that the type of cut $x_p = 0$ changes the solution space slightly; however, fixing a $x_p = 1$ leads to a large set of time-space path that is not available to train r_p. Therefore, only the cuts of $x_p = 1$ are added into the search tree; that is, only the nodes v' with $x_p = 1$ and $Co(v)$ are added to $\overline{V}$.

There are a lot of $0 < x_p < 1$ in a linear solution. It will bring a large computational time if one sets all $0 < x_p < 1$ as the root nodes to branch. Let $X(v) = \{x_p \mid \lambda \leq x_p \leq 1\}$ denote branch node set. Only if x_p is greater than a specific value λ, the nodes with $x_p = 1$ will be generated.

Furthermore, an optimal gap $(UB - LB/UB)\%$ is used to terminate the algorithm to avoid too much time to get the solution $UB = LB$. When $(UB - LB)/UB \leq \varepsilon$, the algorithm stops.

3.3. Using Column Generation Algorithm to Solve Nodes in the Search Tree. TTPTP is a path-based network flow problem. As known that when a path-based network flow problem is relaxed as a linear programming, it can be solved by column generation effective referring to the research [24]. The idea of column generation is to enumerate a small subset
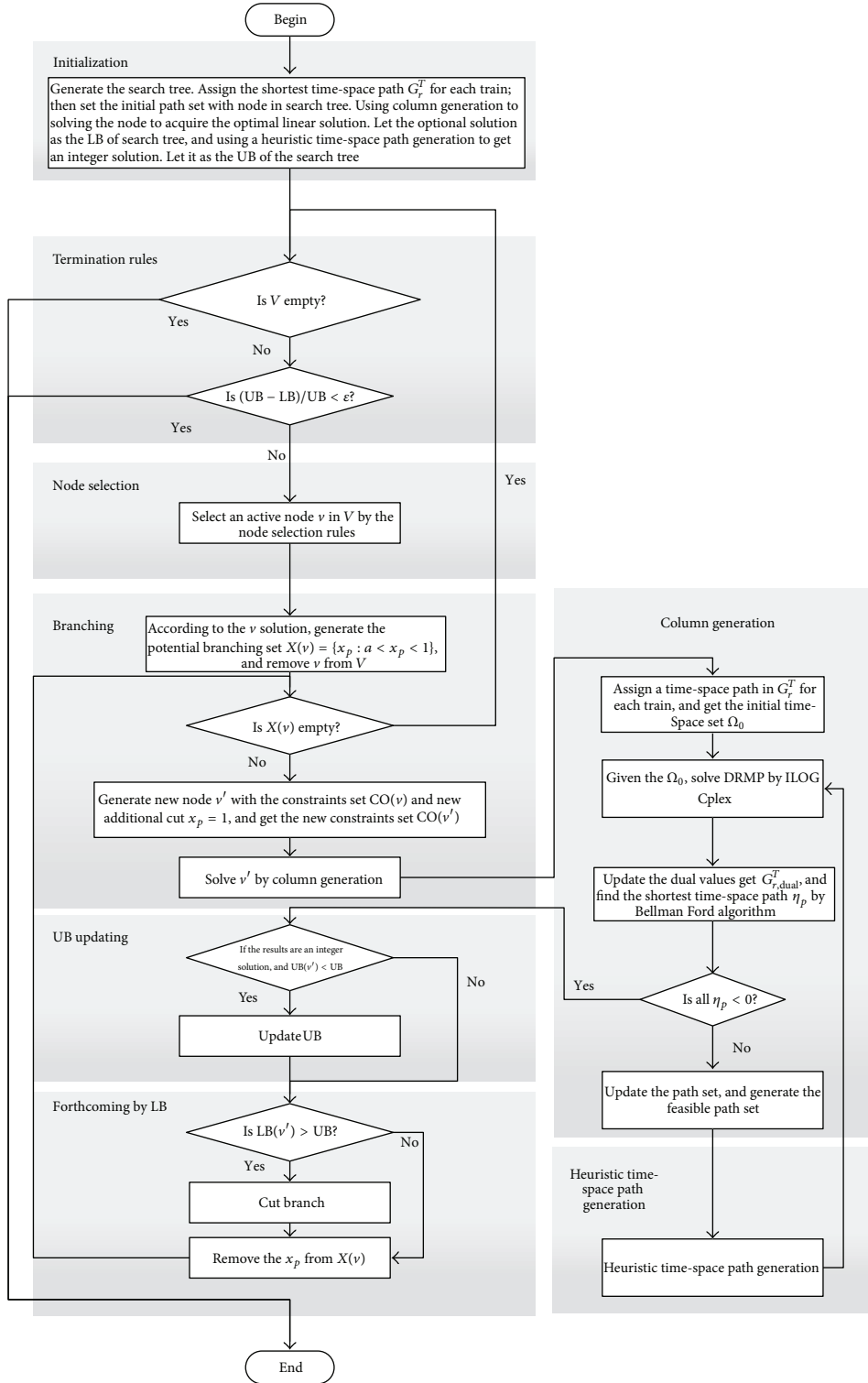
FIGURE 6: IBAP algorithm flow chart.

of the decision variables and then generate decision variables as needed. A restricted master problem (RMP) is solved iteratively by generating the subset of P. The price problems are used to generate new paths and determine whether the optimal solution is achieved or not with the values of the dual variables of RMP.

3.3.1. Restricted Master Problem. The integer variable is relaxed to continuous variables firstly; then a Ω which is a subset of P, is generated. $x_p = \{0, 1\}$ is reformed as continuous variables; that is $0 \leq x_p \leq 1$. Master problem for each node in BAB is called TTPTP (v) with $p \in P$. Then TTPTP (v) with continuous variables x_p and a subset

of $p \in \Omega \subset P$ is defined as restrict master problem (RMP). RMP is expressed as

$$
\text{(RMP)} \quad (2), (3), (4), (5), (6), (7), (8), (9) \quad \text{with}
$$
$$
0 \leq x_p \leq 1 \quad p \in \Omega. \tag{11}
$$

Using the duality of linear problem, the dual restricted master problem (DRMP) can be acquired as follows:

$$
\text{(DRMP)} \quad \max \sum_{r \in R} \lambda_r + \sum_{s \in S} \sum_{\tau \in T} \text{num}_s \pi_{s,\tau}
$$
$$
+ \sum_{s \in S} \sum_{e \in E_{s,\text{out}}} \sum_{p \in P_e} \zeta_{s,e,p} + \sum_{s \in S} \sum_{p \in P_s} \xi_{s,p} \tag{12}
$$
$$
+ \sum_{e \in E} \sum_{p \in P_e} \gamma_{e,p}
$$

$$
\sum_{p \in P_r} \alpha_{p,P_r} \lambda_r + \sum_{s \in S} \sum_{\tau \in T} \sum_{p \in P_{s,\tau}} \alpha_{p,P_{s,\tau}} \pi_{s,\tau}
$$
$$
+ \sum_{tw \in TW} \sum_{p \in P_{tw}} \alpha_{p,P_{tw}} \delta_{tw}
$$
$$
\sum_{e \in E_p} \gamma_{e,p} + \sum_{e \in E} \sum_{p' \in P_{e,p}} \alpha_{p,P_{s,p}} \gamma_{e,p'}
$$
$$
+ \sum_{s \in S_p} \xi_{s,p} + \sum_{s \in S} \sum_{p' \in P_{s,p}} \alpha_{p,P_{s,p}} \xi_{s,p'}
$$
$$
+ \sum_{e \in E_p} \zeta_{s,e,p}
$$
$$
+ \sum_{s \in S} \sum_{e \in E_{s,\text{out}}} \sum_{p' \in P_{s,e,p}} \alpha_{p,P_{s,e,p}} \zeta_{s,e,p'} \geq c_p
$$
$$
p \in \Omega \tag{13}
$$

$$
\lambda_r \geq 0, \quad \pi_{s,\tau} \geq 0, \quad \zeta_{s,e,p} \geq 0,
$$
$$
\gamma_{e,p} \geq 0, \quad \xi_{s,p} \geq 0, \quad -\infty \leq \delta_{tw} \leq +\infty, \tag{14}
$$

where λ_r, $\pi_{s,\tau}$, $\zeta_{s,e,p}$, $\gamma_{e,p}$, $\xi_{s,p}$ and δ_{tw} are the dual variables of Constraints (3), (4), (5), (6), (7), and (8), respectively. $\alpha_{p,P}$ is 0-1 parameters with $\alpha_{p,P} = 1$ if the path p is in the path set P; otherwise, $\alpha_{p,P} = 0$. The path set may be P_r, $P_{s,\tau}$, $P_{s,e,p}$, $P_{s,p}$, $P_{e,p}$, and P_{tw}. Note that Constraint (9) is transformed as a constraint in the price problem next in order to obtain the shortest time-space path. So the dual variables of Constraint (9) are not needed here.

3.3.2. Price Problem.
According to Constraint (12) and the Simplex Algorithm, the reduce cost of path p is expressed as

$$
\eta_p = c_p - \sum_{p \in P_r} \alpha_{p,P_r} \lambda_r - \sum_{s \in S} \sum_{\tau \in T} \sum_{p \in P_{s,\tau}} \alpha_{p,P_{s,\tau}} \pi_{s,\tau}
$$
$$
- \sum_{tw \in TW} \sum_{p \in P_{tw}} \alpha_{p,P_{tw}} \delta_{tw} - \sum_{e \in E_p} \gamma_{e,p} - \sum_{e \in E} \sum_{p' \in P_{e,p}} \alpha_{p,P_{s,p}} \gamma_{e,p'}
$$

$$
- \sum_{s \in S_p} \xi_{s,p} - \sum_{s \in S} \sum_{p' \in P_{s,p}} \alpha_{p,P_{s,p}} \xi_{s,p'} - \sum_{e \in E_p} \zeta_{s,e,p}
$$
$$
- \sum_{s \in S} \sum_{e \in E_{s,\text{out}}} \sum_{p' \in P_{s,e,p}} \alpha_{p,P_{s,e,p}} \zeta_{s,e,p'} \geq 0, \quad p \in \Omega. \tag{15}
$$

Constraint (15) is the reduce cost expression in the Simplex Algorithm. If each trains' cost of time-space path acquired by Constraints (15) is negative, that is $\eta_p \leq 0$, the method acquires the optimal solution and stops the procedure. However, if parts of train's value are positive, the method should solve further. Therefore, the value of lower bound of η_p should be acquired to judge whether the algorithm stops or not. Hence, Constraint (15) is shortest path problems in the network $G_{r,\text{dual}}^T$. $G_{r,\text{dual}}^T$ is acquired by adding the value of dual variables on corresponding time-space nodes and arcs in G_r^T. It is usually called price problem. When a new set of dual variables in DRMP is acquired, a new set of time-space paths for each train can be acquired. Bellman-Ford path search algorithm is adopted to get the shortest path, due to the dual variable with negative values and without a closed loop in the directed time-space network. If all the reduce costs in R are less than zero, there are no improving time-space paths and the algorithm terminates. Otherwise, the newly acquired shortest time-space paths are added into Ω and continue to solve RMP.

3.4. The Heuristic Time-Space Path Generation.
A heuristic time-space path generation is designed to avoid the degeneration in column generation algorithm, like the phenomenon of trapping in local optima in the neighborhood search algorithms. The heuristic time-space path generation can enlarge the neighborhood of the current solution. Then, it can improve the current integer solution with the more generated time-space paths.

The timetable can be described as a matrix {number of section × number of trains} by trains' sequence. As shown in Figure 1(c), there are four trains. The train sequence sorted by down-bound direction can be described as $\left\{ \begin{smallmatrix} 2 \\ 2 & 1 \\ 2 & 1 \\ 1 & 2 \end{smallmatrix} \right\}$ from section C to section D. Description that is more detailed can be seen in research [14].

Given a trains' matrix, a feasible timetable can be acquired by changing the sequence with a simple neighborhood search algorithm. Then, the decoding of the trains' matrix is used by a headway time path for each section and station.

The headway time path with artificial arc is defined to obtain a situation that there are no conflicts on each section. As shown in Figure 7(a), at each time-space arc's sink time-space node, the artificial arc is generated. The cost of artificial arc is set as the time dimension. The headway time path is generated as a directed path with the smallest value. Based on the headway time paths, the feasible timetable is generated by the method in Algorithm 1 and depicted in Figure 7.

Note that the number of parallel tracks at the station is not considered in the heuristic producer.
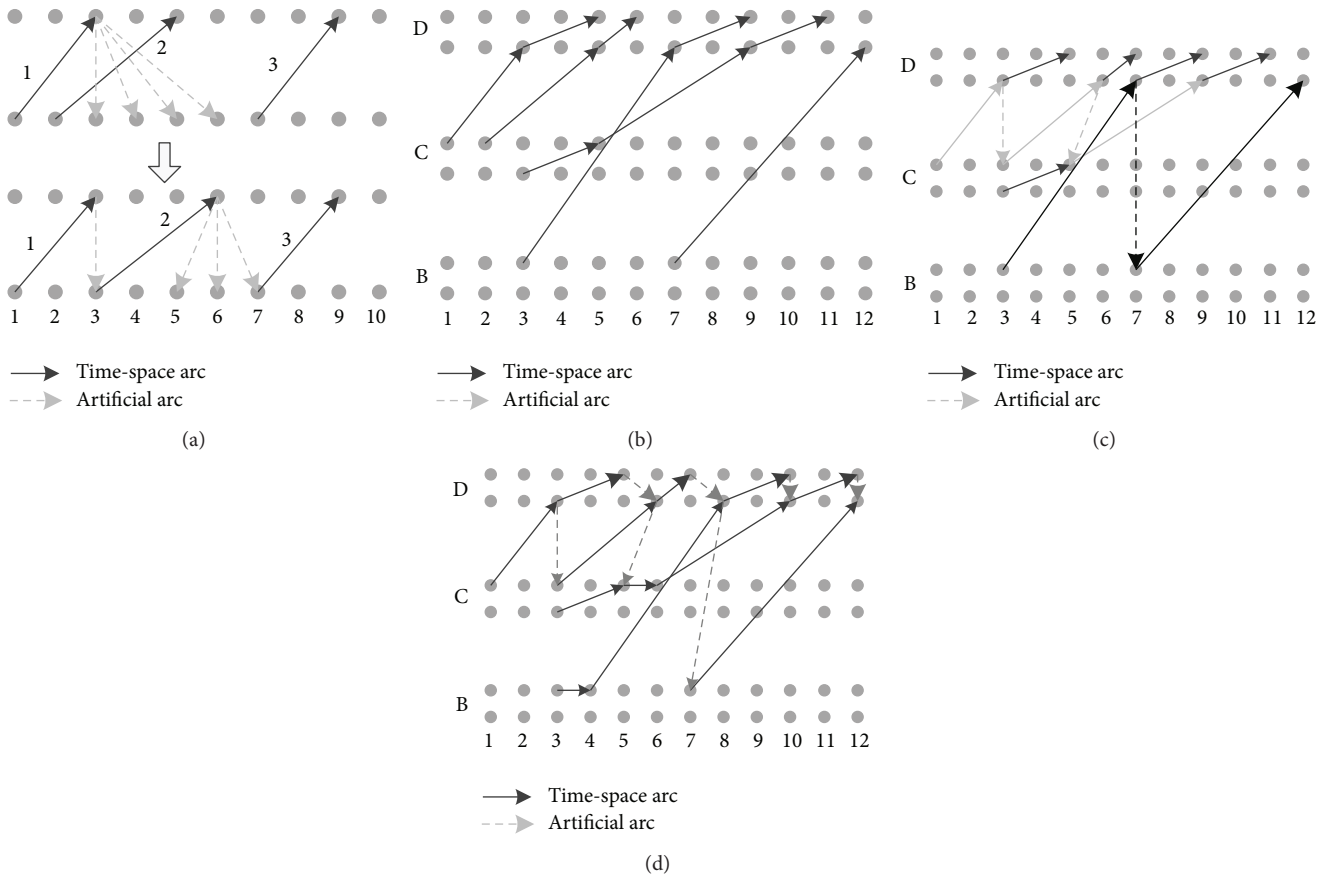
FIGURE 7: An example for heuristic time-space path generation.

Heuristic time-space path generation
Input: train matrix
(1) Assign a feasible time-space path for each train by trains' matrix with time-space arcs in the sections and stations of the train.
(2) **for** all section **do**
(4)   Get a headway path on each section and reassign time-space arc for each train
(5) Copy all stations into the adjustment station set
(5) **While** adjustment station set is not empty **do.**
(6)   Sorte the adjustment stations by the number of passing trains. Select the maximum number station.
     If more than one station's number of passing trains is the maximum number, the down-bound station is selected firstly.
     Moreover, if all the stations have the same direction, the station is selected by down-bound direction.
(7)   For the selected station, find a headway path. Reassign the time-space arc inside the station.
(8)   Adjust the connect section time-space arc and remove the station from adjustment station set.
(11) **end while**
**Output: the feasible timetable**

ALGORITHM 1

4. Computation Experiments

Three features are investigated through a series of computation experiments: (1) the time-space network validity for TTP, (2) the effectiveness of the general time-space path cost to meet the HSR operation requirements, and (3) the effectiveness of the IBAP algorithm in real-size instances. The train types are described above. The algorithm is implemented in Visual Studio 2008 platform with C# language and ILOG Cplex12.4 software for solving DRMP problem in column generation. The experiments are conducted on a PC with a 2.10 Ghz CPU, 2 GB RAM. ε is set to 0.95; λ is 0.75.

4.1. The Time-Space Network Validity and the Effectiveness of the General Time-Space Path Cost. A simple railway network shown in Figure 1(a) is used to verify whether the time-space
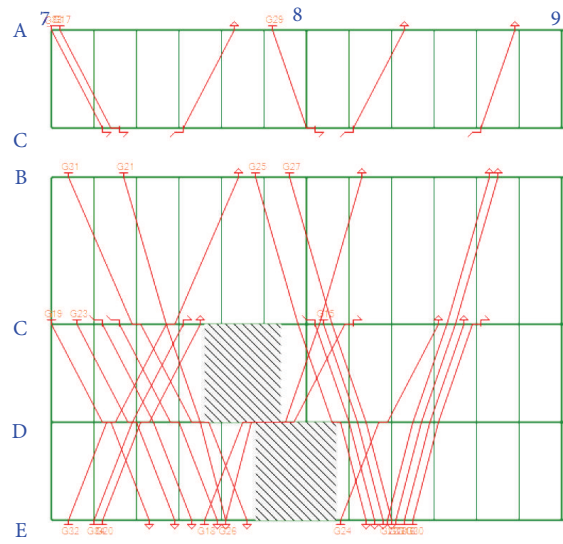
(a)
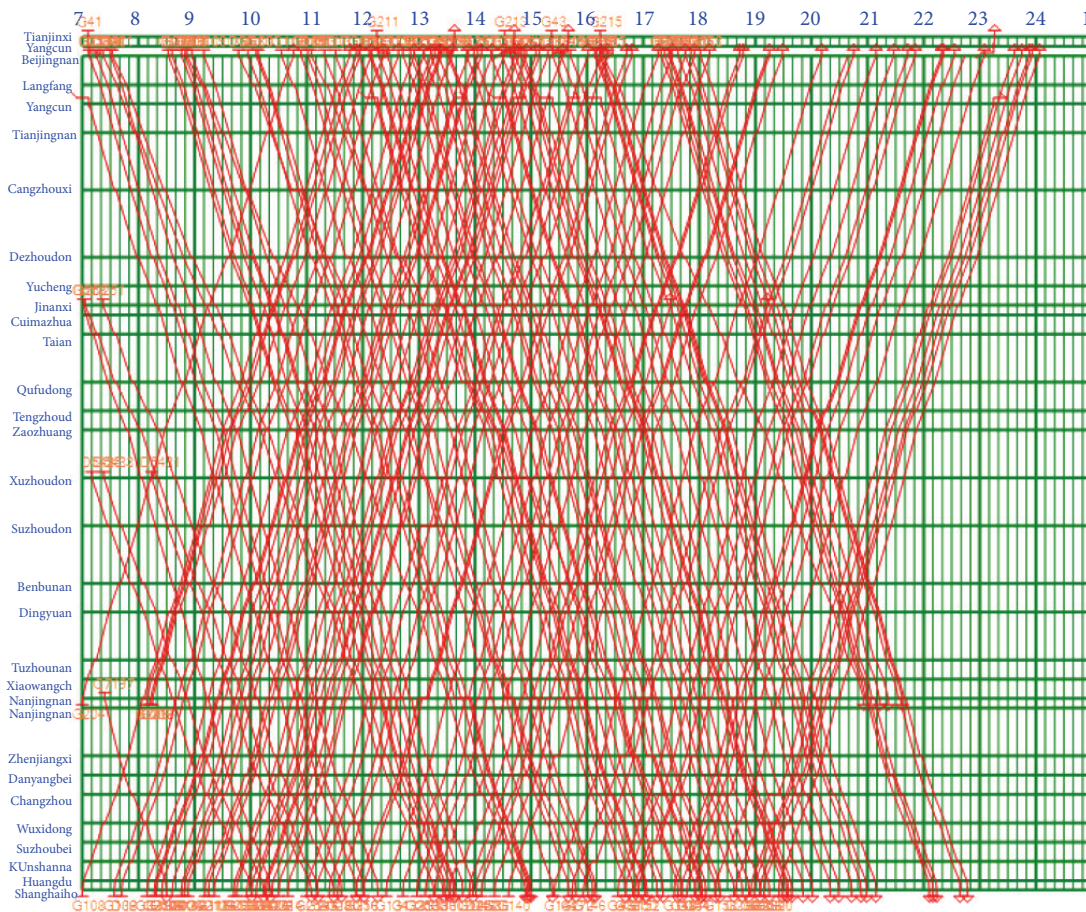
(b)

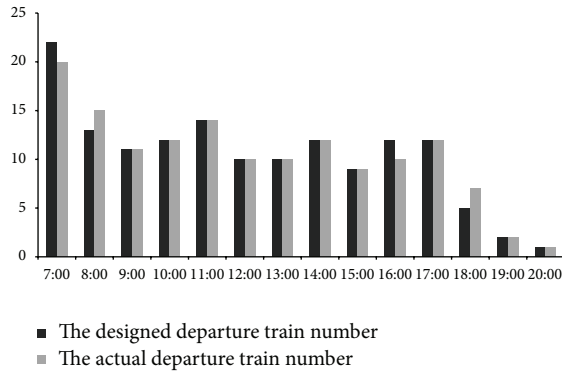FIGURE 8: The timetable for the instances.

FIGURE 9: The actual departure train number and the designed departure train number for each hour.

network can acquire a qualified timetable abides by the constraints in the model. In a 2 h period from 7 am to 9 am, a timetable schedule of 20 trains is shown in Figure 8(a) by the IBAP.

In Figure 8(a), all the trains keep away from the maintenance windows. Trains wait at station D until the maintenance task of section C-D ends, while trains select a later departure time to avoid too much trains waiting at station E to ensure that the total train amount is less than the amount in station D of parallel tracks. Meanwhile, the headway times at stations and sections separate the trains to safety movement. Moreover, trains can overtake each other at stations. The higher priority train with 300 km/h overtakes the train with 200 km/h at station D.

The effectiveness of general time-space path cost is testified by a 17 h period train timetable, which schedules 143 trains on Beijing-Shanghai HSR shown in Figure 8(b). As the remaining hours in a day are undergoing the maintenance task, the 17 h period train timetable is a whole day timetable.

Firstly, the actual departure train number and the design departure train number for each hour in Figure 9 are used to verify the service frequency requirement. As the fact that there are no trains departing after 20:00, only the numbers between 7:00 and 20:00 are counted in Figure 9. Figure 9 shows that in the peak hour, some trains have to depart in the next hour. However, in nonpeak hour, all the trains can depart in the design hour. Hence, the general cost makes the trains depart as soon as possible, and mostly the trains can depart in the designed departure hour.

Then, the total train travel time is 49977 min solved by IBAP, while the actual timetable of Beijing-Shanghai HSR of 2012 is 51133 min. Obviously, service frequency requirement and stopping plan adjustment make the railway system more efficient.

Finally, according to the timetable in Figure 8(b), the number of overtaking is nine in total. The type of 300 km/h train overtaking the type of 200 km/h train is eight, and the type of 300 km/h train overtakes a 300 km/h train only

once. Figure 10 gives three types of the priority of train in the timetable shown in Figure 8(b). There are two overtaking between the type of 300 km/h train and the type of 200 km/h train in Figure 10(a). Train G2 does not overtake train G252 at station Nanjingnan in Figure 10(b) to avoid the overtaking between the same speed trains. In Figure 10(c), train G158 overtakes G156 at station Jinanxi even they have the same speed, due to G158 depart in 18:00 later than the designed departure hour 17:00. Therefore, all the overtaking obeys the rules setting in Section 2.

4.2. The Effectiveness of the Improved Branch-and-Price Algorithm. There are 9 instances with different time horizon, train numbers, and networks in three HSR networks in China, which are Zhengzhou-Xian HSR, Beijing-Shanghai HSR, and Beijing-Guangzhou HSR. The column "Network" is expressed as "(section number, station number)". The periods include 5, 10, and 17 hours, and number of trains is 40, 80, and 140, respectively. The "CPU times" and "solution quality" list the performance of each algorithm, and the values are used to compare the effectiveness of the algorithms.

Three algorithms including the rolling horizon algorithm (RHA) in [24], the general branch-and-price (GBAP) and IBM Cplex 12.4 (a commercial software for linear programming) are selected to compare with IBAP. For each instance, the CPU time in seconds and solutions are provided in Table 2.

It is obvious that IBAP outperforms three algorithm described before. Due to that TTPTP is decomposed into a more small size problem literately, both the IBAP and GBAP are able to solve instance 1–9. IBAP, which adds heuristic time-space path generation, has a more powerful solving ability compared with GBAP. Table 2 verifies the effectiveness of our improvement measures that CPU time for IBAP is less than GBAP. Note that 28 in 147 trains' time-space path are acquired by the heuristic time-space path generation. The rest are acquired by the price problem. Therefore, the heuristic time-space path generation is beneficial to decrease the solving times and to increase the solving ability. As commercial software, Cplex 12.4, cannot handle the large-size instances shown in Table 2 because of CPU memory. For RHA, it has the ability to solve the instances with different size, but the solution is the worst, as well as it takes the longest CPU time. For example, for instance 9 it takes 3225 seconds, which is almost 3 times compared to the IBAP.

5. Conclusions

In this paper, a train timetabling optimization method based on time-space network for HSR is proposed. The time-space network is constructed by the way that the time is discretized into small time interval to describe trains' movement in stations and sections. A general time-space path cost is applied to obtain the HSR operational requirement, and can acquire a higher quality timetable. With a set of computational
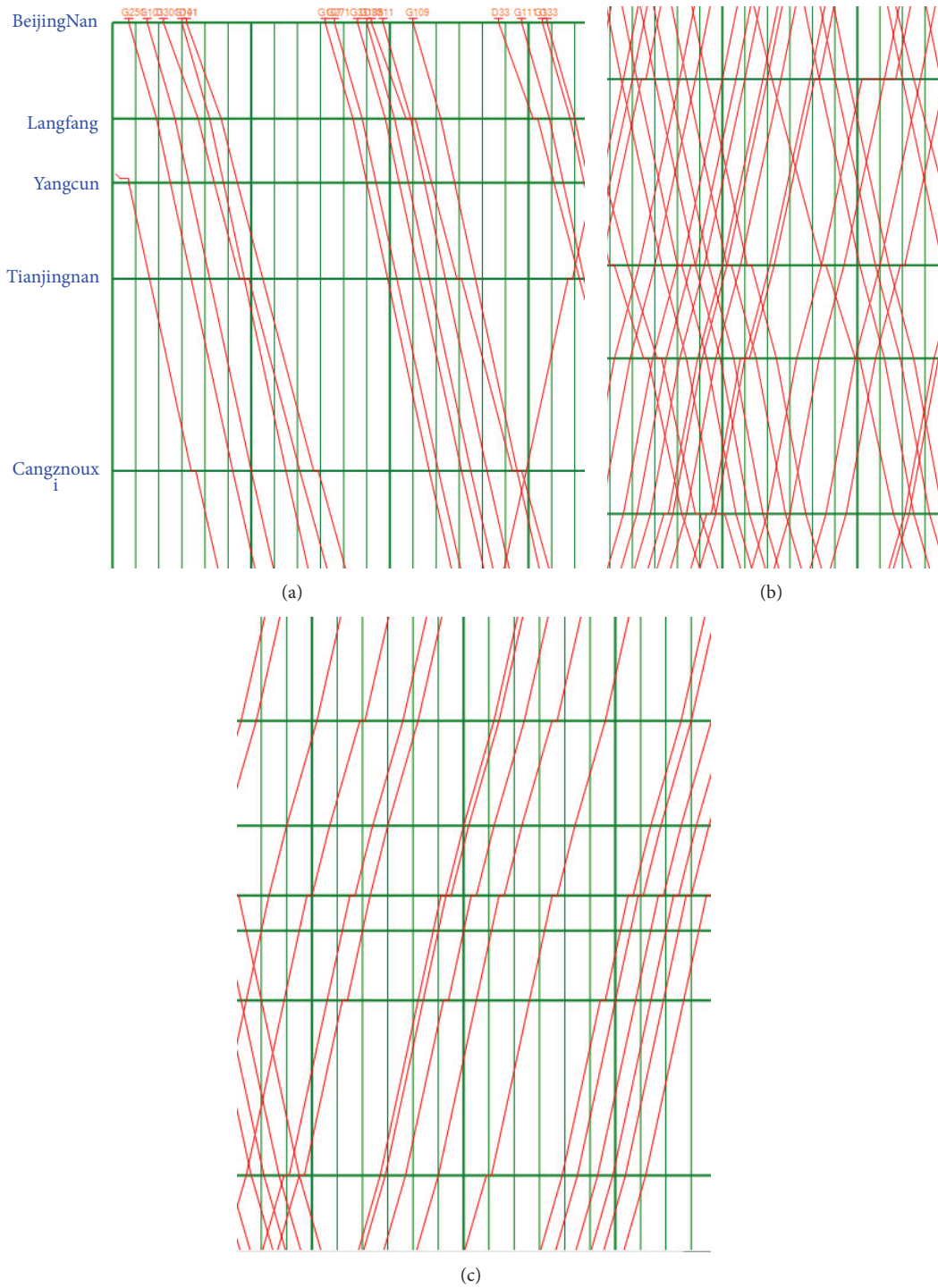
(a)

(b)

(c)

FIGURE 10: The examples for the overtaking.

experiments, the method is verified to be effective to acquire qualified timetables, which abided the HSR requirements. The instances with different sizes show that the designed algorithm outperforms the candidate algorithms.

Our on-going research focuses on two major aspects. First, the movement of trains inside the station is more complex in the real practice and should be implemented into a more precise simulation way. Then, the robustness of the timetable should also be considered in the future.

Conflict of Interests

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

TABLE 2: Solving Statistics for 9 instances.

#	Network	Period	Train numbers	CPU time (s)				Solution			
				Cplex	RHA	GBAP	IBAP	Cplex	RHA	GBAP	IBAP
1	(10, 9)	5	40	331	175	302	241	62300	5923	5802	5451
2	(10, 9)	10	80	657	235	398	354	122501	11323	10689	9878
3	(10, 9)	17	140	—	422	525	467	—	24432	23490	23412
4	(23, 22)	5	40	—	446	464	347	—	14981	13821	13323
5	(23, 22)	10	80	—	876	678	543	—	25234	24723	24591
6	(23, 22)	17	140	—	1113	796	686	—	51234	50321	49943
7	(40, 39)	5	40	—	1042	987	826	—	25189	24890	23414
8	(40, 39)	10	80	—	2283	1531	953	—	46903	46667	45594
9	(40, 39)	17	140	—	3225	1825	1221	—	86443	85564	82432

Acknowledgments

References

[1] B. Richard, S. Andrew, and J. Ying, "2012. High-speed rail: the first three years taking the pulse of China's emerging program," China Transport Topics 4, World Bank, Washington, DC, USA, 2012.

[2] H. Fu, L. Nie, B. R. Sperry, and Z. He, "Train stop scheduling in a high-speed rail network by utilizing a two-stage approach," *Mathematical Problems in Engineering*, vol. 2012, Article ID 579130, 11 pages, 2012.

[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] R. M. P. Goverde, *Punctuality of Railway Operations and Timetable Stability Analysis*, TRAIL Research School, Delft, The Netherlands, 2005.

[5] L. Peeters, *Cyclic Railway Timetable Optimization*, Erasmus Universiteit Rotterdam, Rotterdam, The Netherlands, 2003.

[6] C. Liebchen and L. Peeters, "Integral cycle bases for cyclic timetabling," *Discrete Optimization*, vol. 6, no. 1, pp. 98–109, 2009.

[7] L. Kroon, G. Maróti, M. R. Helmrich, M. Vromans, and R. Dekker, "Stochastic improvement of cyclic railway timetables," *Transportation Research B: Methodological*, vol. 42, no. 6, pp. 553–570, 2008.

[8] T. Schlechte, *Railway Track Allocation: Models and Algorithms*, Technische University Berlin, Berlin, Germany, 2012.

[9] X. Cai and C. J. Goh, "A fast heuristic for the train scheduling problem," *Computers and Operations Research*, vol. 21, no. 5, pp. 499–510, 1994.

[10] X. Cai, C. J. Goh, and A. I. Mees, "Greedy heuristics for rapid scheduling of trains on a single track," *IIE Transactions*, vol. 30, no. 5, pp. 481–493, 1998.

[11] A. Higgins, E. Kozan, and L. Ferreira, "Heuristic techniques for single line train scheduling," *Journal of Heuristics*, vol. 3, no. 1, pp. 43–62, 1997.

[12] A. Higgins, E. Kozan, and L. Ferreira, "Optimal scheduling of trains on a single line track," *Transportation Research B: Methodological*, vol. 30, no. 2, pp. 147–158, 1996.

[13] Y. Lee and C.-Y. Chen, "A heuristic for the train pathing and timetabling problem," *Transportation Research B: Methodological*, vol. 43, no. 8-9, pp. 837–851, 2009.

[14] R. L. Burdett and E. Kozan, "A sequencing approach for creating new train timetables," *OR Spectrum*, vol. 32, no. 1, pp. 163–193, 2010.

[15] A. D'Ariano, D. Pacciarelli, and M. Pranzo, "A branch and bound algorithm for scheduling trains in a railway network," *European Journal of Operational Research*, vol. 183, no. 2, pp. 643–657, 2007.

[16] 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.

[17] A. Caprara, M. Fischetti, and P. Toth, "Modeling and solving the train timetabling problem," *Operations Research*, vol. 50, no. 5, pp. 851–916, 2002.

[18] A. Caprara, M. Monaci, P. Toth, and P. L. Guida, "A Lagrangian heuristic algorithm for a real-world train timetabling problem," *Discrete Applied Mathematics*, vol. 154, no. 5, pp. 738–753, 2006.

[19] V. Cacchiani, A. Caprara, and P. Toth, "A column generation approach to train timetabling on a corridor," *Journal of Operations Research*, vol. 6, no. 2, pp. 125–142, 2008.

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

[21] F. Li, Z. Gao, K. Li, and D. Z. Wang, "Train routing model and algorithm combined with train scheduling," *Journal of Transportation Engineering*, vol. 139, no. 1, pp. 81–91, 2012.

[22] S. Harrod, "Modeling network transition constraints with hypergraphs," *Transportation Science*, vol. 45, no. 1, pp. 81–97, 2011.

[23] R. Lusby, J. Larsen, D. Ryan, and M. Ehrgott, "Routing trains through railway junctions: a new set-packing approach," *Transportation Science*, vol. 45, no. 2, pp. 228–245, 2011.

[24] C. Barnhart, E. L. Johnson, G. L. Nemhauser, M. W. P. Savelsbergh, and P. H. Vance, "Branch-and-price: column generation for solving huge integer programs," *Operations Research*, vol. 46, no. 3, pp. 316–329, 1998.