Hindawi
Journal of Advanced Transportation
Volume 2021, Article ID 6679008, 16 pages
https://doi.org/10.1155/2021/6679008

Research Article

Simultaneous Optimization of Train Timetabling and Platforming Problems for High-Speed Multiline Railway Network

Qin Zhang, Xiaoning Zhu, Li Wang, and Shuai Wang

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

Correspondence should be addressed to Li Wang; liwang@bjtu.edu.cn

Received 8 December 2020; Revised 7 February 2021; Accepted 2 March 2021; Published 17 March 2021

Academic Editor: Andrea D'Ariano

The optimization problems of train timetabling and platforming are two crucial problems in high-speed railway operation; these problems are typically considered sequentially and independently. With the construction of high-speed railways, an increasing number of interactions between trains on multiple lines have led to resource assignment difficulties at hub stations. To coordinate station resources for multiline train timetables, this study fully considered the resources of track segments, station throat areas, and platforms to design a three-part space-time (TPST) framework from a mesoscopic perspective to generate a train timetable and station track assignment simultaneously. A 0-1 integer programming model is proposed, whose objective is to minimize the total weighted train running costs. The construction of a set of incompatible vertexes and links facilitates the expression of difficult constraints. Finally, example results verify the validity and practicability of our proposed method, which can generate conflict-free train timetables with a station track allocation plan for multiple railway lines at the same time.

1. Introduction

High-speed railways play a significant role in modern transportation systems. In China, the length of a high-speed railway has reached 35,000 km after 12 years of operation, which makes China the country with the longest running mileage of high-speed railway.

The train timetabling problem, also called the train scheduling problem, is a fundamental problem in railway operation; the aim is to determine the station arrival and departure times for every train. A train timetable provides a reference for different departments in a train operation system to ensure successful service implementation. Train timetable quality influences not only the utilization of railway capacity but also the work of train dispatchers. A good train schedule can rapidly reduce modification during actual operation, making the service provided by a railway company reliable and competitive.

The train platforming problem is a procedure following the design of a train timetable; the aim for this problem is to assign a specific conflict-free platform in every station to trains that are scheduled in the train timetable. Stations are a complicated component of a railway network. The track resources in a station can be divided into platform tracks and tracks in the throat area; note that the throat area is also sometimes referred to as the bottleneck area. The path connecting one platform and the station boundary may occupy some track resources in the throat area the same as a path connecting another platform and the station boundary. Thus, a reasonable track allocation plan in a station must be conflict-free, not only in the platform area but also in the throat area.

In China, train timetables are generated sequentially as follows: First, a timetable is generated for each direction on each single line. Then, resource utilization in the hub stations that connect different lines is coordinated. If there is no feasible platform plan in one of the stations, the procedure is repeated. With the construction of new railway lines, networks are becoming increasingly complicated, and the interaction between all trains from different directions on different railway lines makes hub station capacity a limiting factor for the whole network. The sequential scheduling method cannot solve the train timetabling and platforming problems effectively and efficiently. As a result, there have recently been urgent calls to construct a framework that is suitable for the simultaneous optimization of the two problems.

As shown in Figure 1, a high-speed railway network is divided into two parts: stations and track segments (track between two stations). The resources in a station include tracks in the bottleneck area and several platform tracks. A solution framework that integrates timetabling and platforming must guarantee that the utilization of all resources, including track segments, platforms, and tracks at the throat area, is conflict-free.

Our paper addresses the integrated optimization of the train timetabling and train platforming problems at the planning level from a mesoscopic perspective. The goal of this study was to design a modeling framework for this integrated problem. Under the framework, trains from different directions can be simultaneously optimized using a flexible track utilization rule, where one train can use all platforms if the station layout allows for it. The current issues in developing such a framework are as follows:

(i) For the train timetabling problem at the planning level, railway networks are mostly modeled from a macroscopic perspective. As noted in Zhang et al. [1]; only a few studies have solved this problem microscopically. Accuracy cannot be guaranteed by the former, while the latter perspective will dramatically increase the number of variables and constraints.

(ii) Specific platform allocation is ignored in the train timetabling problem. As mentioned earlier, timetabling and platforming are treated as separate and sequential procedures. A feasible solution may be difficult and sometimes impossible to find by the sequential scheduling method, especially for a network connecting several railway lines through some complex high-speed railway hub stations.

(iii) For the train platforming issue in the train scheduling problem, in many works, trains have been considered independently if their running directions are different, such as Zhang et al. [2] and Zhou et al. [3]. This method is feasible for most single railway lines. However, in a hub station, trains cannot be separated by their directions. Meanwhile, the track resources in the throat area are sometimes ignored, leading to infeasible or unreliable train platform plans.

This study presents a solution framework to address the three abovementioned issues and simultaneously optimize train timetabling and platforming. The contributions of this work can be summarized as follows:

(i) First, we model the railway network with multiple railway lines from a mesoscopic perspective. The track segments between two stations are considered as a whole at the macroscopic level. In a station, the resources in the station bottleneck area are modeled in terms of routes, and station platform resources are also considered.

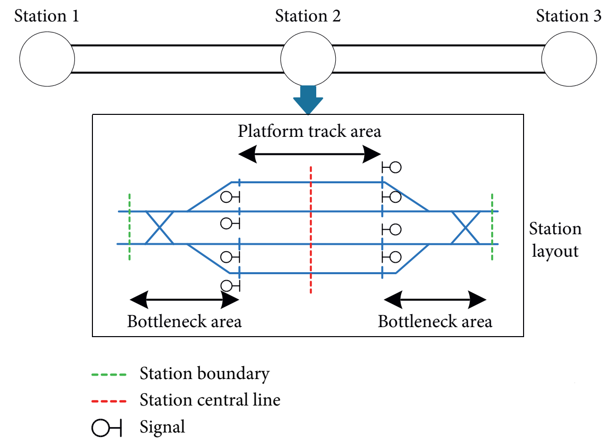

Figure 1: Railway network representation.

(ii) Second, we design a three-part (segment-bottleneck-platform) framework to allow the integration of timetabling and platforming. The three-part framework extends the previous two-part one (segment-station) and divides the station resources into a bottleneck area and platform.

(iii) Third, we schedule all trains at the same time. We generate all possible paths within a station in our proposed space-time network and preprocess the incompatible link set for all links according to the microscopic physical station layout.

The remainder of this paper is organized as follows: Section 2 provides a literature review on the train timetabling and platforming problems. Section 3 describes the construction of the space-time network under the three-part solution framework. Section 4 proposes a 0-1 integer programming model to simultaneously optimize the two problems. Section 5 presents computational results by the commercial CPLEX solver. Conclusions and future research directions are presented in Section 6.

2. Literature Review

2.1. Train Timetabling Problem. Train timetabling is a classical problem in railway operation. Caprara et al. [4] proved that the train scheduling problem is NP-hard, and they timetabled trains in a single-line one-way railway network. The objective was to maximize train profit, and a Lagrangian relaxation method was proposed to obtain their results. Carey and Crawford [5] designed a heuristic algorithm for a railway network with multiple complex stations linked by multiple one-way lines in each direction by extending the one-way single-station method in Carey and Carville [6]. A multiobjective model was constructed, and the objectives were sequenced by a lexicographic rule. Tian and Niu [7] proposed a biobjective integer programming model to maximize train connections while minimizing the passenger transfer waiting time. In the present study, overtaking was ignored in all stations, and trains from different directions were scheduled independently.

D'ariano et al. [8] modeled a railway network in terms of track sections from a microscopic perspective and proposed an alternative graph representation, which was solved by a branch-and-bound method. Sama et al. [9] formulated an integer programming and proposed an ant-colony based method to address the train route section problem in a microscopic network. Zhou and Zhong [10] analyzed the train timetabling problem for a single railway line and designed a resource-constrained scheduling model to minimize the total travel time. Their results were obtained by branch-and-bound and Lagrangian relaxation methods. Sotskov and Gholami [11] used a shifting bottleneck algorithm to design a single-line train schedule. However, every resource of the railway could be occupied only once by every train, which is not reasonable for a railway network connecting several lines. Brannlund et al. [12] constructed an integer programming model and developed a Lagrangian relaxation method to maximize the train profit on a single-line railway network. The static station capacity, that is, the number of platforms, was considered in this model. Nillson and Chen [13] simultaneously optimized the train timetabling and train platforming problems, but the resource utilization in the bottleneck area in a station was ignored.

For a highly congested railway line, Jiang et al. [14] proposed a heuristic method to simultaneously optimize the additional train schedule and the choice of the train stops. To solve the cyclic train timetable problem, Zhang et al. [15] extended the time-space network and then introduced the alternating direction method of multipliers. Luan et al. [16] proposed a mixed-integer linear programming model based on cumulative flow variables to optimize train scheduling as well as the tracking maintenance tasks from a microscopic perspective and used the Lagrangian relaxation method to obtain the revised timetable. Zhang et al. [17] optimized the train timetabling problem considering the detailed track assignment on a double-track railway corridor from a mesoscopic level and the running time on the segment as a constant value. The track utilization within a hub station, where more kinds of conflicts, such as those originating due to arrival and departure of two trains from two different railway lines, occur than those in a normal intermediate station, was not considered in Zhang et al. [17].

Trains from different directions are always considered separately, which is not possible for some complex stations. Gao et al. [18] addressed the scheduling problem for adding the additional trains on a high-speed rail corridor by designing a three-stage optimization method and Zhang et al. [19] integrated the maintenance planning and train night timetable on the high-speed railway. However, both papers independently treated the inbound and outbound trains.

2.2. Train Platforming Problem. The train platforming problem, which is also known as the track allocation problem within a station, has attracted much attention. There are four common methods currently used to solve this problem.

Cardillo et al. [20] first applied the graph-coloring approach to the train platforming problem, where the same color cannot be assigned to two trains with conflicting routes. Similarly, Zwaneveld et al. [21] and Zwaneveld et al. [22] proposed a node-packing method to allocate two trains without conflicting routes to the same platform. Carey and Carville [6] introduced a manual rescheduling technique to simulate the process of track allocation modification. This simulation allowed the authors to take advantage of the practical experience of dispatchers, which is easy to understand.

The most common way to obtain a train platform plan is to construct a single-objective or multiobjective model, such as to minimize the total train running time or maximize passenger convenience in a station. Billionnet [23] formulated a 0-1 integer programming model to represent the track allocation problem in Cardillo et al. [20]; and a heuristic algorithm was introduced to generate the results. Wu et al. [24] proposed a mean-variance optimization model based on Markowitz's portfolio theory to minimizes resource occupation cost, and they used the simulated annealing algorithm to solve this programming. Pellegrini et al. [25] constructed mixed-integer programming to address the route management in complex junctions under traffic perturbations. A track allocation problem for multidirection high-speed railway stations was proposed in Zhang et al. [26] and Zhang et al. [27]. They considered a typical Chinese high-speed railway station layout, and they noted that the dependency between trains from different directions made resource utilization in the bottleneck area of one station difficult but important.

3. Problem Description

A typical space-time network representation of a train timetable, as shown in Figure 2, is a common solution framework currently used by researchers. The space-time path describes train locations in the time dimension. Under the segment-station framework (framework 1), each station is treated as a node, which trains can dwell at or directly pass through. Framework 2 in Figure 2 is a segment-arrival-departure space-time network, where every station node in framework 1 is replaced with arrival and departure events at the station. These two common railway network modeling methods separate trains from different directions and solve them independently. As described in Zhang et al. [17], frameworks 1 and 2 cannot provide a good way to integrate specific track allocation in a station and the utilization of track resources in the throat area into a train timetabling problem.

In China, each platform in a station can typically only be occupied by one train at a time. To address the integration problem of timetabling and platforming, we, therefore, stretch each platform in every station in framework 1, and the train stop point on each platform track is denoted by a physical node. In addition, the arrival and departure boundaries are illustrated as physical nodes to allow the occupation of bottleneck resources to be considered. In practice, there is one basic route that connects one station boundary and one station platform; thus, we assume there exists at most one path between each station boundary and platform. We can extend every physical node along the time

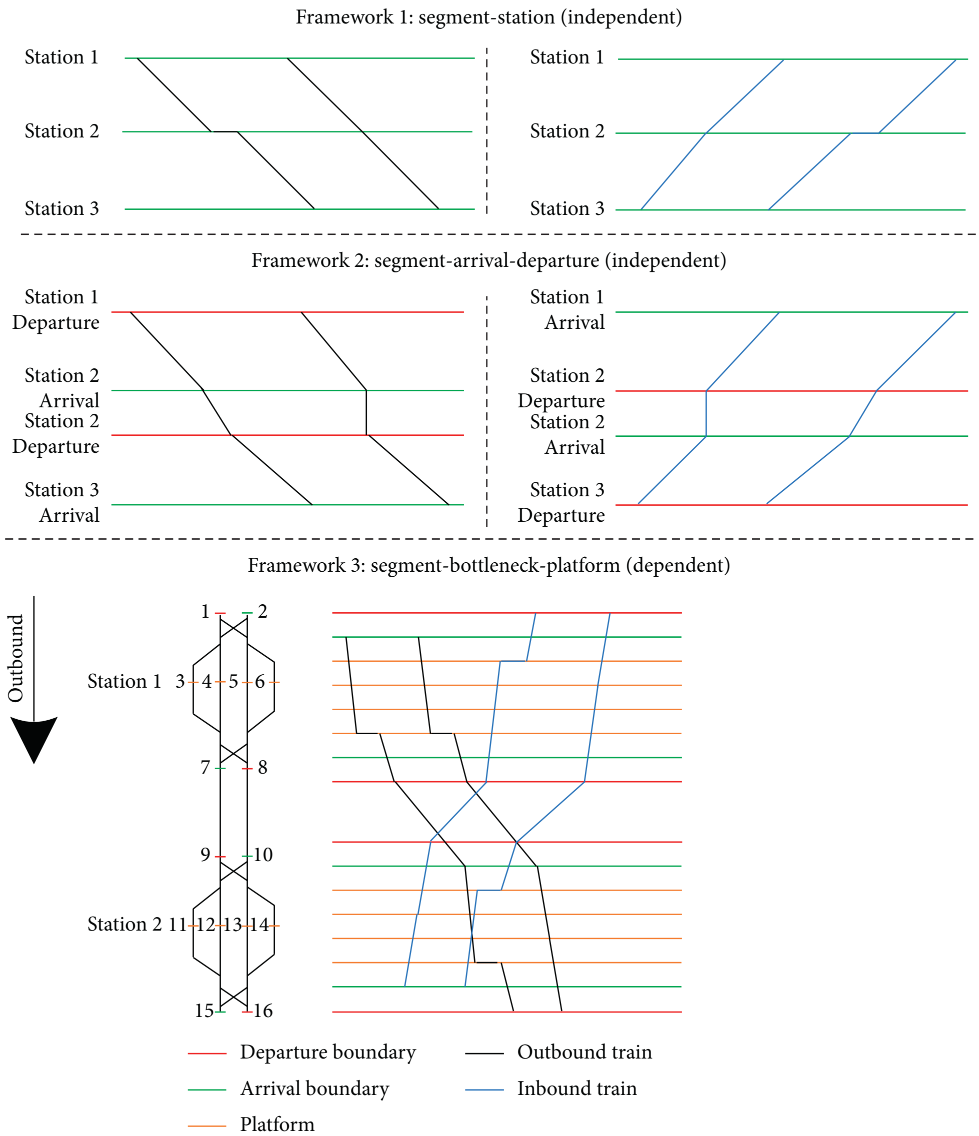

Figure 2: Three railway network frameworks.

dimension and construct a segment-bottleneck-platform space-time network, as shown in Figure 2, which provides the possibility of representing trains running in the station. The utilization of station resources by trains from different directions can be further constrained to be conflict-free. In conclusion, this TPST framework can simultaneously optimize train timetables and station resource utilization for all trains.

The physical railway network in the proposed framework consists of three types of links and four types of node sets. The physical node set N includes the station arrival boundary node set N^a, station departure boundary node set N^d, station platform node set N^{st}, and segment node set N^{se}. Node in the segment node set represents the location where convergence or divergence happens, which is different from the node representation from the microscopic perspective. Meanwhile, the platform node set includes the nonstop platform node set N^{st}_{ns} and stop platform node set N^{st}_{st}. A stop platform track is sometimes referred to as a siding track, and a nonstop platform track represents the mainline or main track in the station. The physical link set E includes the arrival link set E^a, departure link set E^d, and segment link set E^{se}. (i, j) represents a link from nodes i to j. For simplicity of expression, we define $e = (i, j)$ and $o_e = i, d_e = j$.

In the TPST network, we extend the nodes and links in the physical network along the time dimension. t, τ denote the time index. The vertex set in this space-time network can be defined as $V = V^a \cup V^d \cup V^{st} \cup V^{se}$, where V^a, V^d, V^{st}, and V^{se} correspond to the time extensions of N^a, N^d, N^{st}, and N^{se}, respectively. Similarly, the space-time platform vertex set can be categorized into a nonstop platform vertex set V^{st}_{ns} and stop platform vertex set V^{st}_{st}. $v = (i, t)$ is the vertex index, and $n_v = i, t_v = t$.

As shown in Figure 3, we use A to denote the arc set of the TPST network, and this set consists of several arc subsets, which are mainly extensions from the physical link set.

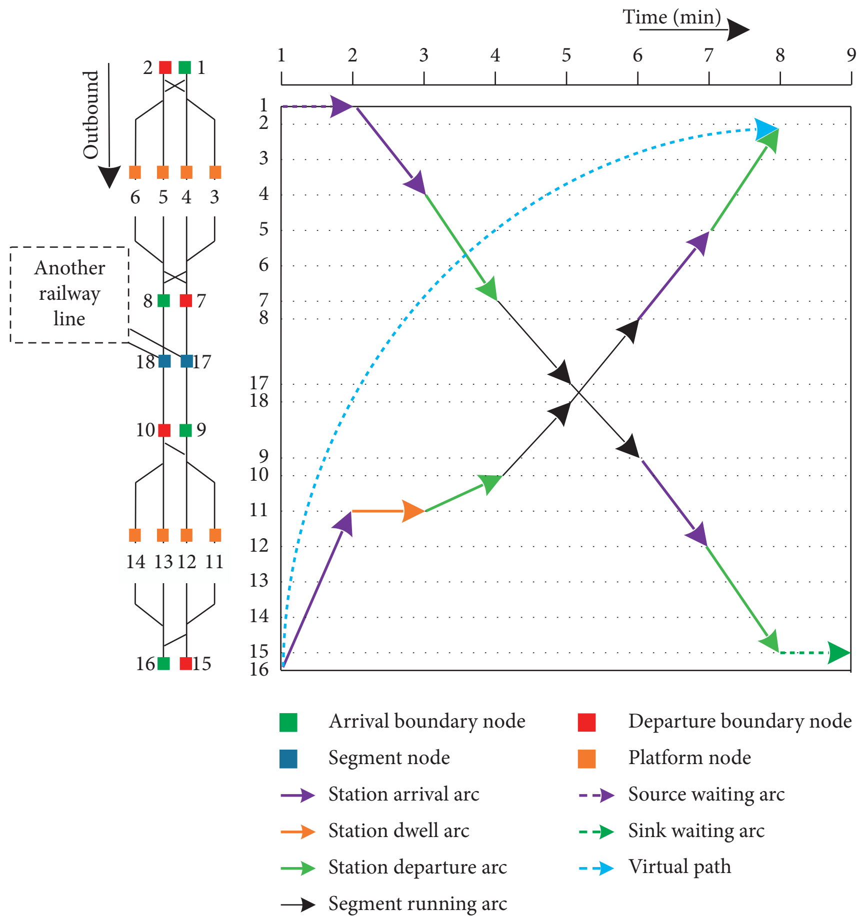

FIGURE 3: Space-time arc in the TPST network.

(i, j, t, τ) represents the arc from vertex (i, t) to vertex (j, τ). In the same manner, we use g to represent the arc (i, j, t, τ), and $o_g = i, d_g = j, \widehat{o_g} = t$, and $\widehat{d_g} = \tau$ for simplicity. A^a denotes the set of station arrival arcs from the arrival vertex to the platform vertex, and A^d denotes the set of station departure arcs from the platform vertex to the departure vertex. To model the dwelling of a train within a station, we introduce the station dwell arc set A^{st} to represent the arcs connecting (i, t) and $(i, t + 1)$, where $i \in N_{\mathrm{st}}^{\mathrm{st}}$, which represents a stop train waiting on a siding track. Train movement on a track segment is illustrated by the segment running arc set A^{se}. The source waiting arc set A^o and sink waiting arc set A^e represent the waiting time at the source node from the earliest available train time and that at the sink node until the latest available train time, respectively. F is the train arc set. In this study, we allowed the cancelation of trains. Therefore, the virtual path set A^{vi} is constructed, and for each train $f \in F$, a corresponding virtual path is chosen from source vertex δ_f to the sink vertex γ_f if it is canceled.

For each train $f \in F$, the available node set, link set, vertex set, and arc set can be represented as $N_f = N_f^a \cup N_f^d \cup N_f^{\mathrm{st}} \cup N_f^{\mathrm{se}}$, $E_f = E_f^a \cup E_f^d \cup E_f^{\mathrm{se}}$, $V_f = V_f^a \cup V_f^d \cup V_f^{\mathrm{st}} \cup V_f^{\mathrm{se}}$, and $A_f = A_f^a \cup A_f^d \cup A_f^{\mathrm{st}} \cup A_f^{\mathrm{se}} \cup A_f^o \cup A_f^e \cup A_f^{\mathrm{vi}}$, respectively. The attributes of all types of space-time arcs for one train are listed in Table 1. For each train, n_f^o and n_f^d are defined as its source and sink nodes, respectively. The last column in Table 1 presents the cost of each kind of arc. The cost of the segment running arc and the station dwell arc are equal to their respective running times. For the station arrival and departure arcs, α_{ij} represents the train preference for station route (i, j). For the preferred arrival and departure route (i, j), the weight α_{ij} can be small. If route (i, j) is not preferred, we can set the value of α_{ij} to be large.

TABLE 1: Attributes of train available arcs in the space-time network.

Train arc	From vertex	To vertex	Description	Arc cost
A_f^o	(i, t)	$(i, t + 1)$	$i = n_f^o$	1
A_f^a	(i, t)	(j, τ)	$(i, j) \in E_f^a$	$\alpha_{ij} * (\tau - t)$
A_f^d	(i, t)	(j, τ)	$(i, j) \in E_f^d$	$\alpha_{ij} * (\tau - t)$
A_f^{st}	(i, t)	$(i, t + 1)$	$i \in N_f^{\mathrm{st}}$	1
A_f^{se}	(i, t)	(j, τ)	$(i, j) \in E_f^{\mathrm{se}}$	$\tau - t$
A_f^e	(i, t)	$(i, t + 1)$	$i = n_f^d$	1
A_f^{vi}	σ_f	γ_f	—	T (time horizon)

4. Problem Formulation and Methods

4.1. Assumptions. Without loss of generality, we make the following assumptions to facilitate the construction of our proposed model:

(1) Railway infrastructure information, train information (including train route, departure time window at train source node, and minimum and maximum dwell time at all stations for every train), and all safety time criteria are given

(2) The interlocking system in a station is assumed to be route-lock route-release

(3) The running time of arrival and departure routes in one station is assumed to be constant for every train

(4) A train is considered as a node; train length is ignored

(5) The granularity of time is one minute

4.2. Train Timetabling and Platforming Model (TTAPM)

4.2.1. Notations and Decision Variables. The notations and binary decision variables used in the model are listed in Tables 2 and 3.

Table 2: Model notations.

Notation	Definition
k	Station index $\in\in K$
K_f	Stations for train f
T_k^r	Running time of arriving or departing route at station $k \in K$
T^r	Headway for utilization of track resources in the throat area
n_k^l	Number of railway lines that merge or disjoin at station k
c_g^f	Cost of arc g for train f
$\delta_f^+(v), \delta_f^-(v)$	Arcs of train f that flow out of/into vertex v
σ_f, γ_f	Origin and destination vertex for train f
D_k^f	Minimum dwell time of train $f \in F$ at station $k \in K$
D_k^{fm}	Maximum dwell time of train $f \in F$ at station $k \in K$
A_k^s	Station dwell arcs at station k
T^s	Usage headway of a stop platform track

Table 3: Model decision variables.

Variable	Definition
x_g^f	Binary decision variable that indicates whether train $f \in F$ uses arc g (=1) or not (=0)
y_g^f	Binary decision variable that indicates whether train $f \in F$ implicitly uses waiting arc $g \in A^{\text{st}}$ (=1) or not (=0)

4.2.2. Objective Function. Objective functions for the train timetabling problem typically minimize the total weighted running time of trains [2, 28] or maximize the satisfaction of passengers [29]. For the train platforming problem, the objective function typically minimizes the occupation time of station resources [24, 26]. In our model, the objective function aims to minimize the total weighted train running costs:

$$\min Z_1 = \sum_{f\in F} \sum_{g\in A_f} c_g^f \cdot x_g^f. \tag{1}$$

4.2.3. Constraints. (1) Train Flow Balance Constraints.

$$\begin{aligned} &\sum_{g\in\delta_f^+(\sigma_f)} x_g^f = 1, \quad \forall f \in F, \\ &\sum_{g\in\delta_f^-(v)} x_g^f - \sum_{g\in\delta_f^+(v)} x_g^f = 0, \quad \forall v \in V_f \backslash \{\sigma_f, \gamma_f\}, \forall f \in F. \end{aligned} \tag{2}$$

(2) Minimum and Maximum Train Dwell Time Constraints.

$$\left(1 - \sum_{g\in A_f^{\text{vi}}} x_g^f\right) \cdot D_k^f \le \sum_{g\in A_k^s \cap A_f} x_g^f \le \left(1 - \sum_{g\in A_f^{\text{vi}}} x_g^f\right) \cdot D_k^{fm}, \quad \forall f \in F, \forall k \in K_f. \tag{3}$$

(3) Train Headway Constraint on a Track Segment. The headway constraints on a track segment include arrival headway constraints, departure headway constraints, and overtaking constraints. The overtaking constraints can be processed as the arrival and departure headway constraints, and we refer the reader to Kroon and Peeters [30] for more details on this.

To build a model considering the headway constraints on a track segment, Zhang et al. [17] assumed the running time of all trains on a single track segment to be the same and constructed the incompatible arc set for each segment running arc. Cacchiani et al. [31] allowed the running time on the segment arcs to vary and constructed the incompatible vertex set for each station arrival or departure node. In this paper, we present the construction of an incompatible vertex set for each departure vertex, segment vertex, and arrival vertex. As depicted in Figure 4, for each vertex $v \in V^d \cup V^{\text{se}}$, we define the incompatible vertex set for the departure headway constraints as $\psi_v^1 = \{v' | t_{v'} \in [t_v, t_v + T^d), n_{v'} = n_v\}$; for each vertex $v \in V^a \cup V^{\text{se}}$, we define the incompatible vertex set for arrival headway constraints as $\psi_v^2 = \{v' | t_{v'} \in [t_v, t_v + T^a), n_{v'} = n_v\}$. Notably, T^d and T^a are the departure and arrival headway times on the track segment, respectively. We introduce the auxiliary decision variable x_v^f to denote that train f occupies vertex v (=1) or not (=0). The connection between arc occupation and vertex occupation can be expressed as the following equations:

$$x_v^f = \sum_{g\in\delta_f^-(v) - A_f^e - A_f^{\text{vi}}} x_g^f, \quad \forall v \in V_f^d \cup V_f^{\text{se}}, \forall f \in F, \tag{4}$$

$$x_v^f = \sum_{g\in\delta_f^+(v) - A_f^o - A_f^{\text{vi}}} x_g^f, \quad \forall v \in V_f^a, \forall f \in F. \tag{5}$$

Therefore, the arrival and departure headway constraints on a track segment can be represented as follows:

$$\begin{aligned} &\sum_{v'\in\psi_v^1} \sum_{f:\, v'\in V_f} x_{v'}^f \le 1, \quad \forall v \in V^d \cup V^{\text{se}}, \\ &\sum_{v'\in\psi_v^2} \sum_{f:\, v'\in V_f} x_{v'}^f \le 1, \quad \forall v \in V^{\text{se}} \cup V^a. \end{aligned} \tag{6}$$

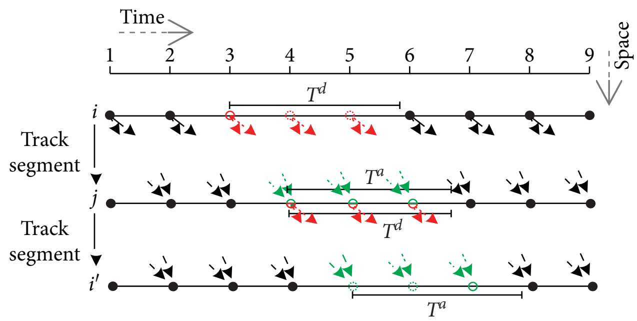

FIGURE 4: Illustration of the arrival and departure constraints on a track segment.

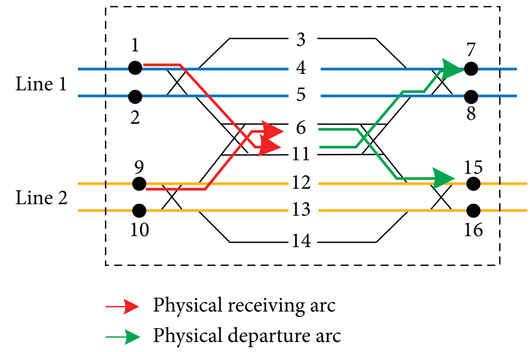

FIGURE 5: Illustration of receiving-receiving and departure-departure conflicts in a hub station.

(4) Headway Constraints at a Station Bottleneck Area. The headway constraints at a station throat area involve the receiving-receiving headway, departure-departure headway, and receiving-departure headway. The first two in a station on a single railway line can be guaranteed by the segment headway for all trains.

For a station connecting more than one railway line, the receiving-receiving headway for trains from the same arrival boundary node and departure-departure headway for trains to the same departure boundary node can also be guaranteed by the arrival and departure headways on a track segment, respectively. However, the receiving-receiving and departure-departure headways between two trains from two different railway lines in a hub station, as shown in Figure 5, should be enforced.

Different from Zhang et al. [17], in this paper, we consider the railway network rather than a railway corridor. According to the microscopic network view of a certain hub station k, we can enumerate the incompatible receiving-receiving physical link pair set ψ_k^{aa} and incompatible departure-departure physical link pair set ψ_k^{dd} for two arrival links or two departure links that connect two different railway lines. As shown in Figure 5, the physical link $(1, 11)$ conflicts with physical link $(9, 6)$ because these two routes occupy the same track resource in the bottleneck area. Similarly, the physical departure link $(11, 7)$ is incompatible with link $(6, 15)$. Accordingly, given the layout of this station, we have

$$\begin{aligned}\psi_1^{aa} &= \{\{(1,11),(9,6)\},\{(1,6),(9,6)\},\{(1,11),(9,11)\}\},\\ \psi_1^{dd} &= \{\{(11,7),(6,15)\},\{(6,7),(6,15)\},\{(11,7),(11,15)\}\}.\end{aligned} \tag{7}$$

When we have the topology for all multiline stations from the microscopic perspective, we can preprocess the incompatible physical link pair set, and then the arrival-arrival and departure-departure headway constraints for different railway lines at the throat area in the multiline stations can be expressed as follows:

$$\begin{aligned}&\sum_{g:\,\{(o_g,d_g)\in\xi,\widehat{o}_g\in(t,t+T_k^r+T^r)\}}\ \sum_{f:\,g\in A_f} x_g^f \le 1,\\ &\forall t,\forall \xi\in\psi_k^{aa}\cup\psi_k^{dd},\forall k\colon n_k^l>1.\end{aligned} \tag{8}$$

For all stations, because we allow trains to use all possible platform tracks so that we can schedule all trains simultaneously, we must construct headway constraints to avoid conflicts between an arrival physical link and departure physical link. In the same manner, from the topology for all stations, we can enumerate the incompatible receiving-departure physical link pair set for all trains ψ_k^{ad}, and then the arrival-departure headway can be constrained as follows:

$$\sum_{g:\,\{(o_g,d_g)\in\xi,\widehat{o}_g\in[t,t+T_k^r+T^r)\}}\ \sum_{f:\,g\in A_f} x_g^f \le 1,\quad \forall t,\forall\xi\in\psi_k^{ad},\forall k. \tag{9}$$

(5) Headway Constraints on a Stop Platform Track. When two trains occupy the same siding track, there should be headway time between the two occupations. To facilitate a model of this type of headway, we introduce the auxiliary decision variable y_g^f to represent train f having already left the siding track o_g but still occupying this siding track implicitly due to safety requirements. The map between the implicit and actual occupations of the station dwell arcs can be expressed as equation (10). Then, the headway constraints on the siding tracks can be transformed into a unique occupation of the siding track, whether implicit or actual, as shown by equation (11):

$$\sum_{g'\in A_f^d\mid\widehat{o}_{g'}\in(\widehat{o}_g-T^s,\widehat{o}_g]} x_{g'}^f = y_g^f,\quad \forall g\in A_f^{\mathrm{st}},\forall f\in F, \tag{10}$$

$$\sum_{f:\,g\in A_f}\left(x_g^f+y_g^f\right)\le 1,\quad \forall g\in A^{\mathrm{st}}. \tag{11}$$

(6) Domain of Variables. The domains of variables are given as follows:

$$\begin{aligned}x_g^f &\in\{0,1\},\quad \forall f\in F,\forall g\in A_f,\\ y_g^f &\in\{0,1\},\quad \forall f\in F,\forall g\in A_f^{\mathrm{st}},\\ x_v^f &\in\{0,1\},\quad \forall f\in F,\forall v\in V_f^d\cup V_f^{\mathrm{se}}\cup V_f^a.\end{aligned} \tag{12}$$

4.3. Solution Method. The TPST network can reduce not only the number of variables compared to modeling from a microscopic perspective but also the number of constraints. Some constraints that can be satisfied when constructing the network are listed as follows:

(1) No dwelling at the mainline of a station: no train can dwell at the main track in a station in our model because there is no dwell arc extended from a nonstop platform track.

(2) No change of a siding track can occur when a train dwells on it. A train cannot move to other siding tracks in the same station it is dwelling at, and there is no transfer arc from one siding track to another.

In addition, the construction of the incompatible link and arc sets facilitates the modeling of difficult headway constraints in the train timetabling and platforming problems.

Based on the above improvements in previous studies, we used the CPLEX solver to obtain our results. All instances presented in this paper were implemented in Python 3.7 and tested on a personal Windows computer with a 2.80 GHz processor and 16 GB RAM.

5. Experiment Results and Discussion

In this section, we report several computational experiments to verify the validity of our proposed TPST network and model based on a small-sized and a medium-sized railway networks with three and nine stations, respectively. We test the performance of our model with the change of some parameters. The benefits of the railway network modeling method from the mesoscopic perspective compared with the microscopic and macroscopic perspective are also given.

5.1. Small-Sized Rail Network and Performance Analysis. The small rail network in this section consists of three stations with 37 nodes and 66 physical links, as shown in Figure 6. The running time of each track segment between two stations is 4 min, and the running time of two track segments that connect two railway lines (i.e., physical links $(36, 34)$ and $(35, 37)$) is 2 min. The running time of an arrival or departure route at Station 1 is 2 min, while that at Stations 2 and 3 is 1 min. The time horizon T is set to 120 min in all cases. The cost for every virtual path is set to the same value with time horizon.

All basic train data are listed in Table 4. In the table, the second and third columns give information on the train source and sink nodes. The trains' station sequences are listed in the fourth column. We also limit the train start time window at its source node in the fifth column. In the last column, the minimum train dwell time at each station is shown.

5.1.1. Performance Comparison with an Increase in the Number of Trains. In this subsection, we report the CPU running time of our model by CPLEX with an increase in the number of trains. The length of the start time window for each train maintains a value of 5. The maximum dwell time for each train at each station is 4 min more than the minimum dwell time. We consider 9 scenarios with the number of trains increasing from 5 to 44, as shown in Table 5. Under each scenario, 10 instances were tested and the CPU running time is given as the mean value of all instances in one scenario.

As can be seen from the table, the amount of CPU time required to obtain a result by the CPLEX solver generally increases when the number of trains increases from 5 to 44, except in scenarios 1-6. However, all results can be obtained within a reasonable time.

We also illustrate the optimal timetable and platform plan in scenarios 1-9 for Line 1 and Line 2 in Figures 7 and 8, respectively. In total, two trains (Trains 3 and 6) are not scheduled into this railway network. All scheduled trains satisfy the safety requirements, which verifies the validity of our proposed model.

5.1.2. Performance Comparison with an Increase in the Number of Train Paths. In this subsection, we report the difference in solving performance when the number of train paths increases and the number of trains is fixed to be 30. For each train, the number of paths depends on the lengths of the start time and dwelling time windows in each station. We consider two environments. In the first environment, the length of the start time window is variable, and the length of the station dwell time window is fixed, and in the second environment, the contrary is true. We constructed 6 and 5 scenarios for each environment, as shown in Table 6.

As listed in Table 6, in scenarios 2-1 to 2-6, the start time window varies from 2 min to 25 min, while the station dwell time window is fixed at 4 min. Further, in scenarios 2-7 to 2-11, the station dwell time window varies from 2 min to 20 min, while the start time window is fixed at 4 min. Additionally, the CPU running time increases rapidly when the start time window or the station dwell time window widens. This indicates that an increase in the number of train paths leads to model complexity and longer solving times. Meanwhile, extending the length of the start time window leads to an improvement in the quality of the result and a reduction in the CPU running time, as compared with the case when the length of the station dwell time window is increased.

5.1.3. Benefit of Mesoscopic Network Compared with Microscopic Network. In the TPST network, we do not divide the track resources of a track segment into block sections, which is the modeling unit from the microscopic perspective. Instead, we split the platform track and consider the track resources in a station bottleneck area in terms of station route, which allows us to assign station resources, as opposed to the case in the macroscopic solving framework. Therefore, the TPST network can be seen as a modeling method from the mesoscopic perspective. We assume that there are 10 extra block sections of a track segment at each direction between two stations, and every place that will be used for trains to change the route in the throat area is marked as a node from the microscopic perspective.

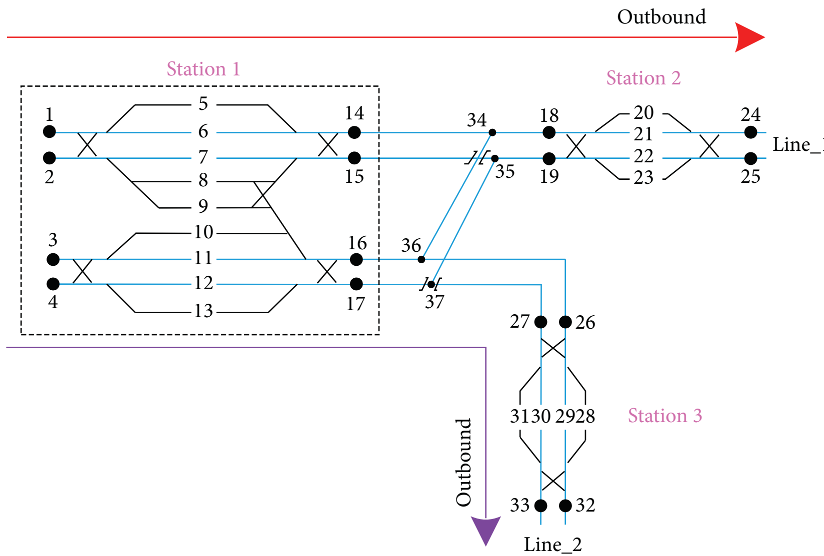

FIGURE 6: Three-station network with 37 nodes and 66 physical links.

TABLE 4: Train data for small railway network.

Train ID	From node	To node	Station	Start time window	Min dwell
1	25	2	2; 1	(4, 9)	1; 1
2	3	32	1; 3	(11, 16)	1; 0
3	25	32	2; 1;3	(7, 12)	0; 1;1
4	1	24	1; 2	(1, 6)	1; 1
5	1	32	1; 3	(2, 7)	1; 1
6	33	24	3; 1;2	(2, 7)	1; 1;1
7	3	24	1; 2	(4, 9)	1; 0
8	3	32	1; 3	(6, 11)	0; 0
9	33	2	3; 1	(7, 12)	0; 1
10	33	4	3; 1	(10, 15)	1; 0
11	25	2	2; 1	(10, 15)	1; 0
12	25	4	2; 1	(14, 19)	0; 1
13	1	24	1; 2	(7, 12)	1; 1
14	1	24	1; 2	(8, 13)	0; 1
15	25	32	2; 1;3	(17, 22)	0; 1;0
16	1	24	1; 2	(13, 18)	0; 1
17	1	24	1; 2	(17, 22)	1; 0
18	25	2	2; 1	(25, 30)	1; 1
19	25	32	2; 1;3	(33, 38)	1; 1;1
20	33	4	3; 1	(14, 19)	0; 1
21	33	24	3; 1;2	(17, 22)	1; 1;0
22	1	32	1; 3	(15, 20)	1; 0
23	3	32	1; 3	(20, 25)	1; 0
24	33	24	3; 1;2	(21, 26)	0; 1;0
25	33	2	3; 1	(24, 29)	1; 1
26	33	4	3; 1	(30, 35)	0; 0
27	3	32	1; 3	(24, 29)	0; 0
28	33	24	3; 1;2	(35, 40)	1; 1;1
29	3	32	1; 3	(28, 33)	1; 1
30	25	2	2; 1	(30, 35)	0; 0
31	1	24	1; 2	(20, 25)	1; 0
32	25	32	2; 1;3	(37, 42)	1; 1;1
33	1	24	1; 2	(24, 29)	0; 0
34	3	32	1; 3	(32, 37)	0; 0
35	33	4	3; 1	(37, 42)	1; 1
36	33	24	3; 1;2	(40, 45)	0; 1;0
37	1	24	1; 2	(39, 44)	1; 1
38	1	24	1; 2	(43, 48)	1; 0
39	3	32	1; 3	(44, 49)	1; 1
40	33	4	3; 1	(44, 49)	0; 1
41	33	24	3; 1;2	(47, 52)	1; 1;0
42	25	32	2; 1;3	(44, 49)	0; 1;1
43	3	32	1; 3	(50, 55)	1; 0
44	1	24	1; 2	(47, 52)	1; 0

Table 5: Performance comparison with an increase in the number of trains.

Scenario	Number of trains	Objective	CPU time (s)
1-1	5	95.9	0.1
1-2	10	205.9	1.3
1-3	15	306.1	6
1-4	20	417.1	6.3
1-5	25	553.2	123.0
1-6	30	667.0	116.4
1-7	35	782.8	163.2
1-8	40	888.2	366.1
1-9	44	997.9	429.8

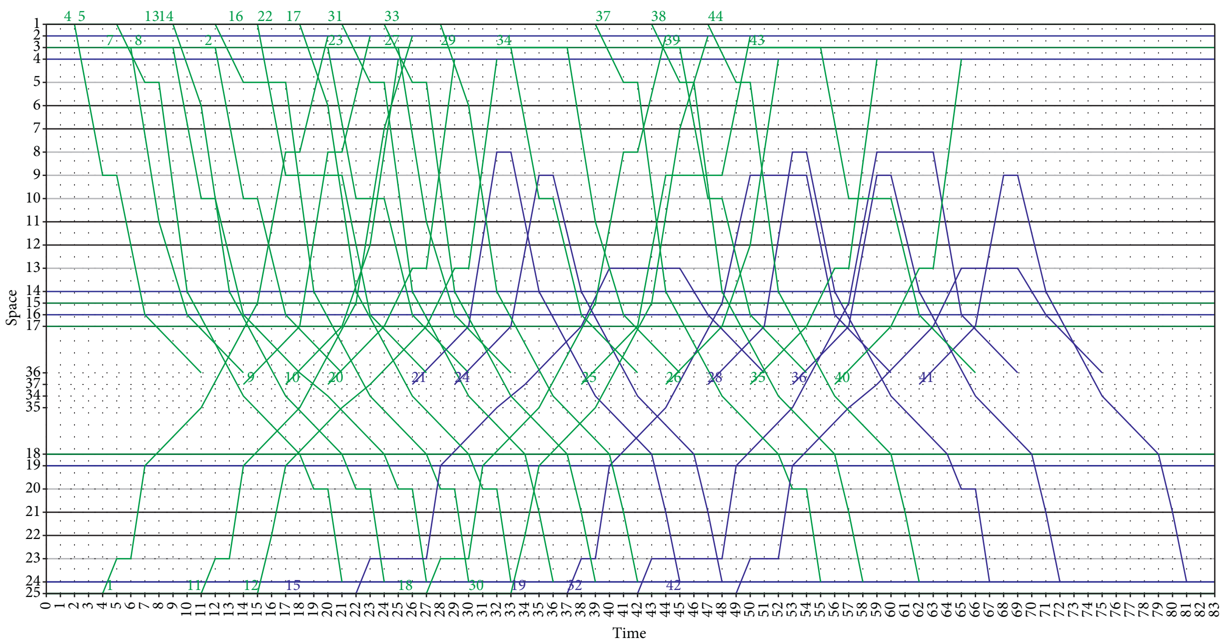

Figure 7: Train timetabling and platforming result for Line 1.

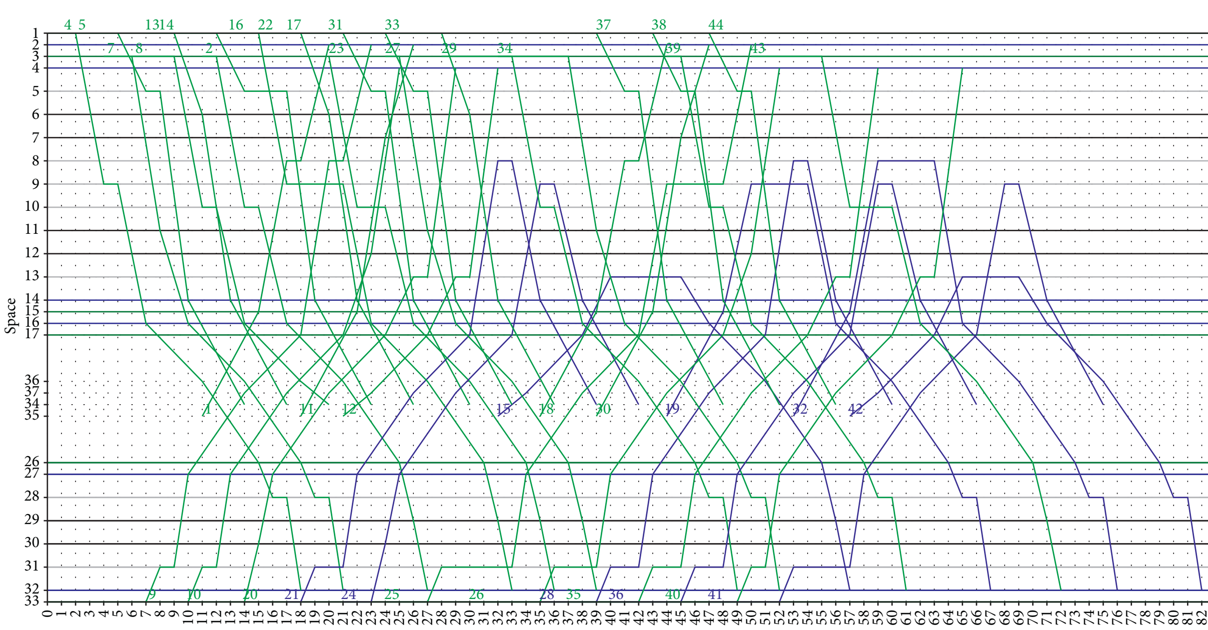

Figure 8: Train timetabling and platforming result for Line 2.

Table 6: Performance comparison with an increase in the number of train paths.

Scenario	Length of start time window (min)	Length of station dwell time window (min)	Objective	CPU time (s)
2-1	2	4	902.8	45.3
2-2	5		760.3	116.4
2-3	10		692.2	177.7
2-4	15		691.3	379.7
2-5	20		691.1	820.6
2-6	25		691.1	1041.6
2-7	4	2	902.6	17.4
2-8		5	775.3	285.4
2-9		10	708.6	231.6
2-10		15	703.5	402.1
2-11		20	703.5	1566

Table 7: Network size comparison from the microscopic and mesoscopic perspectives.

Perspective	Node number	Physical link number
Microscopic	132	>320
Mesoscopic	37	66

Table 7 provides a comparison of network scale from the microscopic and mesoscopic perspectives. As can be seen, the number of nodes can be reduced by at least 70% from 132 in the microscopic framework to 37 in our proposed framework, and the number of physical links can be reduced by almost 80% from 320 to 66. When we extend the physical network along the time dimension, the differences between the numbers of vertices and arcs from the two perspectives will be even larger. Therefore, the number of decision variables in our proposed model is also decreased sharply compared to the microscopic model.

5.1.4. Benefit of Mesoscopic Network Compared with Macroscopic Network. As described in Cacchiani et al. [31], in the macroscopic method, only the arrival and departure time constraints enforce train headway within a station. Further, as explained in Zhang et al. [2], the maximum number of trains within a station at a time should not exceed the station capacity. The station track capacity generally refers to the total number of station platform tracks. As we mentioned earlier, in the mesoscopic railway network, the detailed station layout can be considered, which reduces the difficulty of separating trains according to the train direction in a complex hub station. The constraints within a station are enforced by not only the arrival (departure) headway but also the platform usage and bottleneck resources.

Figure 9 represents the results within Station 1 between 48 min and 65 min in Figure 7. The solid line denotes the train paths solved by our proposed model from the mesoscopic perspective. Train 32 cannot arrive at the arrival boundary of Station 1 earlier than 48 min because of other train path constraints (such as minimum dwell time, start time at the origin node, and running time). According to the station layout, if train 32 arrives at the station through node 15 and departs from node 16, only platforms 8 and 9 can be used.

(1) If the Station Capacity Is Set as the Total Number of Platforms for Station 1. Train 32 will arrive at Station 1 at 51 min constrained by the arrival headway from the macroscopic perspective, and, no platform is available for this train at 53 min.

(2) If the Station Capacity Is Set as 2 (Platforms 8, 9) for Station 1. Train 32 will arrive at Station 1 at 53 min constrained by the arrival headway from the macroscopic perspective. Path 1 and Path 2 can be chosen as the local route for this train. However, the receiving route of both paths will conflict with the departure route of Train 28.

Consequently, the mesoscopic modeling framework, which considers the station resource, can generate a train timetable as well as a feasible train platform plan.

5.2. Medium-Sized Rail Network and Performance Analysis. To further verify the validity of our model, we modify our small-sized rail network to a medium-sized one comprising nine stations on two railway lines with 97 nodes and 188 physical links, as shown in Figure 10. The running time of each track segment on each railway line is either 3 or 4 min. The running time of a single-station route is 2 min at Stations 1, 4, 5, 8, 9, and 1 min at the remaining stations. The other parameters can be set as those in the cases of a small-sized railway network.

The train data are listed in Table 8. In total, 48 trains having conflicting routes are planned to be scheduled. The maximum dwell time for every train at each station is 5 min more than its minimum dwell time.

In this section, we also discuss the efficiency of problem solving when the number of trains is increased, as presented in Table 9. The solving time for each case is limited to 1 hour. The upper and lower bounds are listed in the second and third columns. The GAP in the fourth column is the difference between the upper and lower bounds divided by the upper bound. In general, for a medium-sized rail network, the greater the number of trains, the bigger the

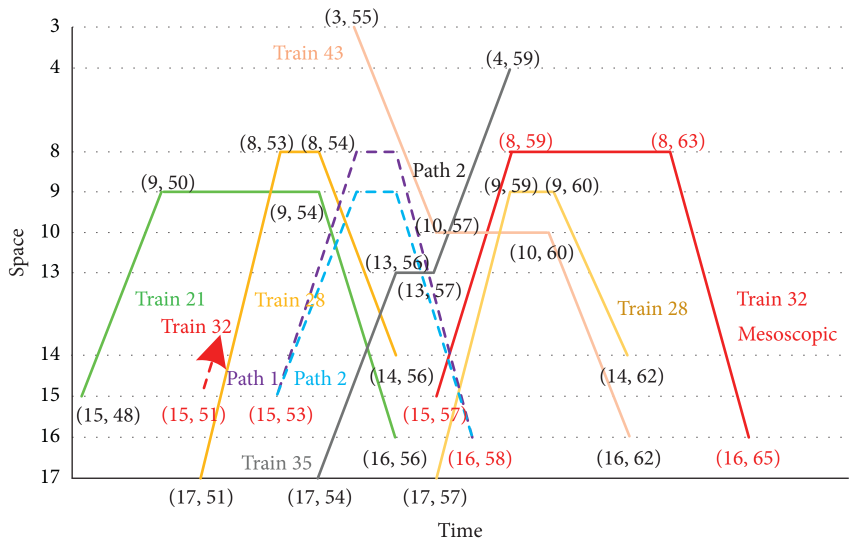

Figure 9: Comparison of the results from the mesoscopic and macroscopic perspectives.

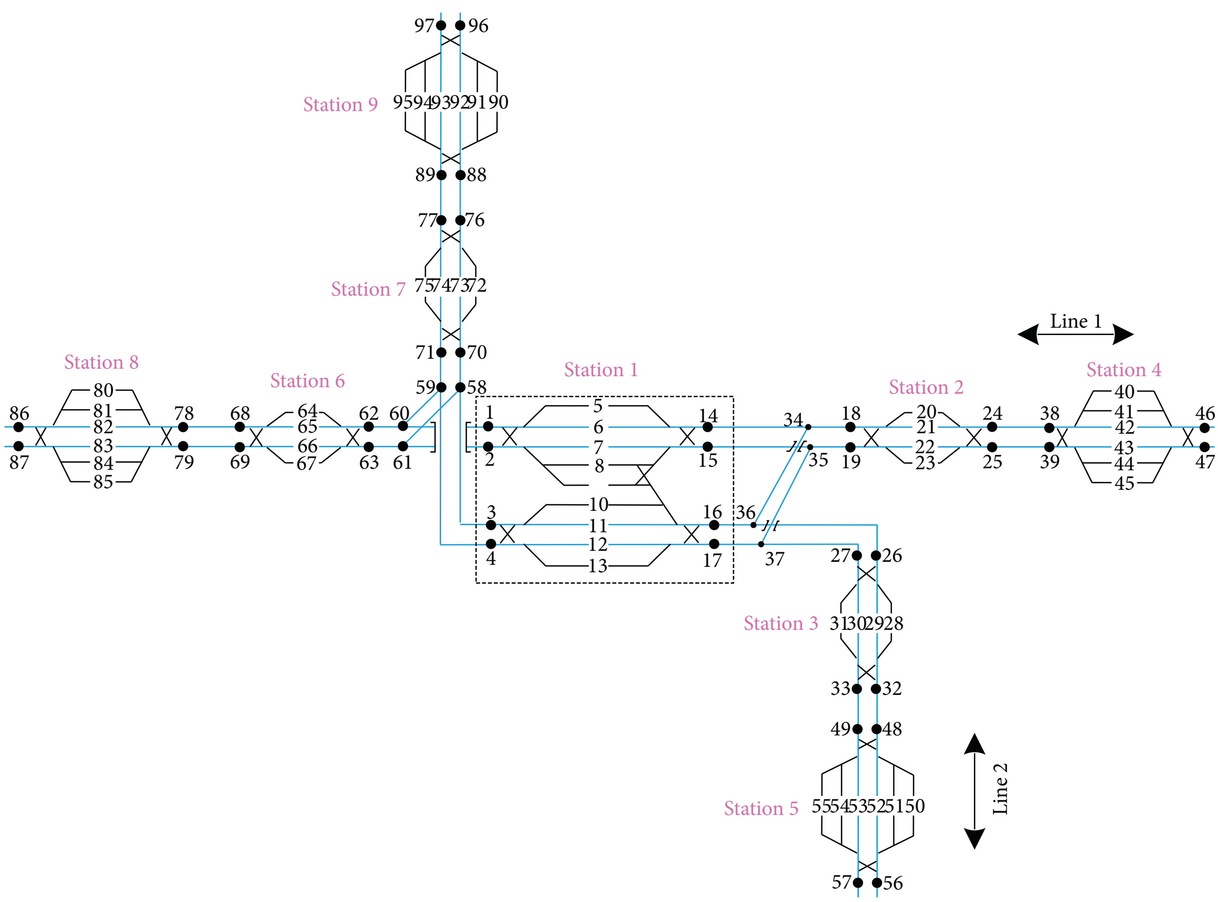

Figure 10: Medium-sized rail network.

gap. Moreover, the gap is still acceptable in the case with 48 trains. The final results for 48 trains are represented in Figures 11 and 12. A total of 11 trains cannot be scheduled into the timetable due to the conflicting routes, and the paths of all the scheduled trains satisfy all the constraints. Based on these results, it can be said that our proposed modeling method can successfully solve relatively large-scale cases.

Table 8: Train data for medium-sized railway network.

Train ID	From node	To node	Station	Start time window	Min dwell
1	86	46	8; 6;1; 2;4	(1, 6)	1; 0;1; 0;0
2	47	87	4; 2;1; 6;8	(2, 7)	1; 0;0; 1;1
3	96	56	9; 7;1; 3;5	(2, 7)	1; 0;0; 0;0
4	57	46	5; 3;1; 2;4	(3, 8)	1; 0;1; 1;1
5	86	56	8; 6;1; 3;5	(4, 9)	1; 1;1; 0;1
6	47	56	4; 2;1; 3;5	(5, 10)	1; 1;1; 1;1
7	96	46	9; 7;1; 2;4	(5, 10)	0; 0;1; 0;1
8	57	87	5; 3;1; 6;8	(6, 11)	1; 1;1; 0;1
9	86	46	8; 6;1; 2;4	(7, 12)	0; 0;0; 1;1
10	47	87	4; 2;1; 6;8	(8, 13)	0; 1;1; 0;1
11	96	87	9; 7;6; 8	(8, 13)	1; 1;0; 0
12	57	97	5; 3;1; 7;9	(9, 14)	1; 1;1; 1;1
13	86	46	8; 6;1; 2;4	(10, 15)	1; 1;0; 1;0
14	96	56	9; 7;1; 3;5	(11, 16)	1; 1;1; 1;0
15	47	97	4; 2;1; 7;9	(11, 16)	0; 1;1; 1;1
16	57	97	5; 3;1; 7;9	(12, 17)	1; 1;1; 1;1
17	86	46	8; 6;1; 2;4	(13, 18)	1; 1;1; 0;1
18	47	56	4; 2;1; 3;5	(14, 19)	1; 1;1; 1;1
19	96	56	9; 7;1; 3;5	(14, 19)	0; 1;1; 1;0
20	57	46	5; 3;1; 2;4	(15, 20)	1; 1;1; 0;1
21	86	97	8; 6;7; 9	(16, 21)	1; 1;1; 1
22	47	87	4; 2;1; 6;8	(17, 22)	0; 0;0; 0;0
23	96	56	9; 7;1; 3;5	(17, 22)	0; 1;1; 0;1
24	57	46	5; 3;1; 2;4	(18, 23)	0; 0;1; 1;1
25	86	46	8; 6;1; 2;4	(19, 24)	1; 1;1; 1;1
26	47	56	4; 2;1; 3;5	(20, 25)	1; 1;1; 1;1
27	96	56	9; 7;1; 3;5	(20, 25)	0; 0;1; 1;1
28	57	87	5; 3;1; 6;8	(21, 26)	1; 0;1; 0;1
29	86	56	8; 6;1; 3;5	(22, 27)	0; 1;1; 1;1
30	47	87	4; 2;1; 6;8	(23, 28)	1; 0;1; 1;0
31	96	56	9; 7;1; 3;5	(23, 28)	1; 1;1; 1;1
32	57	97	5; 3;1; 7;9	(24, 29)	1; 0;1; 1;0
33	86	46	8; 6;1; 2;4	(25, 30)	0; 0;1; 0;1
34	47	56	4; 2;1; 3;5	(26, 31)	1; 1;1; 0;1
35	96	56	9; 7;1; 3;5	(26, 31)	1; 1;1; 1;1
36	57	46	5; 3;1; 2;4	(27, 32)	1; 1;1; 0;1
37	86	46	8; 6;1; 2;4	(28, 33)	1; 0;1; 1;1
38	47	56	4; 2;1; 3;5	(29, 34)	0; 1;1; 1;1
39	96	56	9; 7;1; 3;5	(29, 34)	1; 1;1; 1;1
40	57	97	5; 3;1; 7;9	(30, 35)	0; 1;0; 1;0
41	86	46	8; 6;1; 2;4	(31, 36)	1; 1;1; 1;1
42	96	87	9; 7;6; 8	(32, 37)	1; 1;1; 1
43	57	46	5; 3;1; 2;4	(33, 38)	0; 1;1; 1;1
44	86	46	8; 6;1; 2;4	(34, 39)	1; 1;1; 1;0
45	57	97	5; 3;1; 7;9	(36, 41)	0; 1;1; 0;1
46	86	46	8; 6;1; 2;4	(37, 42)	1; 1;1; 0;1
47	57	46	5; 3;1; 2;4	(39, 44)	1; 1;1; 0;1
48	86	97	8; 6;7; 9	(40, 45)	0; 1;0; 1

Table 9: Performance comparison with an increase in the number of trains for medium-sized cases.

Number of trains	Upper bound	Lower bound	GAP (%)	CPU time (s)
1-15	700	691.1	1.28	3600
1-20	959.5	951.7	0.81	3600
1-25	1207	1186.4	1.17	3600
1-30	1598.7	1513.6	5.32	3600
1-35	2040.4	1837.6	9.94	3600
1-40	2424.8	2203.6	9.12	3600
1-48	2983.6	2622.5	12.06	3600

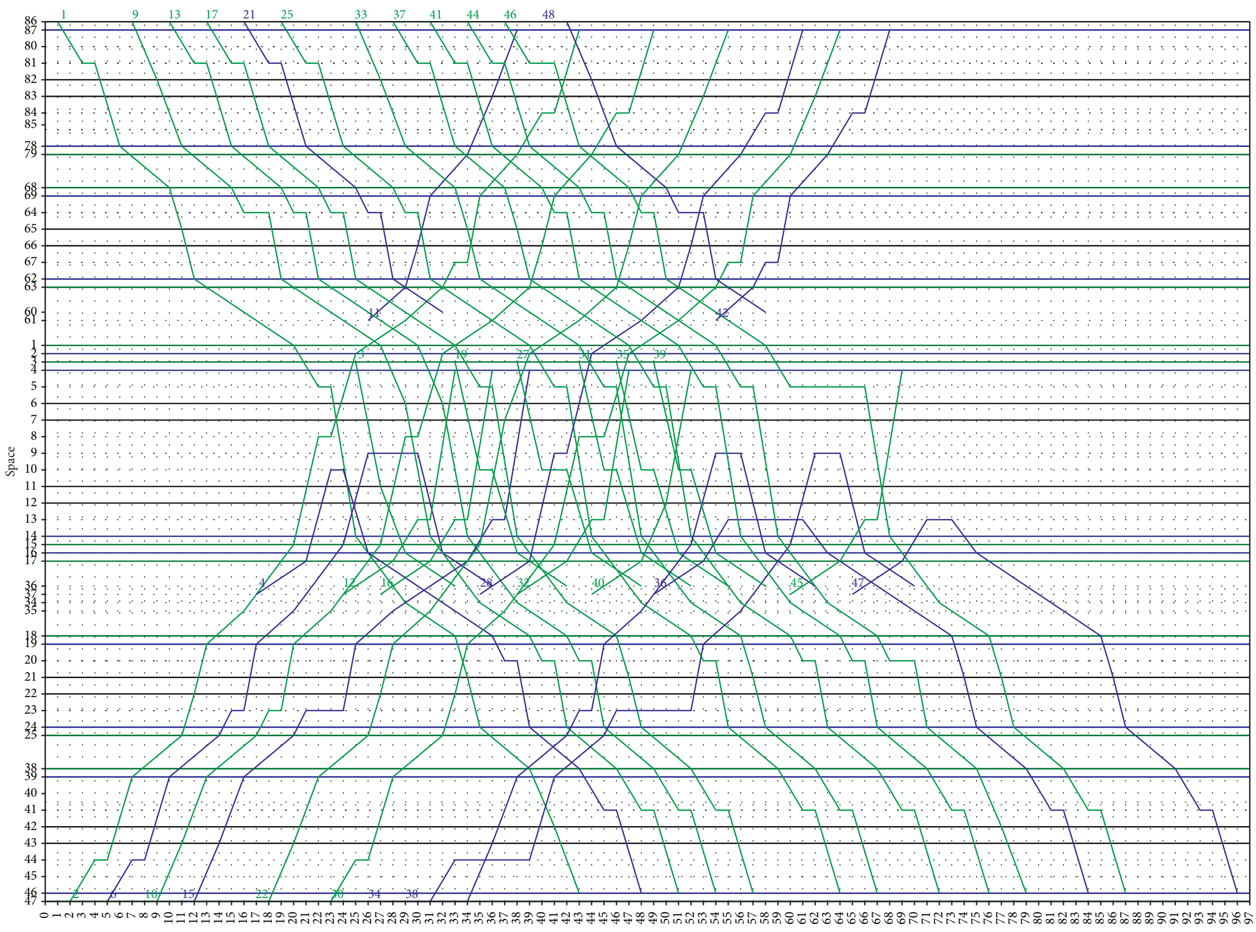

Figure 11: Train timetabling and platforming result for Line 1 for medium-sized rail network.

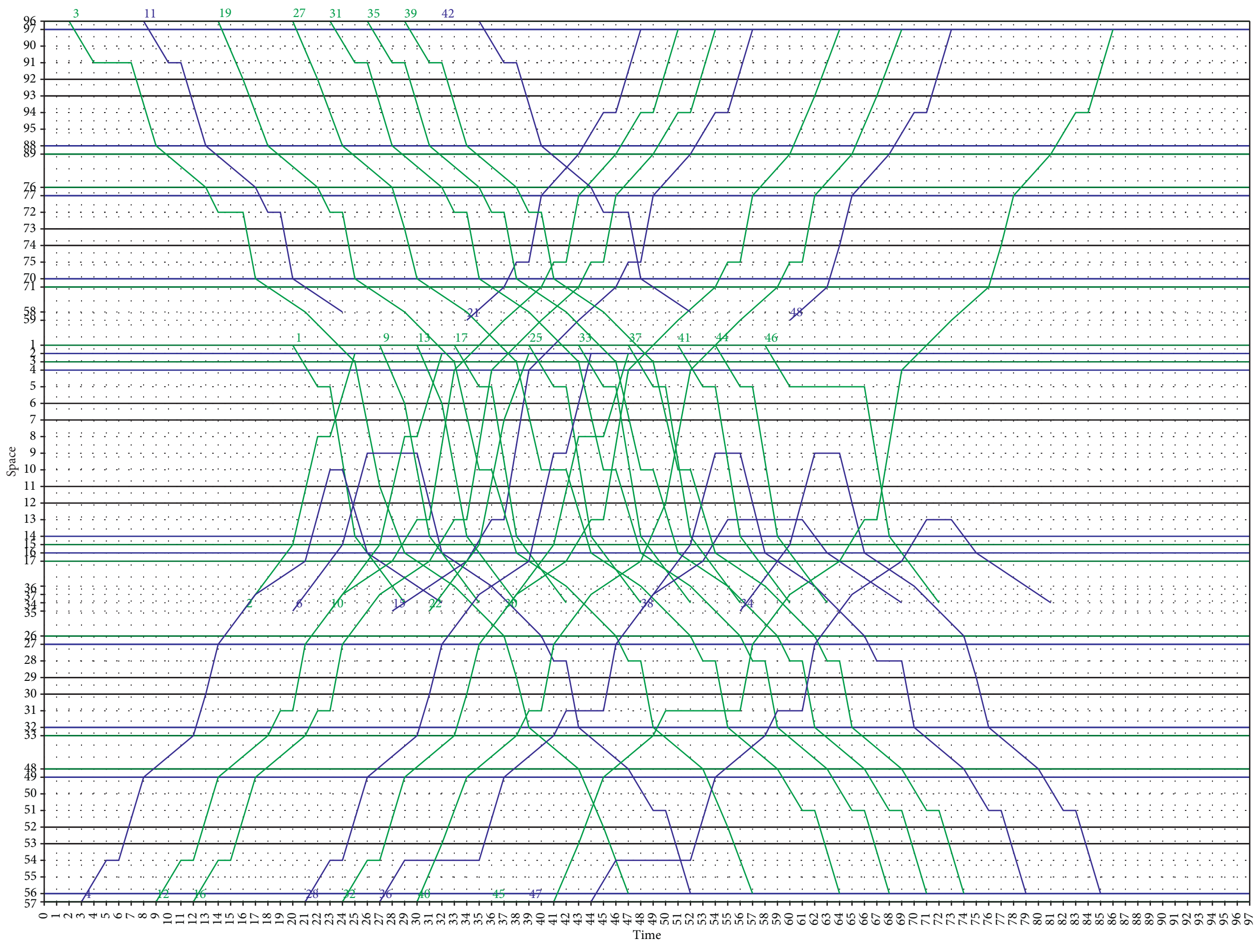

Figure 12: Train timetabling and platforming result for Line 2 for medium-sized rail network.

6. Conclusion and Future Research

In this study, we developed a framework to simultaneously optimize the train timetabling and platforming problems. First, we proposed the TPST network from a mesoscopic perspective. Compared to the railway network modeled from the macroscopic perspective, our proposed framework considers station track resources, including not only the platform track resources but also the track resources in the station bottleneck area. Compared to the microscopic network, the TPST network reduces the network scale by 70% and therefore decreases the number of decision variables. We also highlighted the performance difference with increases in the numbers of trains and train paths. Meanwhile, extending the length of the start time window leads to an improvement in the quality of the result and a reduction in the CPU running time, as compared with the case when the length of the station dwell time window is increased. In future research, a new algorithm should be developed to obtain an optimal or near-optimal solution for a large-scale railway network.

Data Availability

The detailed train and railway data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was supported by the Fundamental Research Funds for the Central Universities (no. 2019YJS082).

References

[1] Y. Zhang, A. D'Ariano, B. He, and Q. Peng, "Microscopic optimization model and algorithm for integrating train timetabling and track maintenance task scheduling," *Transportation Research Part B: Methodological*, vol. 127, pp. 237–278, 2019.

[2] C. Zhang, Y. Gao, L. Yang, Z. Gao, and J. Qi, "Joint optimization of train scheduling and maintenance planning in a railway network: a heuristic algorithm using Lagrangian relaxation," *Transportation Research Part B: Methodological*, vol. 134, pp. 64–92, 2020.

[3] W. Zhou, J. Tian, L. Xue, M. Jiang, L. Deng, and J. Qin, "Multi-periodic train timetabling using a period-type-based Lagrangian relaxation decomposition," *Transportation Research Part B: Methodological*, vol. 105, pp. 144–173, 2017.

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

[5] M. Carey and I. Crawford, "Scheduling trains on a network of busy complex stations," *Transportation Research Part B: Methodological*, vol. 41, no. 2, pp. 159–178, 2007.

[6] M. Carey and S. Carville, "Scheduling and platforming trains at busy complex stations," *Transportation Research Part A: Policy and Practice*, vol. 37, no. 3, pp. 195–224, 2003.

[7] X. Tian and H. Niu, "A bi-objective model with sequential search algorithm for optimizing network-wide train timetables," *Computers and Industrial Engineering*, vol. 127, pp. 1259–1272, 2019.

[8] 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.

[9] M. Samà, P. Pellegrini, A. D'Ariano, J. Rodriguez, and D. Pacciarelli, "Ant colony optimization for the real-time train routing selection problem," *Transportation Research Part B: Methodological*, vol. 85, pp. 89–108, 2016.

[10] 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.

[11] Y. N. Sotskov and O. Gholami, "Shifting bottleneck algorithm for train scheduling in a single-track railway," *IFAC Proceedings*, vol. 45, no. 6, pp. 87–92, 2012.

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

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

[14] F. Jiang, V. Cacchiani, and P. Toth, "Train timetabling by skip-stop planning in highly congested lines," *Transportation Research Part B: Methodological*, vol. 104, pp. 149–174, 2017.

[15] Y. Zhang, Q. Peng, Y. Yao, X. Zhang, and X. Zhou, "Solving cyclic train timetabling problem through model reformulation: extended time-space network construct and alternating direction method of multipliers methods," *Transportation Research Part B: Methodological*, vol. 128, pp. 344–379, 2019.

[16] X. Luan, J. Miao, L. Meng, F. Corman, and G. Lodewijks, "Integrated optimization on train scheduling and preventive maintenance time slots planning," *Transportation Research Part C: Emerging Technologies*, vol. 80, pp. 329–359, 2017.

[17] Q. Zhang, R. M. Lusby, P. Shang, and X. Zhu, "Simultaneously re-optimizing timetables and platform schedules under planned track maintenance for a high-speed railway network," *Transportation Research Part C: Emerging Technologies*, vol. 121, Article ID 102823, 2020.

[18] Y. Gao, L. Kroon, L. Yang, and Z. Gao, "Three-stage optimization method for the problem of scheduling additional trains on a high-speed rail corridor," *Omega*, vol. 80, pp. 175–191, 2018.

[19] C. Zhang, Y. Gao, L. Yang, U. Kumar, and Z. Gao, "Integrated optimization of train scheduling and maintenance planning on high-speed railway corridors," *Omega*, vol. 87, pp. 86–104, 2019.

[20] D. Cardillo, D. Luca, and N. Mione, "k L-list λcolouring of graphs," *European Journal of Operational Research*, vol. 106, no. 1, pp. 160–164, 1998.

[21] P. J. Zwaneveld, L. G. Kroon, H. E. Romeijn et al., "Routing trains through railway stations: model formulation and algorithms," *Transportation Science*, vol. 30, no. 3, pp. 181–194, 1996.

[22] P. J. Zwaneveld, L. G. Kroon, and S. P. M. Van Hoesel, "Routing trains through a railway station based on a node packing model," *European Journal of Operational Research*, vol. 128, no. 1, pp. 14–33, 2001.

[23] A. Billionnet, "Using integer programming to solve the train-platforming problem," *Transportation Science*, vol. 37, no. 2, pp. 213–222, 2003.

[24] J. Wu, L. Kang, H. Sun, and X. Jia, "Track allocation optimization in railway station: mean-variance model and case study," *Journal of Transportation Engineering*, vol. 139, no. 5, pp. 540–547, 2013.

[25] P. Pellegrini, G. Marlière, and J. Rodriguez, "Optimal train routing and scheduling for managing traffic perturbations in complex junctions," *Transportation Research Part B: Methodological*, vol. 59, pp. 58–80, 2014.

[26] Q. Zhang, X. Zhu, and Li Wang, "Track allocation optimization in high-speed railway stations from infrastructure management and service perspectives," *Measurement and Control*, vol. 51, no. 7-8, pp. 243–259, 2018.

[27] Q. Zhang, X. Zhu, and L. Wang, "Track allocation optimization in multi-direction high-speed railway stations," *Symmetry*, vol. 11, no. 4, p. 459, 2019.

[28] A. Karoonsoontawong and A. Taptana, "Branch-and-bound-based local search heuristics for train timetabling on single-track railway network," *Networks and Spatial Economics*, vol. 17, no. 1, pp. 1–39, 2017.

[29] S. Binder, Y. Maknoon, and M. Bierlaire, "The multi-objective railway timetable rescheduling problem," *Transportation Research Part C: Emerging Technologies*, vol. 78, pp. 78–94, 2017.

[30] L. G. Kroon and L. W. P. Peeters, "A variable trip time model for cyclic railway timetabling," *Transportation Science*, vol. 37, no. 2, pp. 198–212, 2003.

[31] V. Cacchiani, L. Galli, and P. Toth, "A tutorial on non-periodic train timetabling and platforming problems," *Euro Journal on Transportation and Logistics*, vol. 4, no. 3, pp. 285–320, 2015.