An integrated model for train rescheduling and station track assignment

Xuelei Meng1,2 | Yahui Wang3 | Wanli Xiang1 | Limin Jia2

1 School of Traffic and Transportation, Lanzhou Jiaotong University, No. 88, Anning West Road, Lanzhou, China
2 State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing, China
3 School of Foreign Languages, Lanzhou Jiaotong University, No. 88, Anning West Road, Lanzhou, China

Correspondence
Xuelei Meng, School of Traffic and Transportation, Lanzhou Jiaotong University, No. 88, Anning West Road, Lanzhou, China.
Email: mengxl_lzjtu@163.com

Abstract
Both train rescheduling and station track assignment have become hot topics in recent years. It is fundamentally important to do the rescheduling and track assignment work at the same time to avoid the feasibility risk of the re-scheduled timetable. The purpose of this paper is to design an integrated model for train rescheduling and track assignment in order to provide an integrative plan for the trains to run on the railway sections and go through stations. Based on the existing train rescheduling model, the model is designed by adding the constraints and the optimization goal of track assignment. The goal of track assignment is to maximize the equilibrium of the track usage time, and the constraint is that two trains cannot occupy a same track at the same time. An artificial bee colony algorithm is used to solve the model to get the operation plan. A computing experiment was carried out to prove the effectiveness of the model and the efficiency of the algorithm. The approach presented in this paper can provide a reference for the developers of a railway dispatching system.

1 INTRODUCTION

Today’s railway system plays an important role in the transportation of passengers and freights. The capacity of the railway system depends on the railway facilities and the transportation organization plans. The most important plan is the train timetable, which decides the train operation order, the arrival time and the departure time of the trains at stations. It determines the quality of the railway system operation. Obviously, it is an essential task to design the train timetable and reschedule the trains in emergencies to keep the railway system operating smoothly.

We now recognize that it is not enough to design the arrival and departure time of the trains at stations. It is possible that a re-scheduled timetable is not feasible, since a certain train cannot enter a station if there is no unoccupied arrival and departure track when the train approaches the station. The indirect reason for this is that too many trains that will arrive at a station in a short period of time. This leads to the result that the number of arrival and departure tracks is not large enough to accommodate the trains. For instance, the arrival and departure tracks are numbered as I, II, 3, 4, 5, and 6 in Figure 1, which obeys the Chinese railway management rules. Tracks I, 3 and 5 are set to accommodate the south-going trains that are all occupied in Figure 1. At this time, train NC cannot enter the station.

Thus, it is quite necessary to do the track assignment work when rescheduling trains in the dispatching sections. Track
assignment can reduce the risk of infeasibility for the timetable and lower the difficulty of dispatching work. In this paper, we integrate the train rescheduling and the track assignment problems, build a model to describe the problem, and present a highly individualized algorithm to solve the problem.

2 | RELATED WORKS

Train scheduling and rescheduling problems have been hot topics for a long period of time. Caprara et al. (2006) designed a train timetabling system that took into account the manual block, the station capacities and the trains to be inserted into the prescribed timetable [1]. Tornquist and Persson (2007) presented an optimization approach to the problem of rescheduling railway traffic in an n-tracked network when a disturbance had occurred [2]. Abril et al. (2008) took railway scheduling problems as constraint satisfaction problems (CSPs) and designed a distributed and asynchronous search algorithm (DTS) to solve the problems [3]. Castillo et al. (2011) studied the rescheduling problem for a mixed multiple- and single-tracked railway network [4]. Corman et al. (2012) studied the problem of coordinating several dispatchers with the objective of solving the train rescheduling problem in the related dispatching railway sections [5]. Min et al. (2011) provided a column-generation-based algorithm to resolve the train-conflict problem on a metropolitan railway network [6]. Kang et al. (2016) studied the last train rescheduling problem. They proposed a mathematical model, taking the difference between the original timetable and the rescheduled timetable as the optimization goal [7]. Wang et al. (2016) used timed coloured petri nets to model the train rescheduling problem and designed a heuristic algorithm to search for an optimal or near-optimal feasible schedule [8]. Šemrov et al. (2016) designed a reinforcement learning approach to solve the train rescheduling problem on a single-track railway [9]. Yang et al. (2015) proposed an optimization method to schedule trains for reducing the energy consumption and travel time synchronously, formulating an integer-programming model with timetable and speed control [10]. Lan and Lin (2005) presented a four-stage DEA approach to measure the railway transport technical efficiency and service effectiveness, and strategies for operating railways [11]. Yang et al. (2018) developed an energy-efficient rescheduling approach under delay perturbations for metro trains, which aimed to minimize the net energy consumption under the premise of reducing or eliminating the delay altogether [12]. Chen et al. (2019) attempted to develop a draft passenger train timetable under the constrained capacities of arrival–departure tracks and servicing tracks for the passenger train coaching stocks in a passenger railway network [13]. Caimi et al. (2012) proposed a dispatching assistant in the form of a model predictive control framework for a complex central railway station area, building a binary linear optimization model, which assigned precomputed blocking-stairways to trains while respecting resource-based clique constraints, connection constraints, platform related constraints and consistency constraints with the objective of maximizing customer satisfaction [14]. Corman (2020) merged train rescheduling problem and routing passengers’ problem and solved this problem with a game theoretical approach, focusing on the solutions corresponding to Nash equilibria of a game involving passengers and infrastructure managers [15]. Zhu and Goverde studied the timetable rescheduling during disruptions, particularly in the case where all tracks between two stations are blocked for hours [16]. Zinder et al. (2020) presented a polynomial-time algorithm for rescheduling traffic when one track of a double-track railway became unavailable, the remaining track had a siding, and there were two categories of trains—priority trains such as passenger trains and ordinary trains such as the majority of freight trains [17]. Zhu and Goverde (2019) proposed a novel timetable rescheduling model, which included flexible stopping and flexible short-turning as well as retiming, reordering, and cancelling trains [18]. Feng et al. (2019) proposed a new dynamic model based on the discrete time method and provides some efficient policies to control the train delay propagation correspondingly [19]. Jin et al. (2019) described a peer-to-peer system to solve train rescheduling problems in railway network bottlenecks and designed a genetic algorithm to solve the problem [20]. Jiang et al. (2019) identified the main variables that contribute to train delay and constructed different model to predict the primary delay recovery [21]. Katori and Izumi (2019) described how to generate automatic rescheduling methods while specifically considering the operated rolling stock types [22]. Josyula et al. (2018) presented the solution space of the train rescheduling problem as a binary tree and designed and implement two fast heuristic algorithms—a sequential, and a parallel [23]. Martinis presented a feed forward simulation-based model that performed speed profile optimization together with minor rescheduling actions with the purpose to provide railway operators and infrastructure managers with energy-efficient solutions that were tailored especially for freight trains [24]. Iswardani and Rusdiansyah studied the development of rescheduling considering predictive delay to minimize the total delay time [25]. Kato et al. (2018) presented an algorithm for extraction of unpredictable delays with original and resultant train timetable data using train operation simulation [26]. Francisco et al. (2018) constructed a model of transportation on-demand to address an incidence which happened in a railway line, forcing to reschedule the service in such transit lines [27]. Shakibayifar et al. (2019) constructed a multi-objective simulation based optimization framework to effectively manage the train conflicts after the occurrences of a disturbance caused by a temporary line blockage [28].

**FIGURE 1** A sketch to describe the track assignment problem at a station.
Some of the papers focused on a real-time train rescheduling problem. Sato and Fukumura (2012) solved the locomotive rescheduling problem by changing the assignment of the locomotives to all trains and considering their periodic inspections [29]. Espinosa-Aranda and García-Ródenas (2013) formulated a real-time train rescheduling problem as a binary integer linear programming problem that tried to maximize consumer satisfaction by minimizing total passenger delay at destinations [30]. Zhan et al. (2015) formulated a mixed integer-programming model to solve a real-time high-speed train rescheduling problem [31]. Chen et al. (2015) also proposed a mixed-integer programming model to describe the real-time train rescheduling problems around bottleneck sections and designed an innovative improved algorithm to solve the problems [32]. Chen et al. (2015) developed a departure time domain for passenger trains under the constrained receiving and dispatching capacities of arrival–departure tracks in a railway station [33]. Dalapati et al. (2017) presented an approach to avoid the collisions in the railway traffic system, including the collisions between the trains in the railway sections and in the stations, which implied that the station tracks usage plan must be designed [34]. Bettinelli et al. (2017) studied the real-time resolution of conflicts arising in real-world train management applications, aiming at defining a set of actions which must be implemented to grant safety, e.g., to avoid potential conflicts such as train collisions or headway violations, and restore quality by reducing the delays [35]. Cavone et al. (2019) constructed a mesoscopic mixed integer linear programming model to solve the real-time train rescheduling problem, filling the gap between macroscopic and microscopic train rescheduling problem modelling [36]. Ikonen et al. (2019) proposed an approach where a neural network is trained to make online decisions on these quantities, as well as on the choice of the rescheduling method (mathematical programming or metaheuristics) [37]. Shakibayifar et al. (2018) proposed a multi-objective simulation-based optimization framework to effectively manage the train traffic after the occurrences of a disturbance caused by a partial/full blockage [38]. Ortega et al. (2018) addressed the rescheduling problem in a transit line that had suffered a fleet size reduction and presented different modelling possibilities depending on the restriction assumptions [39]. Galapitage et al. (2018) developed a system that combined real-time driving advice calculation with real-time junction scheduling to reduce delays at junctions, and had simulated the operation of this junction scheduling system at Neasden Junction in the United Kingdom [40]. Corman et al. (2018) reviewed and evaluated the indices applied to railway traffic control, comparing optimal rescheduling approaches and clarified the benefits of automated traffic control for infrastructure managers, railway operators and passengers [41]. Yang et al. (2019) designed a real-time timetable rescheduling method for energy optimization of metro systems, which took little time to recalculate a new schedule and gave proper solutions for all trains in the network immediately after a random disturbance happens [42]. Altazinab et al. (2020) proposed an approach combining multi-objective optimization, to select rescheduling decisions, and macroscopic simulation, to solve the real-time train rescheduling problem [43]. Huang et al. (2020) developed non-linear mixed integer programming (NMIP) models with two different recovery strategies to solve the train rescheduling problem and designed a two-stage approach to carry out the optimization process [44]. Much attention has also been paid to train routing in railway stations in recent years, which belongs at the control level. Zwaneveld et al. (1996) studied the train routing problem, analysing the problem from the strategic level, the tactical level and the operational level [45]. Kroon et al. (1997) studied the computational complexity of routing train problems at stations [46]. Billionnet (2003) used integer programming to solve the train platforming problem. The goal was to assign the trains to available tracks to make it possible to go through the stations [47]. Corman et al. (2010) addressed the problem of train conflict detection and resolution, which was dealt with every day to adapt the timetable to delays and other unpredictable events occurring in real-time [48]. Schasfoort et al. studied the real-time train assignment problem and modelled it as a mixed integer program that strove to minimize the total weighted delay of trains [49]. König and Schön (2020) presented an optimization model for delay management in railway networks, taking limited capacities of tracks and stations into consideration [50]. Vansteenwegen et al. (2019) proposed a conflict prevention strategy for large and complex networks in real-time railway traffic management based on the analysis of the impact of the unexpected events such as overcrowded platforms or small mechanical defects can cause conflicts [51]. Zhang et al. (2020) proposed a mixed integer linear programming model, which minimized the weighted sum of total train delays and the platform track assignment costs, subject to constraints defined by operational requirements [52]. Researchers have realized that it is necessary to studied train rescheduling and routing problem comprehensively. Carey and Carville (2003) integrated the train scheduling and platforming problems and constructed a comprehensive model to solve the problems [53]. Lee and Chen (2009) proposed an optimization heuristic that included both train pathing and train timetabling, which integrated the train pathing problem and the scheduling problem [54]. Meng and Zhou (2014) proposed a simultaneous approach to study the rerouting and rescheduling problems on an $n$-track network, decomposing the original complex rerouting and rescheduling problems into a sequence of single train optimization sub-problems [55]. Samà et al. (2017) presented a number of algorithmic improvements implemented in an optimization solver in order to improve the possibility of finding good quality solutions quickly for train scheduling and routing problems [56]. In fact, there is another control level problem—the speed control or traction plan. This problem also attracted the researchers’ attention. Xu et al. (2017) built a train rescheduling model that integrated speed management during disruptions of high-speed traffic under the quasi-moving block system [57]. Chang et al. (2017) proposed a model to integrate the train scheduling problem and the equipment usage planning problem, including the trucks, cranes and railway tracks [58]. Luan et al. (2018) studied the integration
of real-time traffic management and train control by using mixed-integer non-linear programming (MINLP) and mixed-integer linear programming (MILP) approaches [59]. Pellegrini et al. (2019) presented a mixed integer linear programming based heuristic for train re-routing and rescheduling problem, proposing valid inequalities to boost the performance of RECIFE-MILP [60].

All these publications have given us much inspiration when studying this problem. In this paper, we present a new model to solve the problem, especially describing the model constraints in a novel way.

3 | PROBLEM STATEMENT

The problem in this paper includes two sub-problems. One sub-problem is how to determine the arrival and departure time of the trains at stations. The other sub-problem is how to decide which arrival and departure track each train will occupy. The railway system may fall into a state of disorder, especially in emergencies. When rescheduling the trains, assignment of the arrival and departure track for the trains appears particularly important; this assignment will assure the feasibility of the rescheduled timetable. For example, we need to design the arrival time and the departure time of train NC1 and train NC2 at station 1 and station 2. When train NC1 approaches station 1, we can assign track 3 to accommodate it as the usage rule in normal conditions, which assigns tracks I, 3, or 5 to receive the south-going trains. However, we find that train NC2 cannot enter station 2 when it approaches because tracks I, 3, and 5 are all occupied by trains. Therefore, in this case, we cannot only consider the arrival and departure time of the trains; we must also the usage plan of the tracks of station 2. In emergencies, the usage rule of the tracks may be changed, allowing all of the tracks to receive all of the trains from both directions. Thus, we can assign train NC2 to occupy track 4 if it needs to stop at station 2.

In some of the references, the track assignment problem was considered for the condition that the station could accommodate the trains. The fact that the track assignment plan affected the train timetable was seldom considered. As seen in Figure 2, the arrival time of train NC2 at station 2 must be changed because there are no tracks left when it approaches station 2.

How can this problem be resolved? We need to change the arrival and departure time of the trains at the stations first to make it possible to platform the trains in order to allow them to arrive on time.

In this paper, we focus on the train rescheduling and track assignment problem, combine them and construct an integrated model. We will apply the artificial bee colony algorithm to solve the model to generate the train rescheduled timetable and track usage plan simultaneously. Since train rescheduling and track assignment problem is a very difficult problem, involving many interfering factors, we must define the scope of the study in this paper. We focused on the train rescheduling and track assignment problem, considering only short delays in the railway system. We only solve the problem on a single railway line, not on a railway network, with the automatic block system or movable block system. And we assume that human resources, including the dispatchers, the engineman and the crew can meet the requirements of the train rescheduled timetable.

4 | INTEGRATED MODEL FOR TRAIN RESCHEDULING AND TRACK ASSIGNMENT

4.1 | Decision variables and parameters

The purpose of the model is to determine the arrival and departure time of the trains at stations and the track assignment plan, which determines which train will occupy which track at a station.

The decision variables are the arrival time $a_{ij}$, the departure time $d_{ij}$ and the occupying sign $x_{i,j,k}$. The decision variables are showed in Table 1 and the parameters used in the model of this paper are listed in Table 2.
4.2 Constraints for train rescheduling

There are numerous prerequisite rules in railway operation designed to ensure safety and determined by facilities such as the blocking systems. The most important rule is to determine the relationships between the arrival and departure time of all the trains, to separate the trains in space. The system constraints are therefore designed as follows.

The difference between a forward train (the former one of two neighbour trains) arriving time and a backward train (the latter one of two neighbouring trains) arriving time at the same stations must be longer than the technical intervals, which produces the constraint shown in Equation (1).

\[
a_{i,j} - d_{i,j} \geq L_s, i \neq j, i, j = 1, 2, \ldots, N; j, i, j = 1, 2, \ldots, M; i \neq j
\]

Likewise, the difference between a backward train departing time and a forward train departing time from the same stations must be longer than the technical intervals. The constraint can be described as:

\[
d_{i,j} - a_{i,j} \geq l_s, i \neq j, i, j = 1, 2, \ldots, N; j, i, j = 1, 2, \ldots, M; i \neq j
\]

It is not allowed to receive and send trains simultaneous at some stations. The interval between two trains must satisfy the departing–arriving interval and arriving-departing interval.

\[
d_{i,j} - a_{i,j} \geq L_s, i \neq j, i, j = 1, 2, \ldots, N; j, i, j = 1, 2, \ldots, M; i \neq j
\]

Since \( t_{\text{depart} \rightarrow \text{arrive}} \) is the minimum time interval between a train leaving a station and another train arriving at the same station, then the constraints are defined in Equation (3).

\[
a_{i,j} - d_{i,j} > t_{\text{depart} \rightarrow \text{arrive}}, i, j = 1, 2, \ldots, N; j, i, j = 1, 2, \ldots, M; i \neq j
\]

The running time of each train according to the rescheduled timetable must be longer than the minimum running time, which can be formulated as follows.

\[
a_{i,j} - d_{i,j} \geq t_{\text{min, run}}^{i,j}, i = 1, 2, \ldots, N; j = 1, 2, \ldots, M; i \neq j
\]

Again, the dwelling time of each train must be longer than the minimum dwelling time, which produces the constraint below.

\[
d_{i,j} - a_{i,j} \geq t_{\text{min, dwell}}^{i,j}, i = 1, 2, \ldots, N; j = 1, 2, \ldots, M
\]

The passenger trains must not leave the stations before the time planned on the timetable, which is made available to the public, so there is a constraint as follows.

\[
d_{i,j} - a_{i,j} \geq 0, i = 1, 2, \ldots, N; j = 1, 2, \ldots, M
\]

4.3 Constraints for track assignment

At first, we analyse the possible variations of the relationship between arrival and departure time of two neighbouring trains if they use a same track at a station. We assume that it is at track 1, station 1. Then, all the probabilities are listed in Table 3 and showed in Figure 3. In each subfigure of Figures 3 and 4, the two lines in each subgraph stand for the occupying time of train 1 and train 2. \( a_{1,1} \) and \( d_{1,1} \) are the arrival and departure time of train 1 at station 1. \( a_{2,1} \) and \( d_{2,1} \) are the arrival and departure time of train 2 at station 1, as marked in Figures 3 and 4. In Figure 3(a), we can see that \( a_{2,1} < a_{1,1} < d_{2,1} < d_{1,1} \) so there is \( d_{1,1} - a_{2,1} > 0 \) and \( a_{1,1} - d_{2,1} < 0 \). Then if train 1 and train 2 occupy track 1 at station 1 (in this situation \( x_{1,1,1} = x_{2,1,1} = 1 \)), there is \( (x_{1,1,1} - d_{1,1} - x_{2,1,1} \cdot a_{2,1}) \)
Another constraint is the track assignment constraint. Just only one track must be assigned to a train when it goes through a station, no matter whether it stops or not. Then the constraint can be described as Equation (8).

$$\sum_{k=1}^{N} x_{i,j,k} = 1, \text{forevery } i$$

(8)

According to the operating rule, a train can stop at a main track as well as at a side track. But when it goes through a station without a stop, it must occupy the main track. So the constraint can be described as follows.

$$x_{i,j,1} = x_{i,j,11} = 1, \text{if } d_{i,j} = a_{i,j}$$

(9)

4.4 Optimizing goals of the model

We still take the total delay time as the first optimizing goal. The total delay time includes the delayed arrival and the departure time of each train at all the stations. Thus, the goal can be described as follows.

$$Z_1 = \sum_{j=1}^{N} \sum_{l=1}^{M} a_{i,j}[(a_{i,j} - a_{l,j}) + (d_{i,j} - d_{l,j})]$$

(10)

Since different types of trains have different weights, we will classify the trains into different groups and set weights for them by group.

In addition, different tracks assigned to the trains affect the running distance in the station, leading to the fact that the running time of a train in a station is determined by the selection of the track. So it is necessary to balance the usage intensity of all the tracks to reduce the maintenance costs. To maximize the track usage equilibrium of a station, we take it as an optimizing goal in this paper, defined as follows.

$$Z_2 = \sum_{j=1}^{N} \sqrt{\frac{\sum_{i=1}^{F_j} (X_{i,j} - \mu_j)^2}{F_j}}$$

(11)

where $X_{i,j}$ is a variable that records the occupied time of track $l$ at station $j$. If track $l$ at station $j$ is occupied by a train, it will be added 1. The goal to set this variable is to count the time that each track is occupied, to guide the assignment result to balance the use of tracks.

We have two optimization goals. One goal is the total delayed time of all the trains at all the stations. The other goal is the track usage equilibrium. Since the two goals do not have the same measurement unit, $\alpha$ is introduced to unify the measurement units before we can add the two goals together to generate the comprehensive optimizing object, which can be obtained by a parameter identification algorithm based on the related

### Figure 3
Four non-compliant variations of the relationship between two neighbouring trains when they use the same track at a station

### Figure 4
Two possible variations of the relationship between two neighbouring trains when they use the same track at a station

$$(x_{1,1,1} \cdot a_{1,1} - x_{2,1,1} \cdot d_{2,1}) < 0.$$. But we know this situation cannot be allowed in train operation work to avoid the train conflicts. In the same analysis manner, we find that the situations described in Figure 3(b), (c) and (d) are not allowed in reality. In Figure 4(a), $a_{1,1} < d_{1,1} < a_{2,1} < d_{2,1}$. Then if train 1 and train 2 occupy track 1 at station 1 (in this situation $x_{1,1,1} = x_{2,1,1} = 1$), there is $(x_{1,1,1} \cdot d_{1,1} - x_{2,1,1} \cdot a_{2,1}) \times (x_{1,1,1} \cdot a_{1,1} - x_{2,1,1} \cdot d_{2,1}) > 0$. This situation is allowed in the train dispatching work. For the same reason, $(x_{1,1,1} \cdot d_{1,1} - x_{2,1,1} \cdot a_{2,1}) \times (x_{1,1,1} \cdot a_{1,1} - x_{2,1,1} \cdot d_{2,1}) > 0$ is right according to Figure 4(b).

In addition, if one of the trains will not occupy track 1, then $x_{1,1,1} = 0$ or $x_{2,1,1} = 0$. Then we get $(x_{1,1,1} \cdot a_{1,1} - x_{2,1,1} \cdot d_{2,1}) \times (x_{1,1,1} \cdot a_{1,1} - x_{2,1,1} \cdot d_{2,1}) > 0$. To summarize all of the cases above, it can be concluded that $(x_{1,1,1} \cdot a_{1,1} - x_{2,1,1} \cdot d_{2,1}) \times (x_{1,1,1} \cdot a_{1,1} - x_{2,1,1} \cdot d_{2,1}) > 0$. We get the following inequality by extending this to the general case.

$$(x_{i,j,k} \cdot d_{i,j} - x_{i,j,k} \cdot a_{i,j}) \times (x_{i,j,k} \cdot a_{i,j} - x_{i,j,k} \cdot d_{i,j}) > 0,$$

$$i = 1, 2, ..., M; j = 1, 2, ..., N; i' = 1, 2, ..., M; j'$$

$$= 1, 2, ..., N; i \neq i', k = 1, 2, ..., F_j$$

(7)
data.

\[ Z = Z_1 + \alpha Z_2 \]  

(12)

Then the integrated model is constructed as follows:

\[
\begin{align*}
\text{min} & \quad Z \\
\text{s.t.} & \quad a_{ij} - a_{ij} \geq l_i, i, j = 1, 2, \ldots, N; j = 1, 2, \ldots, M; i \neq i' \\
& \quad d_{ij} - d_{ij} \geq l_i, i, j = 1, 2, \ldots, N; j = 1, 2, \ldots, M; i \neq i' \\
& \quad a_{ij} - d_{ij} > \tau_{\text{depart-arrive}}, i, j = 1, 2, \ldots, N; j = 1, 2, \ldots, M; i \neq i' \\
& \quad a_{ij} - d_{ij} \geq t_{\text{min,run}}, i = 1, 2, \ldots, N; j = 1, 2, \ldots, M; \Delta \neq j' \\
& \quad d_{ij} - a_{ij} \geq t_{\text{min,dwell}}, i = 1, 2, \ldots, N; k = 1, 2, \ldots, M \\
& \quad d_{ij} - d_{ij} \geq 0, i = 1, 2, \ldots, N; k = 1, 2, \ldots, M \\
& \quad (x_{ij,k} \cdot d_{ij} - \infty_{j,k} \cdot a_{ij}) \times (x_{ij,k} \cdot a_{ij} - \infty_{j,k} \cdot d_{ij}) > 0 \\
& \quad i, j, k = 1, 2, \ldots, N; j = 1, 2, \ldots, M; i \neq i', j \neq j'; \\
& \quad k = 1, 2, \ldots, G_j; k = 1, 2, \ldots, F_j \\
& \quad \sum_{j=1}^{N} x_{ij,k} = 1, \text{ for each } i \\
& \quad x_{ij,1} = x_{ij,1} = 1, \text{ if } x_{ij} = a_{ij} \\
& \quad x_{ij,k} = 0 \text{ or } x_{ij,k} = 1
\end{align*}
\]

(13)

5 | ARTIFICIAL BEE COLONY ALGORITHM FOR TRAIN RESCHEDULING AND TRACK ASSIGNMENT MODEL

5.1 | Classical artificial colony algorithm

The artificial bee colony algorithm (ABC) is an optimization algorithm inspired by the intelligent foraging behaviour of a bee swarm, which was put forward by Karaboga [61]. The ABC algorithm is now one of the study hotspots in evolutionary research fields and it has been used to solve many engineering optimization problems.

In the ABC algorithm, the position of the food source represents a possible solution. Whether the position of the food source is the best position or an acceptable position depends on the amount of the honey in the position. Similarly, the quality of the solution of a problem is determined by the value of the fitness function. The number of the bees is the number of possible solutions that are to be optimized.

The bees in the ABC model are divided into three groups: employed bees (EBs), onlooker bees (OBs) and scout bees (SBs). The EB will search for new positions of the honey and transit the information to the onlookers. The duty of the OB is to search for new positions of the honey source based on the information given by the employed bees, and the OB will then select the right source position. The SB will look for a new honey source position to replace the inferior honey source position. The main idea of the ABC algorithm is to simulate the process of the bees’ search for the best honey source position.

Onlookers watch the dances of employed bees and choose food sources depending on dances. The main steps of the algorithm are given below [61, 62].

1. Initial food sources are produced for all employed bees.
2. Repeat
   - Each employed bee goes to a food source in her memory and determines a neighbour source, then evaluates its nectar amount and dances in the hive.
   - Each onlooker bee watches the dance of the employed bees and chooses one of their sources depending on the dances, and then goes to that source. After choosing a neighbour around the source, she evaluates its nectar amount.
   - Abandoned food sources are determined and are replaced with the new food sources discovered by the scouts.
   - The best food source found so far is registered.
3. Until (requirements are met).

The first step of the algorithm is to generate \( C_p \) solutions, which correspond to the positions of the honey sources. \( C_p \) is the number of the bees. Each candidate solution \( \theta(l) = (l_1, \ldots, l_p) \) is a \( d \)-dimensional vector, where \( d \) is the number of the decision variables. Then, the EB, OB and SB will iterate the calculation process for \( \xi_{\text{max}} \) times, where \( \xi_{\text{max}} \) represents the allowed maximal iteration times. In the process, the EBs are designed to generate the new candidate solution based on the collected information in order to calculate the fitness value. If the fitness value calculated with the new solution is better than the fitness value calculated with the former solution, the new solution will take place of the former solution. When all the EBs finish searching, they will share the candidate solution and fitness value with the OBs. The OBs will evaluate the fitness value of the shared candidate solutions and select the honey resource according to the probability of the amount of the honey. In this case, the OBs will update the candidate solution based on the former candidate solutions and calculate the new fitness value if the fitness value of the new candidate solution is better than that of the former candidate solution; otherwise, we keep the former candidate solution. If the quality of honey of the source is not acceptable, the SB will replace it with a new randomly generated solution.

Set \( \theta(l) = \{\theta(l, q)\}_{l=1}^{C_p} \) to be the solution of a problem, where \( \theta(l, q) \) is generated according to Equation (14).

\[ \theta(l, q) = \theta_{\text{min}}(q) + \text{rand}[0, 1](\theta_{\text{max}}(q) - \theta_{\text{min}}(q)) \]  

(14)

In Equation (1), \( \theta_{\text{max}}(q) \) and \( \theta_{\text{min}}(q) \) are the maximum and minimum values of the \( q \)-th dimension of the solution.
24 | MENG ET AL.

5.2 Special instructions of algorithm design for train rescheduling and track assignment model

A key step of ABC is to design the original solution vector of the model. The number of the vector elements, the code of the element and other parameters should be determined firstly.

1. Number of t elements of a solution vector

According to the rule of ABC algorithm and the requirements of the model presented in this paper, the number of the elements of the solution elements can be calculated out due to Equation (17).

\[ U = 2NM + N \times \sum_{j=1}^{M} F_j \]  

When a train goes through a station, there are two times that should be decided—the arrival time and the departure time. If a train goes through a station without stopping, the arrival time equals to the departure time, while they may be different when rescheduling the trains. So we firstly designed 2NM elements for the solution vector. Then when a train goes through a station, it must occupy a track, no matter whether it stops. For there is \( F_j \) tracks in station \( j \), there are \( F_j \) options for each train when it going through the station. So we design another \( N \times \sum_{j=1}^{M} F_j \) elements for the solution vector. \( x_{i,j,k} \) is a binary variable. If train \( i \) occupied track \( k \) at station \( j \), it is 1, otherwise it is 0.

1. Code of elements of solution vector

It is hard to describe the arrival time and departure time directly due to the time format. So we changed the time to the number of the seconds between the time and the 0 o’clock. For example, 08:10:30 can be changed to 29430. Then we can carry out the calculation process easily by turning the time into this format.

And \( x_{i,j,k} \) is a binary variable. It is 0 or 1. When we designed the computer program to find the solution, we generated a random number and determine whether it is 0 or 1 according to the value of the generated random number. When the random number was larger than a threshold value \( \delta \), \( x_{i,j,k} \) was set to be 1. Otherwise, it is set to be 0. In this paper, we set \( \delta = 0.8 \).

1. Procedure for calculation

A. To set \( \xi_{\text{max}} \) and \( \xi_{\text{limit}} \); set the scale of the artificial bee \( C_p \). Set the weight parameters \( \alpha \) and \( \omega_j \).
B. To initialize solution, generating three kinds of decision variables, \( a_{i,j} \), \( d_{i,j} \) and \( x_{i,j,k} \), according to Equation (17) and the coding rule analysed in Section 5.2.
C. Set the iteration time \( b = 1 \);
D. Calculate the fitness value according to Equations (10)–(12).
E. Decide the new solution according to the computing results of step (D);
F. Calculate the probability of the \( C_p \) solution according to Equation (16);
G. Decide which solution should be updated according to the calculated probability of step (F);
H. Update the solution and calculate the fitness value according to Equation (15);
I. Decide the new solution according to the computing results of step (H);
J. replace the solution which cannot be updated for \( \xi_{\text{limit}} \) times with a solution generated with Equation (14);
K. Select the optimal solution due to the above computing results, comparing the fitness values;

\[ \eta(l, q) = \Theta(l, q) + \varphi \eta(l, q) - \Theta(\sigma, q) \]  

where \( \xi(l) \) is the fitness of the nectar source of \( l \).

The calculation process is as follows.

Initialization: to set \( \xi_{\text{max}} \) and \( \xi_{\text{limit}} \), to generate the solution \( \Theta = [\theta(l, q)]_{l \times Q} \) to be optimized according to Equation (14);

Set the iteration time \( b = 1 \);

While \( (b \leq \xi_{\text{max}}) \) or the fitness value has not reach the expected value

Optimization phase of the EB

Update the solution and calculate the fitness value according to Equation (15);

Decide the new solution of this phase with a greedy algorithm based on the fitness value;

Calculate the probability of the \( C_p \) solution according to Equation (16);

Optimization phase of the OB

Decide which solution should be updated according to the calculated probability;

Update the solution and calculate the fitness value according to Equation (15);

Decide the new solution of this phase with a greedy algorithm based on the fitness value;

Optimization phase of the SB

Replace the solution which cannot be updated for \( \xi_{\text{limit}} \) times with a solution generated with Equation (14);

Select the optimal solution up to now;

Set \( b = b+1 \);

End

Determine the global optimal solution.

\[ \xi(l) = C \sum_{l=1}^{P} \xi(l) \]  

where \( C \) is the number of the element and other parameters should be determined firstly.
FIGURE 5  Original operation diagram between Nanjingnan to Shanghaihongqiao from 12:00 PM to 3:00 PM in the south-going direction

TABLE 4  Minimum running time of the two types of trains in each section

| Section              | Minimum running time of g trains | Adjusted minimum running time of g trains | Minimum running time of d trains | Adjusted minimum running time of d trains |
|----------------------|----------------------------------|------------------------------------------|---------------------------------|------------------------------------------|
| Nanjingnan-Zhenjiangnan | 11.09                            | 11.30                                     | 15.36                           | 16.00                                     |
| Zhenjiangnan-Danyangbei | 5.29                             | 5.30                                      | 7.41                            | 8.00                                      |
| Danyangbei-Changzhoubei | 5.39                             | 6.00                                      | 7.55                            | 8.00                                      |
| Changzhoubei-Wuxidong  | 9.46                             | 10.00                                     | 13.41                           | 12.00                                     |
| Wuxidong-Suzhoubei     | 4.38                             | 5.00                                      | 6.29                            | 6.30                                      |
| Suzhoubei-Kunshannan   | 5.19                             | 6.00                                      | 7.26                            | 7.30                                      |
| Kunshannan-Shanghaihongqiao | 8.34                           | 9.00                                      | 12.00                           | 12.00                                     |

L. Set $b = b + 1$; if $b < \xi_{\text{max}}$, go to step (D), else, go to step (M).
M. Turn the computing results into timetable and the track usage plan.

6  COMPUTING CASE AND RESULTS ANALYSIS

We ran the data experiments on a computer with 8G memory and i5-2400 CPU and took Matlab R2014a as the computing platform. We took the train timetable data of Beijing-Shanghai as the base data of the computing case. The selected section is from the Nanjingnan station to the Shanghaihongqiao station. The train operation diagram is showed in Figure 5. The related trains are divided into two groups. Train numbers in one group start with “D”, while others start with “G”. The minimum running time of the two types of trains in each section are listed in Table 4 and the number of tracks available in each station in this computing case is listed in Table 5.

We then assumed that G1809 was late by 20 min when it left Nanjingnan station for Shanghaihongqiao, G1, G107, and G113 were late by 10 min, G221 was late by 15 min.

According to the model and the algorithm designed in this paper, we instantiated the model and calculated the results. The arrival time and the departure time, as well as the order number that the trains occupied at the stations were seen as the decision variables in the model. The arrival and departure time were transformed to the number of the seconds from 0:00 to the relative time. Then we formed a solution vector with the arrival and departure time of the trains at the stations. There were 34 trains and 8 stations in the computing case. According to Table 5, there were a total of 21 tracks in the south-going direction for the railway section, except for the tracks in
TABLE 5  Number of available tracks in each station in this computing case

| Section         | Total number of tracks | Number of tracks for north-going trains/main tracks | Number of tracks for south-going trains/main tracks |
|-----------------|------------------------|----------------------------------------------------|---------------------------------------------------|
| Nanjingnan      | 8                      | 4/1                                                | 4/1                                               |
| Zhenjiangnan    | 6                      | 3/1                                                | 3/1                                               |
| Danyangbei      | 4                      | 2/1                                                | 2/1                                               |
| Changzhoubi     | 6                      | 3/1                                                | 3/1                                               |
| Wuxidong        | 6                      | 3/1                                                | 3/1                                               |
| Suzhoubei       | 6                      | 3/1                                                | 3/1                                               |
| Kunshan         | 6                      | 3/1                                                | 3/1                                               |

Shanghaihongqiao station. To be precise, we calculated the track assignment plan for the trains at Nanjingnan, Zhenjiangnan, Danyangbei, Changzhoubi, Wuxidong, Suzhoubei, Kunshan, not at Shanghaihongqiao. Thus, the number of elements in the solution vector was $34 \times 8 \times 2 + 34 \times 21 = 1258$. We set the scale of the artificial bee $C_p$ to be 20. $\alpha$ was set to be 100. $\omega_D$ is set to be 1 and $\omega_G$ is set to be 2.

The adjusted operation diagram is shown in Figure 6 and the rescheduled timetable is shown in Table 6. The usage plan of the arrival–departure tracks is shown in Table 7.

Comparing the data of Figure 5 and Table 6 (or Figure 6), we can see that the total delay time of all the trains at the stations is 862.5 min, which is the first part of the optimization goal in Equation (10). We can also calculate the optimization goal, $Z_2 = 0.5$ and $Z_3 = 1$. Thus, the total optimization goal is $Z = 862.5 + 100(0.5 + 1) = 1012.5$. The optimization goal is set to be a limited value that we can accept.

Among the delayed trains, only G1809 recovered the status of running due to the original operation diagram. G1 was late by 12 min when it arrived at Shanghaihongqiao. The main reason was for this was that G1 changed the overtaking station when it overtook G1809. It was planned that G1 should overtake G1809 at Danyangbei. However, it overtook G1809 at Changzhoubi, according to the computing results, because of the constraints in the model. G1809 was late by 8 min at Changzhoubi, which led to the delay of G1 at the following stations. G107 was late by 8 min and 30 s when it arrived at Shanghaihongqiao, which reduced the delayed time by more than one minute. G221 was late by 18 min when it reached Shanghaihongqiao, which only shortened the delay time by 3 min. We can see that the delay time cannot be shortened anymore because of the minimum running time in the sections. G113 was still late by 10 min as it departed from Nanjingnan. G359 was not late when it left Nanjingnan, but it was late by 1 min when it arrived at Shanghaihongqiao; the delay was affected by G113. Similarly, G113 was late at Danyangbei and Changzhoubi because it was affected by G13, although it got rid of the delay when it arrived at Wuxidong.

The computing results gave us not only the adjusted timetable, but also the track usage plan of the stations, which we can see from Tables 6 and 7.

From Table 7, we can see that each side tracks (track 3, track 5 and track 7) of Nanjingnan station was used for seven times, reaching a perfect usage equilibrium. Track 3 and track 5 of Zhenjiangnan station were used five times each. We can see that the same result occurs at Changzhoubi station. At Wuxidong station, track 3 was used seven times and track 5 was used six times.

FIGURE 6  Adjusted operation diagram between Nanjingnan to Shanghaihongqiao from 12:00 PM to 3:00 PM in the south-going direction
### TABLE 6  Re-scheduled timetable between Nanjingnan to Shanghaihongqiao from 12:00 PM to 3:00 PM in the south-going direction of the related trains

| Track | Nanjingnan | Zhenjiangnan | Danyangbei | Changzhoubei | Wuxidong | Suzhoubei | Kunshannan | Shanghaihongqiao | Nanjingnan | Zhenjiangnan | Danyangbei | Changzhoubei | Wuxidong | Suzhoubei | Kunshannan | Shanghaihongqiao |
|-------|------------|--------------|------------|--------------|----------|-----------|------------|-----------------|------------|--------------|------------|--------------|----------|-----------|------------|----------------|
| G1805 | 11:4400    | 12:0500      | 12:2800    | 12:3700      | 12:5600  | 13:1920   | 13:2100    | 13:4600        | 12.3500   | 13.0630     | 13.1600    | 13.2000      | 13.3400  | 13.5000   | 14.0100    | 14.1700        |
|       | 11:4600    | 12:1900      | 12:2800    | 12:4000      | 13:0400  | 13:0530   | 13:4100    | 13:5400        | 12.5200   | 13.0630     | 13.1600    | 13.2000      | 13.3400  | 13.5000   | 14.0100    | 14.1700        |
|       | 11:4100    | 12:2500      | 12:3500    | 12:4400      | 13:1000  | 13:3200   | 13:4100    | 13:4000        | 12.0500   | 13.3400     | 13.4200    | 13.5000      | 13.5000  | 13.5000   | 14.0700    | 14.1800        |
|       | 12.0700    | 12.4630      | 12.5500    | 12.4400      | 13.2200  | 13.3500   | 13.5230    | 13.4100        | 12.3000   | 13.3700     | 13.4200    | 13.5000      | 13.5300  | 13.5300   | 14.0700    | 14.1800        |
|       | 12.3000    | 12.4930      | 12.5700    | 12.4400      | 13.2700  | 13.4100   | 13.5230    | 13.4100        | 12.3500   | 13.3700     | 13.4200    | 13.5000      | 13.5300  | 13.5300   | 14.0700    | 14.1800        |
|       | 12.2300    | 12.4930      | 13.0000    | 12.4400      | 13.2300  | 13.4100   | 13.5230    | 13.4100        | 12.2800   | 13.3700     | 13.4200    | 13.5000      | 13.5300  | 13.5300   | 14.0700    | 14.1800        |
|       | 12.2800    | 12.5300      | 13.0900    | 12.4400      | 13.2300  | 13.4100   | 13.5230    | 13.4100        | 12.3000   | 13.3700     | 13.4200    | 13.5000      | 13.5300  | 13.5300   | 14.0700    | 14.1800        |
|       | 12.2400    | 12.5300      | 13.1500    | 12.4400      | 13.2300  | 13.4100   | 13.5230    | 13.4100        | 12.2800   | 13.3700     | 13.4200    | 13.5000      | 13.5300  | 13.5300   | 14.0700    | 14.1800        |

Note: the arrival and departure time written in italic in Table 6 are the adjusted time.

### TABLE 7  Usage plan of each track for the different stations

| Station       | Track | Trains that occupied the track |
|---------------|-------|--------------------------------|
| Nanjingnan    | 1     | G17, G219                      |
| Nanjingnan    | 3     | G1735, G107, D3051, G13, G115, G117, G213 |
| Nanjingnan    | 5     | G1809, G221, G113, G41, G1813, G119, G15 |
| Nanjingnan    | 7     | G1, G111, G211, G359, D3085, G2811, G229 |
| Zhenjiangnan  | 1     | G11, G105, G1911, G1, G107, G17, G221, G111, G211, G13, G13, G359, G117, G219, G213 |
| Zhenjiangnan  | 3     | G1805, G1809, G41, G1813, G119 |
| Zhenjiangnan  | 5     | G1735, D3051, G115, G1813, G2811 |
| Danyangbei    | 1     | G11, G105, G1911, G1805, G1735, G1, G107, G17, G221, G211, G13, G13, D3051, G359, G41, G115, G1813, D3085, G219, G119 |
| Danyangbei    | 3     | G1809, G111, G117 |
| Changzhoubei  | 1     | D3096, G11, G1, G107, G17, G221, G111, G13, G13, G359, G41, G1813, D3085, G117, G219 |
| Changzhoubei  | 3     | G7595, G1911, G1735, G221, D3051 |
| Changzhoubei  | 5     | G105, G1805, G1809, G113, G115 |
| Wuxidong      | 1     | G101, G11, G105, G1, G107, G17, G13, G13, G359, G41, G115, G117 |
| Wuxidong      | 3     | G7075, D3096, G1805, G1809, G111, D3051, D3085 |
| Wuxidong      | 5     | G7595, G1911, G1735, G221, G211, G1813 |
| Suzhoubei     | 1     | G7545, G7075, G105, G1911, G1805, G1, G17, G221, G111, G13, G359, G41, G115 |
| Suzhoubei     | 3     | G101, G11, G1735, G107, G113, G1813 |
| Suzhoubei     | 5     | G7595, D3096, G1809, G211, D3051 |
| Kunshannan    | 1     | G1801, G7545, G101, G7595, G11, G105, G1911, G1, G1735, G107, G1809, G17, G221, G111, G13, G211, G113, G359, G41 |
| Kunshannan    | 3     | G7075, G1805 |
| Kunshannan    | 5     | D3096, G115 |
times. Track 3 was used six times and track 5 was used five times at Suzhoubei station. Both of the side tracks (track 5 and track 7) were used twice at Kunshannan station. We can see that the usage equilibrium reached a high level. The detailed data is listed in Table 8.

From the computing results, we can see that the model is valid and feasible in solving the problem. We got the acceptable value of the optimization goal and all of the constraints were accepted, which can be proved in the adjusted operation diagram in Figure 6. In addition, the satisfying result is also related to the well-designed algorithm, eliminating the unreasonable result vectors in time to assure the feasibility.

### 7 CONCLUSION

In this paper we first constructed an integrated model for train rescheduling and track and assignment, trying to avoid the possibility that we cannot generate an available track usage plan to adapt to the rescheduled train timetable, while we reschedule the timetable and design the track assignment plan separately. The model described the train rescheduling and track assignment problem accurately. And we successfully applied the artificial bee colony algorithm to solve the model to generate the integrated train timetable and track usage plan.

It is worth mentioning that the track occupation constraint is designed exquisitely in the integrated model for train rescheduling and track assignment. We thoroughly analysed all possible time and space relationship of the neighboured trains at the stations, and derived the constraint formula. It precisely described the safety requirement when the neighboured trains ran across the stations. It assured the generated track usage plan adapt to the rescheduled timetable, which was the key point of the model.

The algorithm is designed to solve the train rescheduling and track assignment problem. We fully utilized the excellent computing performance of ABC algorithm, specifying the parameters carefully to adapt it to the model. When the computing process was carried out, the arrival and departure time of the trains at stations were changed to integers, which assure that they were easy to deal with. The algorithm was capable of handling the constraints typically encountered in real-world train rescheduling and track assignment and, can be easily adapted to accommodate possible further characteristics. Furthermore, a very general hierarchic objective function allows to optimize different aspects of the problem as the on-time rate of all the trains, the passenger satisfaction degree, the average delay and so on with respect to the nominal timetable.

The case computing has been performed by using real-world instances from Beijing–Shanghai high speed railway. The main aim of the testing was to test the availability of the model and the feasibility of the algorithm. The obtained results show that the integrated train rescheduling and track assignment model is available, and the proposed algorithm is consistently capable of resolving track usage conflicts and obtain high quality rescheduled timetable, at the same time, the generated track usage plan assured the feasibility of the train timetable. Although there are many publications on train rescheduling problem and the track assignment problem, we believe that the main contribution of our approach is to solve the train rescheduling and track assignment problem in an integrated manner.

In future works we will continue to integrated the railway operation work. Train rescheduling belongs to the management level, while track assignment and train traction planning belong to the control level. We will try to integrate the three problems, train rescheduling, track assignment and train traction planning, to make contributions to the management and control integration of railway operation.

### ACKNOWLEDGEMENTS

This work is supported by the National Natural Science Foundation of China (Grant No. 71861022), the Young Teachers Program of Lanzhou Jiaotong University (Grant: 2019039), the Foundation of a Hundred Young Talents Training Program of Lanzhou Jiaotong University (Grant No. 1520220210), China National Key Research and Development Project (Grant 2016yfb1200100) and Key Laboratory of Urban Rail Transit Intelligent Operation and Maintenance Technology and Equipment of Zhejiang Province (Grant No. ZSDRTKF2020005). The authors wish to thank the anonymous reviewers and the editor for their comments and suggestions. The authors would like to thank LetPub (www.letpub.com) for providing linguistic assistance during the preparation of this manuscript.

### ORCID

Xuelei Meng https://orcid.org/0000-0002-4243-3125

### REFERENCES

1. Caprara, A., et al.: A language heuristic algorithm for a real-world train timetabling problem. Discrete Appl. Math. 154(5), 738–753 (2006)
2. Tornquist, J., Persson, J.A.: N-Tracked railway traffic re-scheduling during disturbance. Transp. Res. B 41(3), 342–362 (2007)
3. Abril, M., et al.: Distributed search in railway scheduling problems. Eng. Artif. Intell. 21(5), 744–755 (2008)
4. Castillo, E., et al.: Timetabling optimization of a mixed double-and single-tracked railway network. Appl. Math. Model. 35(2), 859–878 (2011)
5. Corman, F., et al.: Optimal inter-area coordination of train rescheduling decisions. Transp. Res. E 48(1), 58–81 (2012)

### TABLE 8 Occupied time of each track for the different stations

| Station     | Track 1 | Track 3 | Track 5 | Track 7 |
|-------------|---------|---------|---------|---------|
| Nanjingnan  | 2       | 7       | 7       | 7       |
| Zhenjiangnan| 15      | 5       | 5       | –       |
| Danyangbei  | 20      | 3       | –       | –       |
| Changzoubei | 14      | 5       | 5       | –       |
| Wuxidong    | 12      | 7       | 6       | –       |
| Suzhoubei   | 13      | 6       | 5       | –       |
| Kunshannan  | 19      | 2       | 2       | –       |
6. Min, Y.H., et al.: An appraisal of a column-generation-based algorithm for centralized train-conflict resolution on a metropolitan railway network. Transp. Res. B 45(2), 409–429 (2011).

7. Kang, I., et al.: A practical model for last train rescheduling with train delay in urban railway transit networks. Omega 50, 29–42 (2016).

8. Wang, P., et al.: Rescheduling trains using Petri Nets and heuristic search. IEEE Trans. Intell. Transp. Syst. 17(3), 726–735 (2016).

9. Šemrov, D., et al.: Reinforcement learning approach for train rescheduling on a single-track railway. Transp. Res. B 86(1), 250–267 (2016).

10. Yang, X., et al.: An optimisation method for train scheduling with minimum energy consumption and travel time in metro rail systems. Transportmetrics B: Transp. Dyn. 3(2), 79–98 (2015).

11. Lan, L.W., Lin, E.T.J.: Measuring railway performance with adjustment of environmental effects, data noise and slacks. Transportmetrics 1(2), 161–189 (2005).

12. Yang, X., et al.: An energy-efficient rescheduling approach under delay perturbations for metro systems. Transportmetrics B: Transp. Dyn. 6(2), 1–15 (2018).

13. Chen, D., et al.: Optimizing the draft passenger train timetable based on node importance in a railway network. Transp. Lett. 11(1), 20–32 (2019).

14. Caimi, G., et al.: A model predictive control approach for discrete-time rescheduling in complex central railway station areas. Comput. Oper. Res. 39(11), 2578–2593 (2012).

15. Corman, F.: Interactions and equilibrium between rescheduling train traffic and routing passengers in microscopic delay management: A game theoretical study. Transp. Sci. 54(3), 785–822 (2020).

16. Zhu, Y., Goverde, R.M.P.: Dynamic and robust timetable rescheduling for uncertain railway disruptions. J. Rail Transp. Plann. Manage. 15, 100196, In Press, https://doi.org/10.1016/j.jrtpm.2020.100196 (2020).

17. Zinner, Y., et al.: Rescheduling traffic on a partially blocked segment of a railway with a siding. Autom. Remote Control 81(6), 955–966 (2020).

18. Zhu, Y., Goverde, R.: Railway timetable rescheduling with flexible stopping and flexible short-turning during disruptions. Transp. Res. B 123, 149–181 (2019).

19. Feng, Z., et al.: Train delay propagation under random interference on high-speed rail network. Int. J. Mod. Phys. C 30(8), 1950059 (2019).

20. Liu, J., et al.: Algorithm and peer-to-peer negotiation strategies for train dispatching problems in railway bottleneck sections. IET Intel. Transp. Syst. 13(11), 1717–1725 (2019).

21. Jiang, C., et al.: Forecasting primary delay recovery of high-speed railway using multiple linear regression, supporting vector machine, artificial neural network, and random forest regression. Can. J. Civ. Eng. 46(5), 353–363 (2019).

22. Katori, T., Izumi, T.: Train rescheduling plan generation based on train route combination considering rolling stock types. Int. J. Transp. Dev. Integr. 3(2), 117–131 (2019).

23. Josyula, S., et al.: A parallel algorithm for train rescheduling. Transp. Res. C 95, 545–569 (2018).

24. Martins, V.D.: Feedforward tactical optimization for energy-efficient operation of freight trains: the Swiss case. Transp. Res. Rec. 2672(10), 278–288 (2018).

25. Iswandani, K., Rusdiansyah, A.: Development model of train rescheduling model consider predictive delay. MATEC Web of Conf. 204, 07004 (2018).

26. Kato, S., et al.: Extraction algorithm for unpredictable delays in train traffic disruption using train operation simulation. IEEE Trans. Ind. Appl. 138(2), 76–83 (2018).

27. Ortega, F.A., et al.: Determining routes for an auxiliary fleet in presence of emergencies on a railway line. Inf. Syst., Logist., Supply Chain 262, 3–13 (2018).

28. Shakibaifar, M., et al.: An intelligent simulation platform for train traffic control under disturbance. Int. J. Model. Simul. 39(3), 135–156 (2019).

29. Sato, K., Fukumura, N.: Real-time freight locomotive rescheduling and uncovered train detection during disruption. Eur. J. Oper. Res. 221(3), 636–648 (2012).

30. Espinosa-Aranda, J.L., García-Rodríguez, R.: A demand-based weighted train delay approach for rescheduling railway networks in real time. J. Rail Transp. Plann. Manage. 3(1–2), 1–13 (2013).

31. Zhan, S., et al.: Real-time high-speed train rescheduling in case of a complete blockage. Transp. Res. B 78, 182–201 (2015).

32. Chen, I., et al.: Modeling and solving real-time train rescheduling problems in railway bottleneck sections. IEEE Trans. Intell. Transp. Syst. 16(4), 1896–1904 (2015).

33. Chen, D., et al.: Genetic algorithm based on conflict detecting for solving departure time domains of passenger trains. Transp. Lett. 7(4), 181–187 (2015).

34. Dalapati, P., et al.: Real-time collision handling in railway transport network: an agent-based modeling and simulation approach. Transp. Lett. 11(8), 458–468 (2017).

35. Bettinelli, A., et al.: A real-time conflict solution algorithm for the train rescheduling problem. Transp. Res. B: Meth. 106, 237–265 (2017).

36. Carone, G., et al.: Railway disruption: a bi-level rescheduling algorithm. In: 2019 6th International Conference on Control, Decision and Information Technologies (CoDIT). IEEE, Paris, pp. 54–59 September 2, (2019) https://doi.org/10.1109/CoDIT.2019.8820380.

37. Ikonen, T.J., Harjunkoski, I: Decision-making of online rescheduling procedures using neuroevolution of augmenting topologies. Comput. Aided Chem. Eng. 46, 1177–1182 (2019).

38. Shakibaifar, M., et al.: A multi-objective decision support system for real-time train rescheduling. IEEE Intell. Transp. Syst. Mag. 10(3), 94–109 (2018).

39. Ortega, F., et al.: On-line timetable rescheduling in a transit line. Transp. Sci. 52(5), 1106–1121 (2018).

40. Galapitage, A., et al.: Optimal real-time junction scheduling for trains with connected driver advice systems. J. Rail Transp. Plann. Manage. 8(1), 29–41 (2018).

41. Corman, F., et al.: Automated real-time railway traffic control: an experimental analysis of reliability, resilience and robustness. Transp. Plann. Technol. 41(4), 421–447 (2018).

42. Yang, G., et al.: A real-time timetable rescheduling method for metro system energy optimization under dwell-time disturbances. J. Adv. Transp. 2019, 1 (2019).

43. Altazinah, E., et al.: A multi-objective optimization-simulation approach for real time rescheduling in dense railway systems. Eur. J. Oper. Res. 286(2), 662–672 (2020).

44. Huang, Y., et al.: Coupling time-indexed and big-M formulations for real-time train scheduling during metro service disruptions. Transp. Res. Part B: Methodol. 133, 38–61 (2020).

45. Zwanenveld, P.J., et al.: Routing trains through railway stations: model formulation and algorithms. Transp. Sci. 30(3), 181–194 (1996).

46. Kroon, L.G., et al.: Routing trains through railway stations: model formulation and algorithms. Transp. Sci. 30(3), 181–194 (1996).

47. Bettinelli, A., et al.: A real-time conflict solution algorithm for the train rescheduling problem. Transp. Res. B: Meth. 106, 237–265 (2017).

48. Corman, F., et al.: A tabu search algorithm for rerouting trains during rail traffic control. Transp. Res. B: Meth. 106, 237–265 (2017).

49. Billionier, A.: Using integer programming to solve the train-platforming problem. Transp. Sci. 37(2), 213–222 (2003).

50. Corman, F., et al.: A tabu search algorithm for rerouting trains during rail operations. Transp. Res. B 44(1), 175–192 (2010).

51. Schafroth, B.B.W., et al.: A dynamic model for real-time track assignment at railway yards. J. Rail Transp. Plann. Manage. 14, 100198 (2020).

52. König, F., Schöna, C.: Railway delay management with passenger rerouting considering train capacity constraints. Eur. J. Oper. Res. 2020, 5609524 (2020).

53. Vansteenwegen, P., et al.: A conflict prevention strategy for large and complex networks in real-time railway traffic management. Linköping Electron. Conf. Proc. 60, 1099–1116 (2019).

54. Zhang, Y., et al.: A fast approach for reoptimization of railway train platforming in case of train delays. J. Adv. Transp. 288, 450–465, In Press. https://doi.org/10.1016/j.jrtpm.2020.055 (2020).

55. Carey, M., Carville, S.: Scheduling and platforming trains at busy complex stations. Transp. Res. A 37(3), 195–224 (2003).

56. Lee, Y., Chen, C.Y.: A heuristic for the train pathing and timetabling problem. Transp. Res. B 43(8–9), 837–851 (2009).
55. Meng, L., Zhou, X.: Simultaneous train rerouting and rescheduling on an N-track network: A model reformulation with network-based cumulative flow variables. Transp. Res. B 67(3), 208–234 (2014)
56. Samà, M., et al.: A variable neighbourhood search for fast train scheduling and routing during disturbed railway traffic situations. Comput. Oper. Res. 78, 480–499 (2017)
57. Xu, P., et al.: A train rescheduling model integrating speed management during disruptions of high-speed traffic under a quasi-moving block system. Transp. Res. B 104(1), 638–666 (2017)
58. Chang, Y., et al.: Integrated scheduling of handling operations in railway container terminal. Transp. Lett. 11(7), 402–412 (2017)
59. Luan, X., et al.: Integration of real-time traffic management and train control for rail networks - part 1: optimization problems and solution approaches. Transp. Res. B 115, 41–71 (2018)
60. Pellegrini, P., et al.: Efficient train re-routing and rescheduling: valid inequalities and reformulation of RECIFE-MILP. Transp. Res. B 120, 33–48 (2019)
61. Karaboga, D.: An idea based on honey bee swarm for numerical optimization. Technical Report TR06, Erciyes University, Kayseri, Turkey, October (2005)
62. Xiang, W., et al.: An improved artificial bee colony algorithm based on the gravity model. Inf. Sci. 429, 49–71 (2018)

How to cite this article: Meng X, Wang Y, Xiang W, Jia L. An integrated model for train rescheduling and station track assignment. IET Intell Transp Syst. 2021;15:17–30. https://doi.org/10.1049/itr2.12001