Transportation Research Part B 91 (2016) 178–210

Contents lists available at ScienceDirect

Transportation Research Part B

journal homepage: www.elsevier.com/locate/trb

# Energy-efficient metro train rescheduling with uncertain time-variant passenger demands: An approximate dynamic programming approach

Jiateng Yin[a], Tao Tang[a], Lixing Yang[a,*], Ziyou Gao[a], Bin Ran[b]

[a] State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing, 100044, China
[b] Department of Civil and Environmental Engineering, University of Wisconsin-Madison, Madison, WI, 53706, USA

A R T I C L E  I N F O

*Article history:*
Received 6 December 2015
Revised 16 May 2016
Accepted 17 May 2016
Available online 2 June 2016

*Keywords:*
Metro train rescheduling
Passenger delay
Energy efficiency
Approximate dynamic programming
Stochastic programming

A B S T R A C T

In a heavily congested metro line, unexpected disturbances often occur to cause the delay of the traveling passengers, infeasibility of the current timetable and reduction of the operational efficiency. Due to the uncertain and dynamic characteristics of passenger demands, the commonly used method to recover from disturbances in practice is to change the timetable and rolling stock manually based on the experiences and professional judgements. In this paper, we develop a stochastic programming model for metro train rescheduling problem in order to jointly reduce the time delay of affected passengers, their total traveling time and operational costs of trains. To capture the complexity of passenger traveling characteristics, the arriving ratio of passengers at each station is modeled as a non-homogeneous poisson distribution, in which the intensity function is treated as time-varying origin-to-destination passenger demand matrices. By considering the number of on-board passengers, the total energy usage is modeled as the difference between the tractive energy consumption and the regenerative energy. Then, we design an approximate dynamic programming based algorithm to solve the proposed model, which can obtain a high-quality solution in a short time. Finally, numerical examples with real-world data sets are implemented to verify the effectiveness and robustness of the proposed approaches.



## 1. Introduction

Metro traffic plays an important role in urban economic and social developments for big cities (e.g., Beijing, London, New York), since it is regarded as an environmentally friendly transportation mode with high capacity, good punctuality and low energy-consumption (Xu et al., 2016a; Yang et al., 2016). With the expansion of residents in large cities (e.g., in Beijing), metro operational companies are usually undertaking great pressures in order to transport large amount of passengers under the infrastructure limitations. For example, in order to increase transport capacity in peak-hours, the headway time is compressed within only 2 min for Beijing metro Line 1 and Line 2. Accordingly, the operated trains have to follow the arrival and departure times given in the tight timetable with a high service frequency.

In reality, the real-time operations of a metro system are always disrupted by some unexpected disturbances, which are usually caused by train arrival time delays, departure time delays, and unplanned stops due to unsteady driving behaviors,

* Corresponding author.
*E-mail address:* lxyang@bjtu.edu.cn (L. Yang).

http://dx.doi.org/10.1016/j.trb.2016.05.009

passenger demand variations, infrastructure failures, signal errors, etc. Once these disturbances last for several minutes, they may reduce the service quality greatly and even cause serious interferences in peak hours. For example, in 2010, a total of 126 times of drivers' pulling the emergency handles occurred in the Helsinki Metro system, most of which delayed the trains for about 3–5 min and affected the passengers' traveling plan in different degrees (Karvonen et al., 2011). This situation seems to be even worse in a heavily congested metro line, where the first delayed train will affect the following trains sequentially, resulting in the infeasibility of the original timetable, longer traveling time of passengers, and even the disruption of a whole block. In such cases, the metro dispatchers are needed to change the timetable to recover from an incident to a feasible situation (Cacchiani et al., 2014) as soon as possible, which is called a real-time train rescheduling (TTR) problem.

Although some researches in literature (Cacchiani et al., 2014; Cacchiani and Toth, 2012; Corman et al., 2012; Dundar and Sahin, 2013; Li et al., 2016; Veelenturf et al., 2015, etc.) begin to stress this problem, most metro dispatchers presently still take these decisions manually on the basis of their experiences and professional judgments, which is typically lack of rigorous computation and optimization. In general, the difficulty of this problem is caused by the following reasons. (1) The real-time metro train operations are subject to the influencing factors caused by the complex passenger characteristics. Different from railway systems, the passenger flow in urban rail transit has a different characteristic, since passengers usually do not care about the train timetables before trips, leading to the time-variant and uncertain features (Yang et al., 2015). It is realistically difficult to formulate a feasible decision-making model that simultaneously takes these factors into account. (2) The headway time in a congested metro line is only 2–3 min. Then, a rescheduled timetable should be acquired in a short computational time, which disenables the practical applications of most existing optimization approaches. With these concerns, it is practically significant to propose a unified approach for quickly solving the rescheduling problem with the high-quality solutions. This study will formally address this issue.

### *1.1. Literature review*

The train scheduling/rescheduling problem, which aims to obtain a timetable that is carried out by the trains in the operations, has attracted tremendous attention over the last decades (Huang et al., 2016; Niu and Zhou, 2013; Niu et al., 2015b; Wong et al., 2008; Xu et al., 2015; Yang et al., 2015; Zhou and Zhong, 2007). In general, the train scheduling problem is usually formulated by means of some mathematical models, e.g., a mixed integer programming (MIP) model (Wong et al., 2008; Zhou and Zhong, 2007), an alternative graph model (Cacchiani and Toth, 2012) and a nonlinear inter programming model (Niu and Zhou, 2013; Niu et al., 2015b), in which a schedule is optimized with various objectives functions, such as the total travel time (Zhou and Zhong, 2007), transfer waiting time (Wong et al., 2008), energy consumption (Li and Lo, 2014), etc. Meanwhile, the solution methodologies can be summarized into three categories, namely, (1) optimization approaches which use some mathematical optimization softwares (e.g, GAMS (Yang et al., 2013a)) and accurate algorithm (e.g., branch and bound algorithms (Zhou and Zhong, 2007)), (2) heuristic methods (Lee and Chen, 2009) (e.g., Lagrangian Relaxation (Caprara et al., 2002)) and (3) simulation methods (e.g., Dorfman and Medanic (2004); Mu and Dessouky (2013); Xu et al. (2015) ). In addition, we can refer to Cacchiani et al. (2014) for the surveys that focus on this railway timetabling problem.

In practice, the railway operations might be often subject to a variety of disturbances (Karvonen et al., 2011), leading to the ineffectiveness and infeasibility of the original schedules. Due to the rising concerns on the service level and operation efficiency of a railway system, the disturbance management has currently become an active research area in operations research and railway operations (Cacchiani et al., 2014). Different from the train timetabling problem, the railway train rescheduling problem for disturbance management mainly focuses on reducing the delay of trains or the delay of passengers/freights, in order to regenerate a conflict-free timetable in case of unexpected disturbances in a railway network. For example, D'Ariano et al. (2007) treated the rescheduling problem as a huge job scheduling problem with no-store constraints, and they proposed a detailed alternative graph model for decision-makings of dispatchers by utilizing the accurate real-time information of train positions and speeds. Then, a real-time traffic management system, i.e., ROMA (railway traffic optimization by means of alternative graph), was developed, which can be applied to rail systems with heavy disturbances to solve expected route conflicts and increase the punctuality (D'Ariano, 2009). Corman et al. (2012) proposed a bilevel programming model for regional control centers to coordinate the dispatchers' work. In this model, the variables are the border constraints and the objective is to minimize the deviation between the real-time traffic and the off-line timetable. Louwerse and Huisman (2014) formulated integer programming models for adjusting a railway timetable in case of disturbance that all tracks of a segment are blocked and no trains can be operated in this segment. The objective aims to make a trade-off between canceling and delaying trains to maximize the service level.

In case of a sudden disruption, it is practically desirable to regenerate the optimal or near-optimal alternative timetable rapidly so as to recover the disturbance as soon as possible. Then, the computational speed is usually recognized as one of the key indictors to evaluate the rescheduling approaches. Along this line, Xu et al. (2016a) proposed the idea that the delayed trains are allowed to be rescheduled by using tracks in the opposite direction through crossover tracks to generate an optimal rescheduled plan that minimizes the total train delay, and an efficient train rescheduling strategy (ETRS) was developed to generate high-quality solutions within a fairly short computational time (at millisecond level). In order to solve the rescheduling problem on an *N*-track rail network, Meng and Zhou (2014) developed a time-space network modelling framework, and the original problem was decomposed into a sequence of single train optimization problems, which was solved by a standard label correcting algorithm. Aiming to find good solutions quickly, Tornquist (2012) developed a greedy algorithm based on an optimization-based approach for train rescheduling that could reduce the train delays. Dundar and Sahin (2013) proposed a genetic algorithm (GA) and artificial neural network (ANN) for train rescheduling, where GA was applied to reschedule trains by minimizing total delays in conflict resolutions, and the ANN was trained by actual data to mimic the decision behavior of train dispatchers. Then, two approaches were compared and evaluated, which illustrated that GA was able to find good solutions for small-size problems in short time. Besides, Corman and Quaglietta (2014) gave a framework of closed loop control scheme for rail traffic management, and quantitative analyses were conducted based on a conflict detection and resolution system and a microscopic railway traffic simulation model. Furthermore, for better understanding of rescheduling problem, we can refer to Cacchiani et al. (2014), which gives a comprehensive review of recovery models and methodologies for real-time railway disturbance managements.

Recently, some researches have focused on the rescheduling problem with the uncertainty and dynamics in a real-time railway environment. For example, Meng and Zhou (2011) developed a single-line train rescheduling model where uncertain segment running times, recover times and the possibility of rescheduling decisions are considered. They designed a multi-layer branching algorithm to minimize the total train delay time. Yang et al. (2014) also proposed a fuzzy optimization approach for train rescheduling in a double-track railway network. A fuzzy variable-based recovery time was derived based on empirical judgments to handle the uncertain durations caused by perturbations.

Due to the rising concerns on urban road congestions and environmental problems, the urban metro system is becoming one of the most convenient and environmental-friendly modes for citizens traveling. In a metro line, trains are scheduled over the space and time to transport passengers from their origin stations to their destinations. Recently, some studies are observed on designing demand-oriented metro timetables in order to reduce passengers' traveling time and the energy-consumption. For example, Niu and Zhou (2013) derived a nonlinear optimization model to optimize a metro passenger train timetable in a heavily congested urban rail corridor, in which the waiting passengers may not board the next arrival train due to train capacity. The model is on the basis of the time-dependent, origin-destination passenger records, and the objective is to minimize the total number of waiting passengers and weighted remaining passengers. The model is solved by a GA through a special binary coding method. Wang et al. (2014) adopted a train scheduling model, i.e., an MINP model (mixed-integer nonlinear programming) that considers the origin-destination-dependent passenger demands. The model aims to minimize the passengers' traveling time and the total energy consumption. Subsequently, they extended this basic model into a small metro network (Wang et al., 2015) by considering the passenger transfer behaviors. Note that, however, although these metro scheduling approaches demonstrate their effectiveness in generating high-quality timetables, they may be unpractical in a real-time metro rescheduling problem. On one hand, the negative influence (such as train delay time, passenger delay, or number of canalled trains) by disturbances is an essential performance indicator in a metro rescheduling problem, which is often not considered in scheduling researches. On the other hand, solving a rescheduling problem is required to be fast enough in practice, whereas timetable planning is an off-line task, for which the computational time may not satisfy the real-time requirements.

### 1.2. The focuses of this study

As shown above, although a variety of studies have focused on the train scheduling problem in normal railway and metro systems, few of them consider the train rescheduling problem for a metro system in order to reduce the negative influences caused by stochastic disturbances. Moreover, there is still no related research for rescheduling metro trains with the consideration of passenger demands, which is typically a practically significant and theoretically challenging issue for the real-world applications since it is urgent to transport the delayed passengers as quickly as possible once an incident occurs. With this concern, we are particularly interested in developing a metro train rescheduling model according to the time-dependent and origin-destination passenger information, as well as an efficient rescheduling algorithm with fast computational speed. Specifically, this paper aims to make the following contributions to the study of metro train rescheduling problems.

(1) A metro train rescheduling model, which not only considers the uncertain and time-variant passenger flow characteristics but also the energy-consumption and regenerative braking energy for train operations, is rigorously formulated. The purpose is to generate a rescheduled solution trade-off among the passenger delay time caused by disturbance, passenger traveling time and total energy-consumption. Since very few of the existing researches consider the inherent uncertainty and dynamics in the real-time rescheduling problem, this is actually a new idea in the literature.
(2) An efficient solution methodology based on approximate dynamic programming (ADP) is developed to produce the rescheduled timetable for the affected trains in case of a disturbance. As indicated by some pioneering researches by Powell (2007), Bertsekas (1995) and Schmid (2012), ADP is a very powerful approach for solving large-scale optimization problems with uncertain and dynamic properties, involving locomotive optimization (Bouzaiene-Ayari et al., 2016), inventory routing problems (Papagergiou et al., 2015), capacity allocation problems (Schutz and Kolisch, 2012), etc, although ADP has never been applied to train rescheduling problem for metro systems. In our study, we first reformulate the proposed rescheduling model into a decision-making process by the definitions of states, policies, state transitions and contribution functions. Then, an integrated train rescheduling algorithm is developed by using function approximations to increase the algorithmic computational speed. The numerical experiments demonstrate that

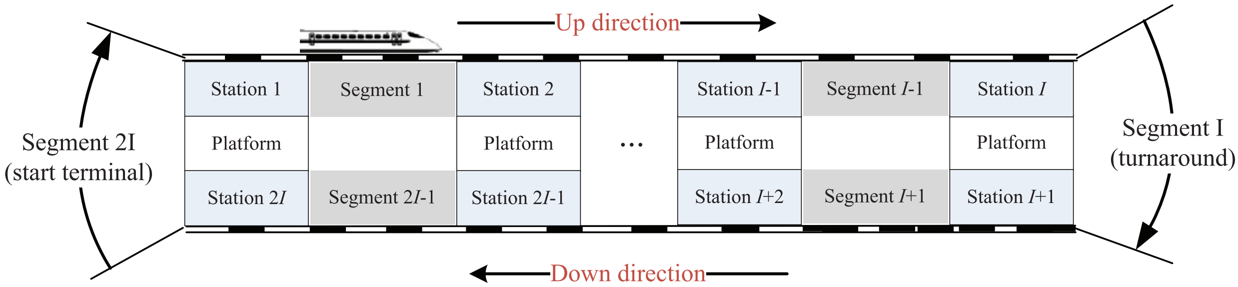



**Fig. 1.** Bi-directional urban rail line.

the proposed approaches can effectively find approximately optimal rescheduled plans by comparing the results with CPLEX solver.

The rest of this paper is organized as follows. In Section 2, we give a description for the metro train rescheduling problem in a double-track line. In Section 3, the formulation of our rescheduling model is described in detail. Then, we propose the solution methodology for metro train rescheduling based on the ADP in Section 4. In Section 5, we give two examples, i.e., a small case and a real-world case based on the real-world detected data in Yizhuang line of Beijing metro, to demonstrate the effectiveness and efficiency of the proposed approaches. We conclude this paper in Section 6.

## 2. Problem statements

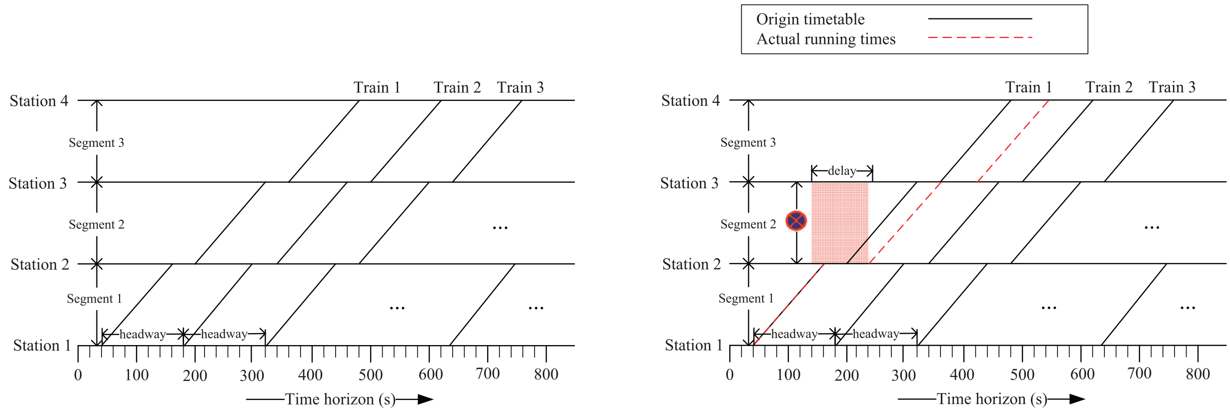
This paper considers a bi-directional urban rail transit line as the operation environment. As shown in Fig. 1, the stops are numbered as station 1, station 2,..., station $2I$. The start terminal and return terminal are indexed by station 1 and station $I$, respectively. A metro train timetable is pre-specified offline, which is usually optimized to achieve a trade-off between service quality and efficiency. The timetable can be expressed as a set of $\bar{a}_{ki}$ and $\bar{d}_{ki}$ for $k \in \{1, 2,..., K\}$ and $i \in \{1, 2,..., 2I\}$, which represent the arrival and departure times for train $k$ at station $i$. According to the predetermined timetable, a train begins its journey at station 1 with up-direction until it reaches station $I$. After a given turn-around time at the return terminal, it goes back to station $2I$ (i.e., station 1 phyically) and then waits for the next cycle.

Different from conventional railway lines that have long distances between adjacent stations, the metro timetables, which are more sensitive with respect to disturbances, should be formulated more rigorously. In general, an urban rail transit line has the following characteristics. (1) Trains follow each other strictly according to the given sequence, which means that both overtaking and crossing operations are prohibited at any time and locations. (2) Since the aim of a metro system is to provide a convenient way for residents' daily traveling while minimizing the operation costs, it is important to consider the potential delays of on-board passengers when a disturbance happens. (3) Even though we can roughly summarize a regular change of passenger demands in each day, the passenger characteristics are difficult to be predicted by deterministic models because of the high uncertainty and dynamics. (4) Due to heavy operation pressures in rush hours, the headway between adjacent trains is usually around 2 min, and some slight time delays (or disturbances) may cause serious consequences without real-time timetable rescheduling and rolling stock adjustments.

According to the definition given by Cacchiani et al. (2014), the disturbances correspond to the fact that certain railway train operation processes, involving driving in a segment or dwelling at a station, last longer than the time specified in the timetable. Once disturbances happen, trains may depart or arrive later than the planned schedule, which results in the so-called delay for both trains and passengers. In general, the primary disturbances in a metro system can be divided into the following three classes. The first type of disturbances refers to the minor incidents that are usually caused by human factors, such as passenger crowding, train driving error, etc, and last for tens of seconds or several minutes. The second of disturbances are caused by the technical failure of infrastructures, for instance the breakdown of track circuits, vehicles, communication or signal equipments, which usually affect the operations of trains for about several minutes. The last type of disturbances is severe accidents due to external factors, for example extreme weather, a fire diaster, although these disturbances seldom occur in the daily operations of a metro system.

An example is shown in Fig. 2. First, a pre-specified timetable for the metro system with four stations and three segments is given by Fig. 2(a). For illustrative purpose, the planned dwelling times at all stations, the planned running times in all segments and the departure headways are 40 s, 120 s and 140 s, respectively. In an ideal environment without any disturbances, the trains depart from the origin station one by one after every departure headway and end at station 4 to prepare the turn-around operations.

We consider the situation that a disturbance is caused by a technical equipment failure. As shown in Fig. 2(b), the incident happens at time 140 s due to power interruption and it disables segment 2 completely for 100 s. This indicates that train 1 cannot leave station 1 until time 240 s when the problem is resolved by metro engineers. If train 1 still follows its planned dwelling time and running time, it will cause the following results at least:




(a) A pre-planned timetable (b) Timetable affected by disturbance

**Fig. 2.** An illustration for metro disturbance that affects the normal timetable.

- Service quality for the passengers is reduced. Due to the time delay, the amounts of waiting passengers at the following stations will greatly increase. The congestion will slower the transportation process (Wang et al., 2014) so that some passengers fail to board the train during the dwelling time. Additionally, the travel time of in-vehicle passengers will be prolonged.
- The following trains are influenced. Since train 1 has to stay at station 1 until the fault is solved, train 2 cannot follow the original timetable in order to guarantee the headway constraints.

For keeping service quality and cost-saving operations once an incident occurs, it is significant for dispatchers to take appropriate decisions to recover from a disturbed situation as soon as possible. Currently, the metro managers in practice have contingency plans to deal with disturbances. For example, in Beijing metro system, a comprehensive plan is formulated and implemented, which contains three levels, i.e., train rescheduling, traveling intervention and Rail-BRT (Bus Rapid Transit) interchange. First, the dispatchers will reschedule the timetable and rolling stock, or change the stop-skip scheme and vehicle marshalling plan. Second, the metro managers can guide the passengers' traveling behaviors by controlling incoming passengers, closing stations or advising them to take other lines (Xu et al., 2014). If the rescheduled rail line still cannot satisfy the need of passengers, the public transportation system has to add Rail-BRT interchange vehicles for passengers to travel with bus transit. We can find that, since the methods by passenger interventions and Rail-BRT interchanges alter passengers' traveling plans, they inevitably impair the passengers' convenience to take metro rail lines. In this sense, train rescheduling is thus the most direct and effective way to help recovery from disturbances. Practically, however, the rescheduling operation is usually a challenging issue if we aim to regenerate a high-quality feasible train timetable quickly with consideration of uncertain features of passenger demands and operational costs. With these concerns, in reality the dispatchers often reschedule the timetable or the rolling stock manually based on their experiences in case of disturbances.

To formulate a metro train rescheduling model and develop effective rescheduling algorithms, we first make the following assumptions throughout this paper:

**Assumption 1.** In order to simplify this problem, we assume in this paper that trains depart from the start terminal in a first-in-first-out manner. Each station can only accommodate one train at any time, and overtaking and crossing operations are prohibited at any positions, which is consistent with what we have featured above.

**Assumption 2.** The disturbances caused by external factors are excluded. This is because that external factors, e.g. extreme weather and serious accidents, are very rare and they will probably result in the large delay of trains. Thus, we only consider the former two types of disturbances that often occur in a metro system and cause relative short time delay. Additionally, we assume that these delays will not affect the rolling stock plan.

**Assumption 3.** For real-time rescheduling in a metro system, we further assume that the stop-skipping strategy is not permitted, since it is not yet widely used in realistic metro networks although a stop-skipping strategy might be effective to reduce the passenger travel time and the operation costs.

Based on the above-mentioned assumptions, the problem of metro train rescheduling in this paper is formally described as follows. With the traveling information of waiting passengers at every stations, the proposed model aims to regenerate an optimal timetable of the affected trains in case of disturbances, which minimizes the delay time of all the affected passengers, their traveling time and total energy consumption. To handle the uncertainty and complexity of the proposed model, and find a good feasible solution quickly, we design an algorithmic framework based on approximate dynamic programming, in which the problem is reformulated as a multistage stochastic decision process and each stage is modeled by the train movement from one station to the next. On each stage, the timetable is rescheduled (if necessary) on the basis of all future expected costs on the trains and at all future times, and then, the system state is updated according to the dwelling time, running time and the number of alighting and boarding passengers. By solving the small subproblems, this process is repeated forward in time until the end of time horizon.

## 3. Mathematical formulation

In this section, we first give notations and decision variables in Section 3.1. The passenger characteristics which describe the dynamic and uncertain features of passenger arriving rates are introduced in Section 3.2 with the given time-varying origin-to-destination passenger demand matrices. In Section 3.3, we present an energy consumption calculation model that considers the regenerative braking energy. In Section 3.4, the metro rescheduling problem is formulated into a stochastic optimization model with the dynamic passenger characteristics and the energy consumption.

### *3.1. Notation description*

#### *3.1.1. Parameters and notations*

The relevant notations are listed below to describe and understand the problem more conveniently.

| | |
|---|---|
| $\mathcal{K}=\{1,2,\ldots K\}$ | Set of affected trains with respect to disturbance |
| $k$ | Index of affected trains, $k\in\mathcal{K}$ |
| $\mathcal{I}=\{1,2,\ldots 2I\}$ | Set of involved stations |
| $i, j$ | Indices of stations and rail segments, $i, j\in\mathcal{I}$ |
| $s_i$ | Length of segment $i$ between station $i$ and station $i+1$ |
| $C_k$ | Boarding capacity of train $k$ |
| $M_k$ | Mass for train $k$ |
| $m_p$ | Average passenger weight |
| $t$ | Time index |
| $t^*$ | Time when a disturbance occurs |
| $t_d$ | Duration time of the disturbance |
| $\alpha_{d,k}$ | Time delay threshold with respect to train $k$ in rescheduling process |
| $t_{turn}$ | Turnaround time |
| $N^i_{t_0,t}$ | Number of arriving passengers at station $i$ from time $t_0$ to $t$ |
| $N^w_i(t)$ | Number of waiting passengers at station $i$ at time $t$ |
| $N^v_k(t)$ | Number of passengers in train $k$ at time $t$ |
| $\lambda_i(t)$ | Passenger arriving rate of station $i$ at time $t$ |
| $N_{a,k,i}$ | Number of alighting passengers of train $k$ at station $i$ |
| $N_{b,k,i}$ | Number of boarding passengers of train $k$ at station $i$ |
| $\overline{a}_{k,i}$ | Arrival time of train $k$ at station $i$ in original schedule |
| $\overline{d}_{k,i}$ | Departure time of train $k$ at station $i$ in original schedule |
| $h_{min}$ | Minimum safe headway |
| $t^{min}_{r,j}$ | Minimal running time in segment $j$ |
| $t^{min}_{d,i}$ | Minimal dwelling time at station $i$ |
| $[t_0, t_{end}]$ | Considered time horizon |
| $s_{ki}(t)$ | Position of train $k$ at segment $i$ and time $t$ |
| $v_{ki}(t)$ | Velocity of train $k$ at segment $i$ and time $t$ |
| $u_{ki}(t)$ | Controller output of train $k$ at segment $i$ and time $t$ |
| $u^{max}_a$ | Maximum accelerating rate |
| $u^{max}_b$ | Minimum braking rate |
| $\alpha, \beta, \gamma$ | Davis parameters of friction forces |
| $\theta_s$ | Gradients of each segment |
| $c_r$ | Regenerative energy-usage coefficient |
| $c_a$ | Available energy percentage after transmission loss |

#### *3.1.2. Decision variables*

| | |
|---|---|
| $a_{k,i}$ | Arrival time to be rescheduled for train $k$ at station $i$ (*Integer variable, unit: second*) |
| $d_{k,i}$ | Departure time to be rescheduled for train $k$ at station $i$ (*Integer variable, unit: second*) |
| $\widetilde{x}_{k,i}$ | Disturbance indictor for train $k$ at station $i$ (*Binary variable*) |

Essentially, the metro train rescheduling problem is to regenerate a set of arrival and departure times for the affected trains, which can keep headway between two adjacent trains to guarantee safe operations. Moreover, we define another

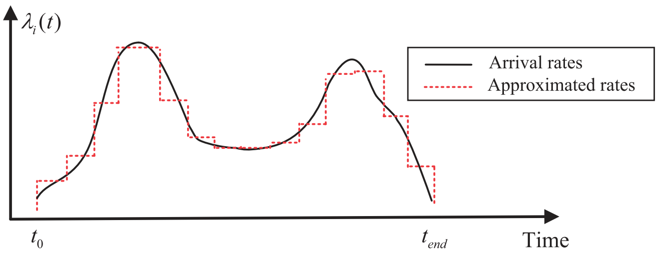



**Fig. 3.** An illustration for passenger arriving rate at a station in a weekday.

decision variable $\widetilde{x}_{k,i}$, i.e., a binary variable to indicate if a train is disrupted in a segment. For example, train $k \in \mathcal{K}$ is delayed at station 2 (see in Fig. 2), and we plan to reschedule its timetable in its following journey. In this case, we let $\widetilde{x}_{k,i} = 0$ for $0 < i < 2$, and $\widetilde{x}_{k,i} = 1$ for $2 \le i \le 2I$, which indicates that we need to reschedule its departure and arrival times at the following stations.

## 3.2. Passenger characteristics

### 3.2.1. Description of passenger arrivals

According to the previous studies by Niu and Zhou (2013), Barrena et al. (2014) and Niu et al. (2015b), the passenger characteristics can be modelled by a time-dependent origin-destination (TOD) table that specifies passengers' arriving and alighting at each station. Given a time period $t \in [t_0, t_{end}]$, the TOD table for a bi-directional urban line with $2I$ stations is given as

$$TOD(t) = \begin{pmatrix} \tau_{1,1}(t) & \cdots & \tau_{1,I}(t) & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \cdots & \vdots \\ 0 & \cdots & \tau_{I,I}(t) & 0 & \cdots & 0 \\ 0 & \cdots & 0 & \tau_{I+1,I+1}(t) & \cdots & \tau_{I+1,2I}(t) \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & \tau_{2I,2I}(t) \end{pmatrix}.$$

In this $2I \times 2I$ TOD table, we use $\tau_{i,j}(t)$ for $i, j \in \mathcal{I}$ to illustrate the passenger arriving amount at station $i$ at time $t$ with destination station $j$, which is subjected to

$$\begin{cases} \tau_{i,j}(t) \ge 0, & \text{if } (1 \le i \le I, 1 \le j \le I, i < j) \cup (I < i \le 2I, I < j \le 2I, i < j) \\ \tau_{i,j}(t) = 0, & \text{otherwise.} \end{cases}$$

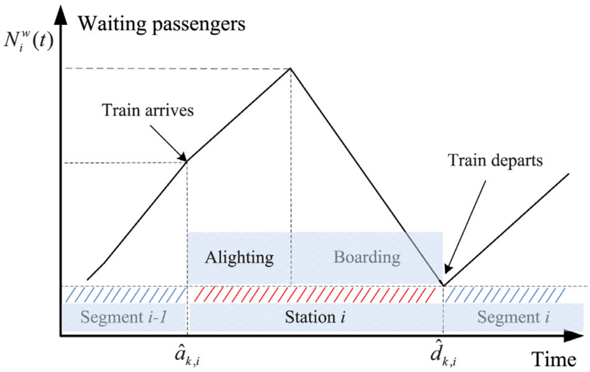
Since station $I$ (i.e., station $I+1$) is the turnaround station, we assume that all the passengers in an up-direction train get off before its turnaround, and the passengers with down-direction will get off at the terminal station $2I$ (i.e., station 1) as well.

We can estimate the approximated time-dependent passenger arriving rates by using the information of historical data or smart card technology (SCD) (Sun et al., 2014; Wang et al., 2015). A typical profile that illustrates the time-variant passenger arriving rates at station $i$ is given with the solid curve in Fig. 3, in which $\lambda_i(t)$ represents the passenger arriving rate at station $i$ and time $t$, which is the sum of passenger arriving rates $\tau_{i,j}(t)$ with destination $j$ for $j \in \{i+1, i+2, \dots, 2I\}$. It can be seen that the passenger arriving amount in peak hours is much larger than that during the off-peak hours.

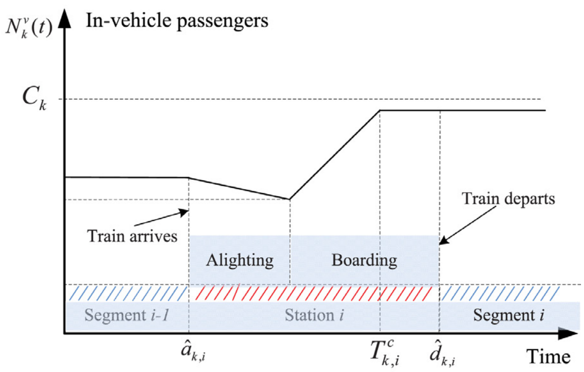
In the previous studies (Niu and Zhou, 2013; Niu et al., 2015b; Wang et al., 2014; 2015), the passengers' demands were all given as deterministic values denoted by time-variant passenger arrival rate $\tau_{i,j}(t)$. Nevertheless, according to some more recent studies (see Xu et al. (2014); 2016b); Yalcinkaya and Bayhan (2009)), the features of arriving passengers at a station over a time period are usually dynamic and uncertain, indicating that it is more reasonable and effective to model the passenger demands as some statistical distributions, e.g., a poisson distribution.

We assume that arrival of passengers at each station $i$ with destination $j$ is a stochastic counting process denoted by a nonhomogeneous poisson process $\{N_c^{i,j}(t), t \ge 0\}$. The intensity function of this poisson process is represented through the historical passenger origin-destination demand $\tau_{i,j}(t)$, which is the input parameter obtained from historical data. Then, we can denote a random variable $N_{t_0,t,\xi}^{i,j} = N_c^{i,j}(t) - N_c^{i,j}(t_0)$ to represent the stochastic number of passengers arriving from time $t_0$ to $t$ at station $i$ with destination $j$. Consequently, given the passenger arriving rate function $\tau_{i,j}(t)$ for $\forall i \in [1, 2I]$ and $\forall j \in (i, 2I]$, the probability that $n_{i,j}$ passengers arrive to the platform with destination $j$ from time $t_0$ to $t$ is given as follows.

$$\Pr\left\{N_{t_0,t,\xi}^{i,j} = n_{i,j}\right\} = \frac{\left[\int_{t_0}^{t} \tau_{i,j}(t')dt'\right]^{n_{i,j}}}{n_{i,j}!} \exp\left[-\int_{t_0}^{t} \tau_{i,j}(t')dt'\right], \tag{1}$$



(a) Number of waiting passengers at station $i$



(b) Number of in-vehicle passengers of train $k$

**Fig. 4.** An illustration of passenger number variations of train $k$ at station $i$.

in which $n_{i,j} \in \{0, 1,\ldots\}$ is a non-negative integer. By using above formulations, both dynamics and randomness of passenger arrivals can be well-described in a unified formulation.

**Remark 3.1.** Since poisson process $\{N_c^{i,j}(t), t \geq 0\}$ has independent increments according to its definition (Ross, 2014), the number of arriving passengers at any time interval is a nonnegative random variable, i.e., $N^{i,j}_{s,s+t,\xi} \geq 0$, for all $s, s+t \in [t_0, t_{end}]$. And the expected value of random variable $N^{i,j}_{s,s+t,\xi}$ is a function with respect to the estimated passenger arrival rates $\tau_{i,j}(t)$, i.e.,

$$E[N^{i,j}_{s,s+t,\xi} = N_c^{i,j}(s+t) - N_c^{i,j}(s)] = \int_s^{s+t} \tau_{i,j}(y)dy.$$

Moreover, the arriving passengers $N^i_{t_0,t,\xi}$ at station $i$ from time $t_0$ to $t$ is equal to the sum of arriving passengers $N^{i,j}_{t_0,t,\xi}$ with destination $j$ for all $j \in \{i+1, i+2, \ldots, 2I\}$, i.e., $N^i_{t_0,t,\xi} = \sum_{j=i+1}^{2I} N^{i,j}_{t_0,t,\xi}$.

*3.2.2. Passenger boarding and alighting*

Next, we aim to analyze the passengers' transfer process at stations. For clarify, we illustrate a situation that train $k$ arrives at station $i$ and time $t$ ($1 \leq k \leq K$, $1 \leq i \leq 2I$, $t_0 \leq t \leq t_{end}$) and dwells for the passengers' alighting and then boarding with the constraints of train boarding capacity. Two scenarios will be considered for train $k$ at station $i$. In the first case, no disturbance occurs and the original timetable is used. The other case considers the rescheduled train timetable. By introducing decision variable $\widetilde{x}_{k,i}$, we then use $\widehat{a}_{k,i}$ and $\widehat{d}_{k,i}$ to integrate arrival and departure times of these two scenarios, given below:

$$\widehat{a}_{k,i} = (1 - \widetilde{x}_{k,i})\bar{a}_{k,i} + \widetilde{x}_{k,i}a_{k,i},$$

$$\widehat{d}_{k,i} = (1 - \widetilde{x}_{k,i})\bar{d}_{k,i} + \widetilde{x}_{k,i}d_{k,i}.$$

First, we denote two time-dependent variables, i.e., $N_k^v(t)$ and $N_i^w(t)$, which represent the numbers of in-vehicle passengers for train $k$ and waiting passengers at station $i$, respectively. As shown in Fig. 4, $N_k^v(t)$ is a constant when the train is running in a segment, and it changes when passengers alight and board, while $N_i^w(t)$ keeps changing over the time horizon since passengers always come to the station and then wait for trains. As shown in Fig. 4(a), $N_i^w(\widehat{a}_{k,i})$ and $N_i^w(\widehat{d}_{k,i})$ represent the number of waiting passengers at station $i$ when train $k$ arrives and departs. In other words, $N_i^w(\widehat{d}_{k,i})$ represents the passengers who are left by train $k$ at station $i$ due to maximum train capacity. Since the waiting passengers are composed of the delayed passengers who are left by the former train and new arrival passengers, we have

$$\begin{aligned} N_i^w(\widehat{a}_{k,i}) &= \sum_{j=i+1}^{2I} N_{i,j}^w(\widehat{a}_{k,i}) \\ &= \sum_{j=i+1}^{2I} \left[ N_{i,j}^w(\widehat{d}_{k-1,i}) + N^{i,j}_{\widehat{d}_{k-1,i},\widehat{a}_{k,i},\xi} \right], \end{aligned} \tag{2}$$

where $N_{i,j}^w(\widehat{a}_{k,i})$ is the number of waiting passengers at station $i$ with destination $j$ for all $j \in \{i+1, i+2, \ldots, 2I\}$ when train $k$ arrives, and $N_{i,j}^w(\widehat{d}_{k-1,i})$ is the number of passengers left by train $k-1$. Meanwhile, the number of in-vehicle passengers, as shown in Fig. 4(b), remains to be constant when the train is running in each segment, and varies when the train is dwelling

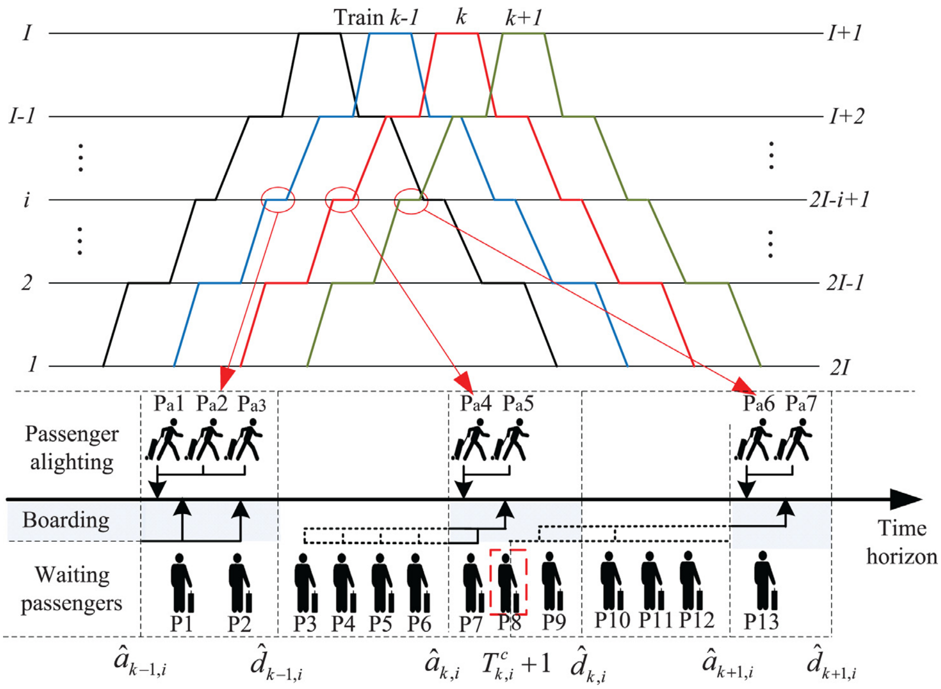



**Fig. 5.** An example for passenger queuing and transfers.

at a station due to passenger boarding and alighting. This means that, the number of in-vehicle passengers in train $k$ when it leaves station $i$ can be expressed as

$$N_k^v(\widehat{d}_{k,i}) = N_k^v(\widehat{a}_{k,i}) - N_{a,k,i} + N_{b,k,i}. \tag{3}$$

where $N_{a,\,k,\,i}$ and $N_{b,\,k,\,i}$ are the numbers of alighting and boarding passengers during the dwelling time period of train $k$ at station $i$. Next, we focus on how to calculate $N_{a,\,k,\,i}$ and $N_{b,\,k,\,i}$. In the first station, i.e., $i = 1$, we have $N_{a,k,i} = 0$ since there is no alighting passengers. For each $i > 1$, we note that, the passengers alighting from train $k$ at station $i$ are composed of passengers who have boarded the train with destination $i$, denoted by

$$N_{a,k,i} = \sum_{r=1}^{i-1} N_{b,k,r,i},$$

where $N_{b,\,k,\,r,\,i}$ is the number of boarding passengers with destination station $i$ for all stations $r \in \{1, 2, \ldots, i-1\}$. To obtain the number of boarding passengers, Niu and Zhou (2013) introduced the concept of effective passenger loading time period for each train, which considers two different cases rigorously subject to the train capacity and arriving passengers amount. (1) The first case is that train $k$ has enough boarding capacity so that all the passengers can get on the train before it departs. In this case, we can use a time window $(\widehat{a}_{k,i}, \widehat{d}_{k,i}]$ to represent the effective passenger boarding process. (2) For the second case, however, there will be passengers left by train $k$ if the amount of queueing passengers is larger than currently remainder train capacity, and the passenger boarding time window in this case is denoted by $(T^c_{k-1,i}, T^c_{k,i}]$, where $T^c_{k,i} \in (T^c_{k-1,i}, \widehat{d}_{k,i}]$ is defined as a critical timestamp (see in Fig. 4(b) for illustration) that represents the latest arrival time of passengers at station $i$ who can board the current train $k$, i.e.,

$$T^c_{k,i} = \min\left\{\widehat{d}_{k,i}, \max\left\{t | N_k^v(t) < C_k \le N_k^v(t) + N^i_{t,t+1,\xi}\right\}\right\}, \quad \forall i \in \mathcal{I}, k \in \mathcal{K}; \quad T^c_{0,i} = 0. \tag{4}$$

An example is shown in Fig. 5, for which the capacities for train $k-1$, $k$ and $k+1$ are assumed to be 35 when they arrive at station $i$. Each symbol “P” (P1 to P13) and “Pa” (Pa1 to Pa7) represent 10 boarding passengers and 10 alighting passengers, respectively, who arrive at this station over discrete timestamps (Here, the time can be discretized by seconds). In the operations, when train $k-1$ arrives at station $i$, a total of 20 currently waiting passengers (denoted by P1 and P2) can board train $k-1$ after 30 persons (denoted by Pa1, Pa2 and Pa3) alight this train due to the enough train capacity. Then, passengers P3-P9 subsequently arrive at station $i$ before train $k$ departs from this station. After passengers Pa4 and Pa5 alight from train $k$, the remainder capacity of train $k$ is 55, meaning that at most 55 queuing passengers are allowed to board this train. Obviously, the passengers P3-P7 who are in the front of the waiting queue will board this train before timestamp $T^c_{k,i}$. Then, at time $T^c_{k,i}+1$ (unit: second), another ten persons (P8) arrive. In this situation, since the remaining train capacity is only left 5, these ten passengers (i.e., P8) are prohibited to board this train for computational simplicity even with slightly empty space in the train. It is worth noting that, this assumption will not make an perceptional influence on the result if the time interval is short enough (e.g., 1 s). Based on the above analyses, the boarding passengers at station $i$ with destination $j$ is given as $N_{b,k,i,j} = N^{i,j}_{T^c_{k-1,i}, T^c_{k,i}, \xi}$ for any $k \in \mathcal{K}$.

Then, the total number of boarding passengers at station $i$ is composed of boarding passengers with destination $j$ for all $j \in \{i+1, i+2, \ldots, 2I\}$, i.e.,

$$N_{b,k,i} = \sum_{j=i+1}^{2I} N_{b,k,i,j}.$$

We can thus calculate the number of passengers that are left by train $k$ when it departs from station $i$ by

$$N^w_i(\widehat{d}_{k,i}) = \max\left\{N^w_i(\widehat{a}_{k,i}) + N^i_{\widehat{a}_{k,i}, \widehat{d}_{k,i}, \xi} - N_{b,k,i}, 0\right\}. \tag{5}$$

This means that, if the train capacity satisfies passenger demands and all the waiting passengers can board train $k$, nobody will be left on the platform. Otherwise, the left passengers have to wait for the next train $k+1$.

### 3.3. Energy-consumption calculation

When a train departs from station $i$, the electricity energy will be consumed for starting the train to head for the next station. In general, the energy-consumption for metro train operations is mainly affected by the running time of a train in each segment. Commonly, the shorter running time in a segment will lead to the exponential increase of energy-consumption (Cucala et al., 2012). Then, in this rescheduling problem, we are also interested in considering the trade-off between the service quality (traveling time and delay time of passengers) and the total energy-consumption. To this end, this section aims to analyze the estimation of energy-consumption for delayed trains on a metro line. For illustrative convenience, we describe the train operation model in a single rail segment to calculate the regenerative energy (which is produced in the braking process and can be utilized by accelerating trains simultaneously) and the total energy-consumption.

#### 3.3.1. Energy-efficient metro train operation analyses

First, a general metro train control model for each train $k \in \mathcal{K}$ in segment $i \in \{1 \le i < 2I, i \ne I\}$ can be given as follows

$$M_{ki}\frac{dv_{ki}(t)}{dt} = M_{ki}u_{ki}(t) - f_r(v_{ki}) - f_g(s_{ki}), \tag{6}$$

$$\frac{ds_{ki}(t)}{dt} = v_{ki}(t), \quad \forall t \in [\widehat{d}_{k,i}, \widehat{a}_{k,i+1}] \tag{7}$$

$$M_{ki} = M_k + m_p N^v_k(\widehat{d}_{k,i}), \tag{8}$$

$$v_{ki}(\widehat{d}_{k,i}) = 0, \tag{9}$$

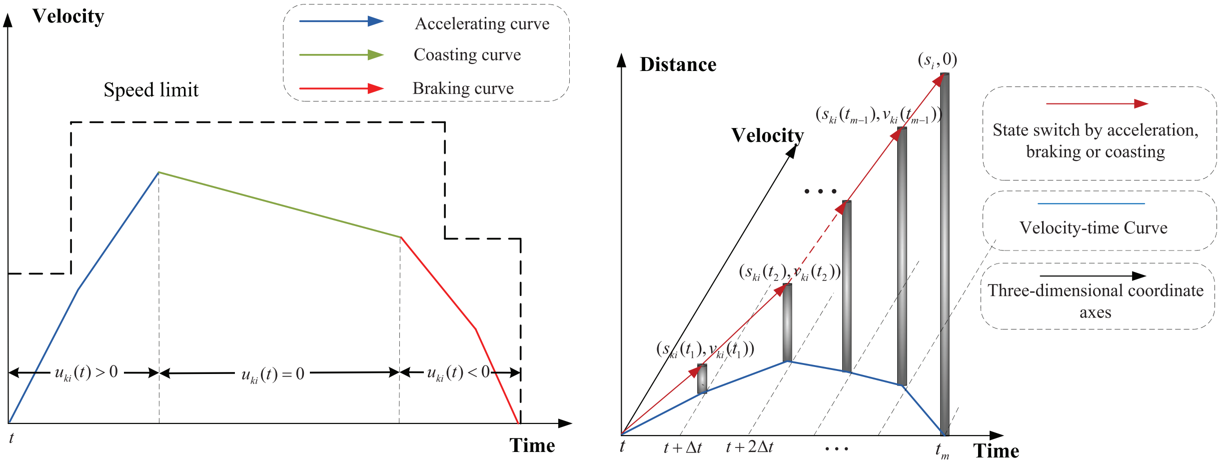
$$v_{ki}(\widehat{a}_{k,i+1}) = 0, \tag{10}$$

$$f_r = M_{ki}(\alpha + \beta v_{ki} + \gamma v_{ki}^2), \tag{11}$$

$$f_g = M_{ki} g \sin(\theta_s), \tag{12}$$

where Eqs. (6) and (7) are train motion equations, $s_{ki}(t)$, $v_{ki}(t)$ and $u_{ki}(t)$ represent the traveling distance, speed and control rate, respectively, which are subject to $0 \le s_{ki}(t) \le s_i$ and $u_{min} \le u_{ki}(t) \le u_{max}$, $M_{ki}$ expresses the weight $M_k$ of train $k$ plus the weight of in-vehicle passengers in segment $i$, Eqs. (9) and (10) indicate the initial and ending speed of train $k$ in segment $i$, in which $\widehat{d}_{k,i}$ and $\widehat{a}_{k,i+1}$ represent the departure time from station $i$ and arrival time at station $i+1$ of train $k$, and $f_r$, $f_g$ represent the friction resistance and gradient force, respectively.

Then, we consider the movement of train $k$ with the speed $v_{ki}(t)$ in a given time window $t \in [\widehat{d}_{k,i}, \widehat{a}_{k,i+1}]$ for each train $k \in \mathcal{K}$ in segment $i \in \{1 \le i < 2I, i \ne I\}$. As indicated by Howlett and Pudney (1995), Albrecht et al. (2015b) and Li and Lo (2014), the energy-efficient operations for a train with continuous control on a practical rail segment with relatively flat slope typically contain four phases, i.e., accelerating, cruising, coasting and braking. Nevertheless, in a metro line with short travel distances and shallow slopes, these four train operation phases can be simplified into three phases, i.e., accelerating, coasting and braking (see in Fig. 6(a)). In order to calculate the energy-consumption over each phase, we split the time horizon $[\widehat{d}_{k,i}, \widehat{a}_{k,i+1}]$ into $m$ intervals denoted by $\mathcal{T} = \{0, t_1, t_2 \cdots, t_m\}$, where timestamp 0 and $t_m$ represent the initial and final time for train departure and arrival. As indicated in Fig. 6(b), the duration of a time interval is $\Delta t$, indicating that for each $t_p \in \mathcal{T}$, $t_p = p\Delta t$. Given train controller output $u_{ki}(t_p)$ of train $k$ in segment $i$ for each time interval, as well as the train velocity $v_{ki}(t_p)$ and position $s_{ki}(t_p)$ at timestamp $t_p \in \mathcal{T}$, the resistances (i.e., friction resistance and gradient forces)




(a) A velocity for metro train operations (b) State transition on a discrete model

**Fig. 6.** A typical metro train operation process.

are computed by Eqs. (11) and (12). Then, on the basis of the train control model in Eqs. (6) and (7), the train states (i.e., velocity, position) at time stamp $t_{p+1}$ can be further calculated, expressed as

$$(s_{ki}(t_p), v_{ki}(t_p)) \xrightarrow{u_{ki}(t_p)} (s_{ki}(t_{p+1}), v_{ki}(t_{p+1})).$$

The above procedure is repeated at each timestamp to update the train state and we can obtain the total energy consumption in traction phase by

$$E^a_{ki} = \sum_{t=\widehat{d}_{k,i}}^{t^{acc}_{k,i}} E^a_{kit} = \sum_{t=\widehat{d}_{k,i}}^{t^{acc}_{k,i}} M_{ki} u_{ki}(t) v_{ki}(t), \tag{13}$$

in which $E^a_{kit}$ represents the accelerating energy-consumption in one time interval $[t, t+\Delta t]$ and $t^{acc}_{k,i} \in [\widehat{d}_{k,i}, \widehat{a}_{k,i+1}]$ denotes the end time of accelerating phase. In the coasting phase, the train has no energy consumption with $u_{ki}(t) = 0$ for $t \in (t^{acc}_{k,i}, t^{coa}_{k,i})$, where $t^{coa}_{k,i}$ is the end time of coasting phase. In the braking phase, the kinetic energy of the running train is converted to heat by friction and braking force. In recently years, metro systems with regenerative braking technique have been developed (Li and Lo, 2014; Yang et al., 2013b), which can use an electric motor as an electric generator when the train is braking, and then feed a part of regenerated electricity energy back to contact lines. Considering the weight of train $k$ with in-vehicles passengers, the regenerative energy is given by

$$E^b_{ki} = \sum_{t=t^{coa}_{k,i}}^{\widehat{a}_{k,i+1}} E^b_{kit} = \sum_{t=t^{coa}_{k,i}}^{\widehat{a}_{k,i+1}} M_{ki} c_r u_{ki}(t), \tag{14}$$

where $E^b_{kit}$ represents the regenerative braking energy at time interval $[t, t+\Delta t]$ and $c_r$ is the regenerative energy coefficient from kinetic energy to regenerative energy.

Based on the above analysis, the energy consumption for train $k$ in a running segment $i \in \{1 \le i < 2I, i \ne I\}$ is expressed as

$$E_{ki} = \sum_{t=\widehat{d}_{k,i}}^{t^{acc}_{k,i}} E^a_{kit} + \sum_{t=t^{acc}_{k,i}}^{t^{coa}_{k,i}} E^c_{kit} = \sum_{t=\widehat{d}_{k,i}}^{\widehat{a}_{k,i+1}} \varepsilon(u_{ki}(t)) M_{ki} u_{ki}(t) v_{ki}(t),$$

where $\varepsilon(u_{ki}(t))$ is a function to indicate whether the train is accelerating, i.e.,

$$\varepsilon(u_{ki}(t)) = \begin{cases} 1 & \text{if } u_{ki}(t) > 0, \\ 0 & \text{otherwise.} \end{cases}$$

It is worth noting that a train also consumes energy in turnaround process at stations $I$ and $2I$. Since we consider the turnaround time $t_{turn}$ as a constant value in this paper, the energy-consumption in the turnaround process can be treated as constant values, denoted by $E^a_{kI}$ and $E^a_{k(2I)}$.

### 3.3.2. Regenerative energy

Next, we analyze the utilization of regenerative energy. Note that the regenerative energy by a braking train can be used only if at least one train is accelerating within the same substation simultaneously. Assume that there are $Y$ substations over a metro line denoted by a set $\mathcal{Y}$, and the regenerative energy in each time unit $[t, t + \Delta t]$ for trains is

$$E_r(t) = \sum_{y\in\mathcal{Y}} \min\left\{\sum_{k\in\mathcal{K}} E^a_{kit}\varphi(i, y), c_a \sum_{k\in\mathcal{K}} E^b_{kit}\varphi(i, y)\right\} \tag{15}$$

where $c_a$ is the percentage of available regenerative energy after transmission loss, and $\varphi(i, y)$ defines the relationship between segment $i$ and substation $y$, i.e., $\varphi(i, y) = 1$ if segment $i$ belongs to substation $y$; $\varphi(i, y) = 0$, otherwise. Finally, the total energy-consumption is described as the subtraction of the energy usage for train accelerating and regenerative energy, given below.

$$E_{total} = \sum_{k=1}^{K}\sum_{i=1}^{2I} E_{ki} - \sum_{t=t_0}^{t_{end}} E_r(t). \tag{16}$$

**Remark 3.2.** In order to compute the total energy consumption in Eq. (16), it is obvious that we need to know the train speed $v_{ki}(t)$ or control rate $u_{ki}(t)$ for each $1 \le k \le K$ and $1 \le i \le 2I$. We note that, numerical algorithms for train energy consumption calculation can be very fast and efficient based on an optimal maximum acceleration, coast, maximum brake journey (see Albrecht et al. (2015a, 2015b); Howlett and Pudney (1995); Howlett et al. (2009); Khmelnitsky (2000); Liu and Golovitcher (2003)). In this study, we use a real-time train operation algorithm proposed in Yin et al. (2014) and Yin et al. (2015), which can quickly compute the near-optimal train control rate in a single rail segment with a given trip time. Different from the former works in Yin et al. (2014) and Yin et al. (2015), this study focuses on the energy consumption of a whole system with two directions, multiple segments and more than one trains. In addition, we further consider the number of on-board passengers and regenerative braking energy for the energy consumption calculation in this paper. We present this algorithm in Section 4.

## 3.4. Optimization model

For the train rescheduling problem in a metro line, the objective of this model is to minimize the expected passengers' time delay, traveling time and energy-consumption in a stochastic environment with disturbances. The passengers' delay in this paper is defined as the total deviations between the arrival time of original timetable and that of rescheduled timetable for all the affected passengers. For example, there are 30 persons in a train who board at Jiugong station (the fourth station) and plan to alight at Rongjing station (the eighth station) in Yizhuang metro line. The original timetable for this train's arrival at these two stations are 8: 00 and 8: 09. The train is delayed in Wanyuan-Rongjing segment for 2 min due to signal equipment faults, and it arrives at Rongjing station at 8: 11. Then, the total delay time at Rongjing station for 30 passengers can be calculated by $30 * 2 * 60 = 3600$ s. Therefore, the passenger delay for each train $1 \le k \le K$ at destination $1 \le i \le 2I$ can be expressed as

$$T^{delay}_{k,i} = N_{a,k,i}(\widehat{a}_{k,i} - \overline{a}_{k,i}). \tag{17}$$

In addition, the passengers' traveling time is the dwelling time at station $i$ and train running time in segment $i$ multiplied by the number of in-vehicle passengers, expressed by

$$T^{travel}_{k,i} = \begin{cases} \left[N^v_k(\widehat{a}_{k,i}) - N_{a,k,i}\right](\widehat{d}_{k,i} - \widehat{a}_{k,i}) + N^v_k(\widehat{d}_{k,i})(\widehat{a}_{k,i+1} - \widehat{d}_{k,i}), & \text{if } 1 \le i < 2I \\ 0, & \text{otherwise.} \end{cases} \tag{18}$$

Note that the optimal objective of delay time can be also regarded as a penalty on the traveling time, because the delay time is essentially a part of the traveling time. If a train is delayed at a station for some time period, the traveling time will increase as well. Meanwhile, the objective of traveling time and energy-consumption may influence each other. For example, we can reduce the passengers' traveling time by shortening the running times in each segment, which will always increase the energy-consumption. To obtain a trade-off solution, we give three weight factors, i.e., $w_d$, $w_t$ and $w_e$, to indicate the importance of delay time, traveling time and energy-consumption, respectively. In this sense, we actually transform the time and energy-consumption into their generalized costs with the same unit scales in the objective function. Finally, the train rescheduling problem for a metro line can be formulated as a stochastic optimization model, given below:

$$\begin{aligned} \min_{\mathbf{a},\mathbf{d}} \quad & z = \mathbb{E}\left[ \sum_{k=1}^{K} \sum_{i=1}^{2I} \left( w_d T_{k,i}^{delay} + w_t T_{k,i}^{travel} \right) + w_e E_{total} \right] \\ s.t. \quad & 0 \le d_{k,i} - \overline{d}_{k,i} < \alpha_{d,k}, \quad \forall 1 \le i \le 2I, k \in \mathcal{K} \\ & 0 \le a_{k,i} - \overline{a}_{k,i} < \alpha_{d,k}, \quad \forall 1 \le i \le 2I, k \in \mathcal{K} \\ & a_{k,i+1} - d_{k,i} \ge t_{r,i}^{min}, \quad \forall (1 \le i < I) \cup (I < i < 2I), k \in \mathcal{K} \\ & d_{k,i} - a_{k,i} \ge t_{d,i}^{min}, \quad \forall 1 \le i \le 2I, k \in \mathcal{K} \\ & d_{k,I} + t_{turn} = a_{k,I+1}, \quad \forall k \in \mathcal{K} \\ & N_k^v(t) \le C_k, \quad \forall k \in \mathcal{K}, t \in [t_0, t_{end}] \\ & a_{k,i} + h_{min} \le a_{k+1,i}, \quad \forall 1 \le i \le 2I, 1 \le k < K \\ & d_{k,i} + h_{min} \le d_{k+1,i}, \quad \forall 1 \le i \le 2I, 1 \le k < K \\ & a_{k,i} \in \mathbb{N}^+, \quad \forall 1 \le i \le 2I, k \in \mathcal{K} \\ & d_{k,i} \in \mathbb{N}^+, \quad \forall 1 \le i \le 2I, k \in \mathcal{K} \end{aligned} \tag{19}$$

The first and second constraints intend to guarantee the limited deviation between the rescheduled timetable and the original timetable, in which $\alpha_{d,\,k}$ is defined as the tolerance threshold for the affected trains. For example, a 200 s delay time is caused in a segment, which affects four trains numbered by 1, 2, 3, and 4. We can set $\alpha_{d,1} = 250$ s, $\alpha_{d,2} = 180$ s, $\alpha_{d,3} = 100$ s and $\alpha_{d,4} = 20$ s. Then, these constraints enable to reduce the affected trains after we reschedule the timetable of these four trains. The third and fourth constraints, respectively, represent the minimal train running time in each segment, and the minimal dwelling time at each station. The fifth constraint requires that the turnaround time for each train should be a constant value. The sixth constraint indicates the train loading capacity, i.e., the maximum number of in-vehicle passengers. The seventh and eighth constraints make sure the safe operations of the adjacent trains on the metro line in the sense of headway constraints. The last two constraints denote that the decision variables are positive integer values.

Essentially, the aforementioned model is a stochastic nonlinear programming model with constraints (we also show in Appendix A that the demand-oriented train rescheduling model (19) can be simplified into a mixed integer quadratic programming (MIQP) model under a deterministic environment), which is often computationally intractable with some classical methods, such as branch and bound algorithm, Lagrangian relaxation, CPLEX, LINGO and GAMS, due to (1) the large number of possible options for each train's arrival time and departure time at each station; (2) the uncertain and dynamic properties of the train rescheduling model; (3) strict requirements for solution computation time. In view of this fact, an approximate dynamic programming (ADP) based algorithm will be designed in the next section in order to seek an approximate optimal rescheduled timetable with short computational time.

## 4. Metro train rescheduling with approximate dynamic programming

Dynamic programming (DP) is a powerful mathematical technique for solving many transportation problems, e.g., train scheduling problem in railways (see Cacchiani et al. (2016); Niu et al. (2015a)), vehicle routing problem with pickup and delivery services (Mahmoudi and Zhou, 2016) and optimal coordination of public transit vehicles (Hadas and Ceder, 2010). Nevertheless, the state space for many real-world applications can be immense, making the tradition DP algorithm very computational intensive (Jiang and Powell, 2015). Different from traditional DP algorithms, the foundation of approximate dynamic programming is based on an algorithmic strategy that steps forward through time dimension (or lookahead policy) and value-function approximations to overcome the *curse of dimensionality*, which has been proved to be effective in solving large-scale stochastic optimization problems in an uncertain and dynamic decision-making environment (Powell, 2014). For the ADP approach, a problem is in general formulated into a multi-stage decision process, and the decisions (e.g., rescheduled timetable in this study) that are made on a stage do have an impact on future consequences (e.g., train arrival time, number of in-vehicle and waiting passengers in this study). Then, if the state variables are too complex or too large, we need to design a functional approximation in order to reduce the computing complexity. In the following, we explicitly discuss how to reformulate the problem and then design an integrated metro train rescheduling algorithm based on the ADP.

### 4.1. ADP framework for metro train rescheduling

In practice, one of the main challenging tasks for metro train rescheduling is how to obtain a high-quality solution within a short computational time, which can typically reduce the delays of traveling passengers and the operational costs. Next, we first formulate the train rescheduling problem into a sequential (i.e., multistage) stochastic decision problem, which is also called a stochastic programming model. The fundamental elements for a sequential stochastic optimization problem (Powell, 2014) typically involves the *state*, *decision*, *transition function* and *objective function*.

We propose our model by assuming that the decisions are made in discrete stages when a train reaches each station, i.e., $i = 1, 2, \ldots, 2I,$ since the metro train timetable is allowed to be changed only when a train is found to be delayed at

**Table 1**
Description of elements for metro train rescheduling as a decision-making process.

| | |
|---|---|
| $\mathbf{S}_{k,i}$ | The state vector that describes train $k$ when it arrives at station $i$ |
| $\mathbf{x}_{k,i}$ | The decision variables that train $k$ can make at station $i$ |
| $R(\mathbf{S}_{k,i}, \mathbf{x}_{k,i})$ | Reward function (or contribution function) that refers to the passengers' delay time, traveling time and energy-consumption with implanting the action set, i.e., $\mathbf{x}_{k,i}$, when train $k$ is at state $\mathbf{S}_{k,i}$ |
| $\eta \in (0, 1]$ | Discount factor |
| $\mathbb{S}$ | The set of all possible state vectors |
| $\mathbb{X}$ | The set of possible decisions used to act on a train |

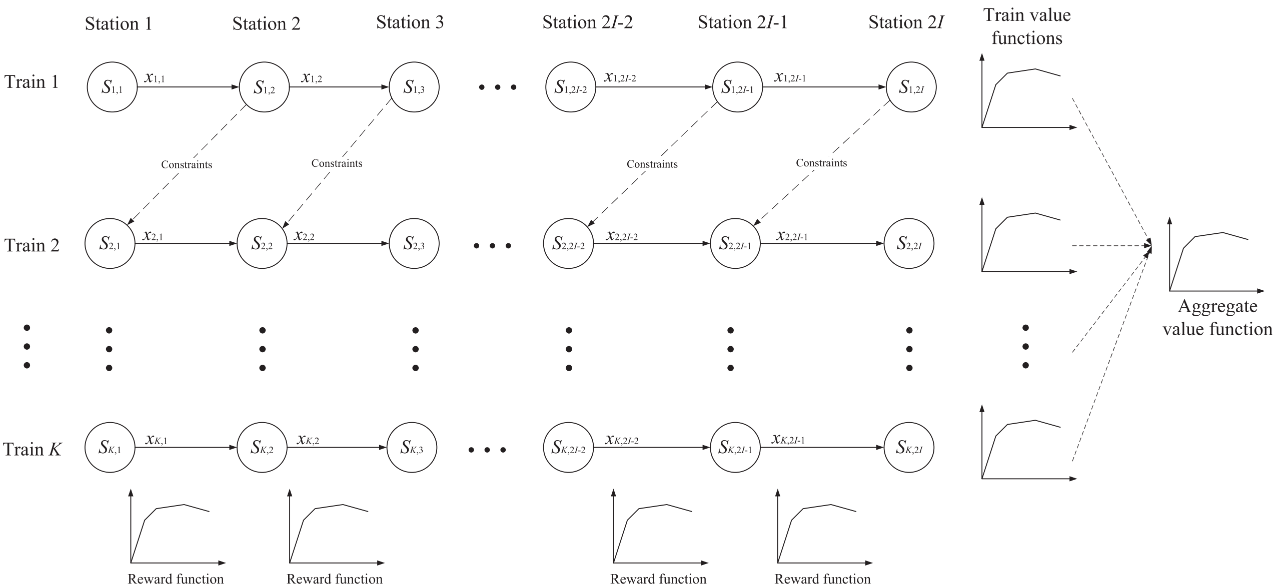



**Fig. 7.** Illustration of train rescheduling policy using a dynamic programming modeling.

a station. In addition, we do not reschedule the timetable of trains that have passed the disabled segment, because the disturbance has no impacts on the operations of these trains. Practically, the first-delayed train can be numbered as train 1, and the following affected trains are numbered as 2, 3,... and $K$ successively. Next, we define the key elements for metro train rescheduling problem in Table 1, using the notation of Powell (2007).

**Definition 1. State and action:** According to the definition given by Powell (2007), a state variable is the minimally dimensioned function of history that is necessary and sufficient to compute the decision function, transition function, and contribution function. For the train rescheduling problem in this study, we should consider not only the arrival and departure times of trains for the rescheduled timetable, but also its influence on the waiting and in-vehicle passengers. For example, if the arrival time of a train is delayed at a station, passengers queueing at the next station will increase due to consecutively arriving passengers. When the train arrives, in some cases, they may be unable to board this train if it is oversaturated. Accordingly, we define the state variable that specifies the arrival time at a station, information of the potential delays, and passenger characteristics. Specifically, we use $T_{a,k,i}$ to represent the actual arrival time of train $k$ at station $i$, and the state variable for train $k$ at station $i$ is denoted by a vector tuple $\mathbf{S}_{k,i} = (\mathbf{T}_{k,i}, \mathbf{c}_{k,i})$, where

$$(\mathbf{T}_{k,i}, \mathbf{c}_{k,i}) \in \mathbb{S}, \quad \forall 1 \le k \le K, 1 \le i \le 2I, \tag{20}$$

$$\mathbf{T}_{k,i} = \{t^*, t_d, T_{a,k,i} | 1 \le i \le 2I, 1 \le k \le K\}, \tag{21}$$

$$\mathbf{c}_{k,i} = \{N_k^v(T_{a,k,i}), N_i^w(T_{a,k,i}) | 1 \le i \le 2I, 1 \le k \le K\}. \tag{22}$$

The elements of state vector $\mathbf{T}_{k,i}$ defines the time when the disturbance happens, the duration of this disturbance and the current time of train $k$ when it arrives at station $i$, respectively. The state vector $\mathbf{c}_{k,i}$ denotes the number of in-vehicle passengers and the number of waiting passengers when train $k$ arrives at station $i$. Thus, we can suppose the scenario that, a sequence of trains $k \in \mathcal{K}$ are delayed at station $i_s$ due to an incident that occurs at segment $i_s$ and time $t^*$. According to the definition, the original state in this ADP formulation can be denoted by $\mathbf{S}_{k,i_s} = (t^*, t_d, T_{a,k,i_s}, N_k^v(T_{a,k,i_s}), N_i^w(T_{a,k,i_s}))$, in which $N_i^v(T_{a,k,i_s})$ and $N_i^w(T_{a,k,i_s})$ are given parameters that represent the original number of in-vehicle and waiting passengers.

In a dynamic programming problem, a policy is defined as $\pi(s)$ that returns an action vector $\mathbf{x}$ at state $\mathbf{S}$. Fig. 7 illustrates the detailed state transition process of trains in a metro train rescheduling process, by following a sequence of policies. For instance, train $k$ arrives at station $i$ with state $\mathbf{S}_{k,i}$ (i.e., disturbance information, current time, numbers of in-vehicle and waiting passengers). Thus, the action vector $\mathbf{x}_{k,i} \in \mathbb{X}$ should be defined as its dwelling time and running time in its next state, expressed by

$$\mathbf{x}_{k,i} = \{\widetilde{x}_{k,i}, t^d_{k,i}, t^r_{k,i} | 1 \le i \le 2I\}, \quad \forall 1 \le k \le K, \tag{23}$$

where $t^d_{k,i}$ and $t^r_{k,i}$ represent the potential dwelling time and running time for train $k$ at station $i$ and segment $i$, respectively. Also, we can get the relationship between the action vector defined in Eq. (23) and the decision variables in the rescheduling model (Eq. 19) for $\forall 1 \le k \le K$, denoted by

$$\begin{aligned} t^d_{k,i} &= \widehat{d}_{k,i} - \widehat{a}_{k,i}, \quad \forall 1 \le i \le 2I, \\ t^r_{k,i} &= \begin{cases} \widehat{a}_{k,i+1} - \widehat{d}_{k,i} & \text{if } (1 \le i < I) \cup (I < i < 2I), \\ t_{turn} & \text{if } i = I \text{ or } i = 2I, \end{cases} \end{aligned} \tag{24}$$

where the running time $t^r_{k,i}$ in segments $I$ and $2I$ is defined as the turnaround time $t_{turn}$.

Next, we model the dynamics of metro train management, i.e., the transition of the state of trains and passenger amounts at each decision stage. The state transition process of train rescheduling problem is illustrated in Fig. 7, which explicitly depicts the system state transition network. An example is that train $k$ arrives at station $i$ with state $\mathbf{S}_{k,i}$. The dispatcher makes a decision $\mathbf{x}_{k,i} = X^{\pi}(\mathbf{S}_{k,i})$ to decide if it is needed to reschedule the timetable. The train dwells and departs from this station according to the given instructions, and then it arrives at the next station with state $\mathbf{S}_{k,i+1}$ according to a state transition function, that is,

$$\mathbf{S}_{k,i+1} = S^M(\mathbf{S}_{k,i}, \mathbf{x}_{k,i}). \tag{25}$$

Specifically, the state transition function $S^M$ (which is also called a *base model*) involves the transition of nodes for trains and number of passengers. First, the arrival time $T_{a,k,i+1}$ of train $k$ is denoted by its previous state (i.e., its arrival time at station $i$) and the corresponding action taken at station $i$. In this process, we should consider three cases: 1) no disturbance happens in segment $i$ and this train arrives at station $i+1$ normally; 2) a disturbance occurs in segment $i$, which delays the train in this segment for time $t_d$; 3) the train is at the turnaround station. These three cases can be formulated by

$$T_{a,k,i+1} = \begin{cases} T_{a,k,i} + t^d_{k,i} + t^r_{k,i} & \text{if } t^* \le T_{a,k,i}, i \ne I, \\ T_{a,k,i} + t^d_{k,i} + t^r_{k,i} + t_d & \text{if } T_{a,k,i} \le t^* < T_{a,k,i} + t^d_{k,i} + t^r_{k,i}, i \ne I, \\ T_{a,k,i} + t^d_{k,i} + t_{turn} & \text{if } i = I, \end{cases} \tag{26}$$

for all $1 \le k \le K$. Besides, Eqs. (2)–(5) that model the passengers' arriving, alighting and boarding processes can be reformulated to describe the transition of passengers, i.e., the transition of in-vehicle passengers and waiting passengers when train $k \in K$ travels from station $i$ to station $i+1$, expressed as

$$N^v_k(T_{a,k,i+1}) = \begin{cases} N^v_k(T_{a,k,i}) - N_{a,i,k} + N_{b,i,k} & \text{if } 1 < i < 2I, i \ne I, \\ 0 & \text{if } i = 1 \text{ or } i = I, \end{cases} \tag{27}$$

$$N^w_{i+1}(T_{a,k,i+1}) = \begin{cases} N^w_{i+1}(t_0) + N^{i+1}_{t_0, T_{a,k,i+1}, \xi} & \text{if } k = 1, \\ N^w_{i+1}(T_{a,k-1,i+1} + t^d_{k-1,i+1}) + N^{i+1}_{T_{a,k-1,i+1}, T_{a,k,i+1}, \xi} & \text{otherwise}, \end{cases} \tag{28}$$

where $N^w_i(t_0)$ represents the initial number of waiting passengers at station $i$, $N^{i+1}_{t_0, T_{a,k,i+1}, \xi}$ and $N^{i+1}_{T_{a,k-1,i+1}, T_{a,k,i+1}, \xi}$ are random variables, and we define $k = 1$ as the first affected train. To handle the randomness of input demand data, in the searching process of ADP, we need to sample different random numbers according to the nonhomogeneous poisson distribution in Eq. (1) to simulate the arrival rate of the passengers (for more details for dealing with uncertainties by ADP, please also see Powell (2007)).

Note that, in this study, we consider the rescheduling of more than one trains, and the first affected train may also have impacts on the following trains. As shown in Fig. 7, we first derive the state transition of train 1 from station $i$ to destination $2I$ according to Eqs. (26), (27) and (28). The rescheduling of train 1 will probably bring some new constraints (e.g., safe headway constraints) to the following trains, and it will also change the number of waiting passengers at each station. On the basis of rescheduled plan for train 1, we then denote the state transition of train 2, and the process is repeated until all the affected trains are recovered from a disrupted situation.

#### 4.1.1. Objective function

In order to optimize the given objective function in an ADP problem, we first need to define a contribution function (or reward function), which is used to evaluated the action $\mathbf{x}_{k,i}$ on state $\mathbf{S}_{k,i}$ (see Fig. 8). Since the objective of this rescheduling model is to minimize the expected value of passenger delay time and traveling time, as well as the energy-consumption in an uncertain environment with disturbance, formulated by Eq. (19), the reward for each $i$ and $k$ ($1 \le k \le K$, $1 \le i < 2I$) is

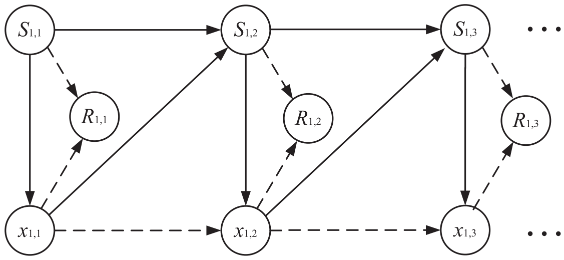



**Fig. 8.** Illustration of state transition with reward function.

$$R(\mathbf{S}_{k,i}, \mathbf{x}_{k,i}) = w_d T_{k,i}^{delay} + w_t T_{k,i}^{travel} + w_e \left( E_{ki} - \sum_{t=T_{a,k,i}}^{T_{a,k,i+1}} E_r(t) \right), \quad \forall\ \mathbf{S}_{k,i} \in \mathbb{S}, \mathbf{x}_{k,i} \in \mathbb{X}. \tag{29}$$

Then, the objective aims to find the optimal policy that minimize the expected total costs for all trains at every decision stage, i.e.,

$$\min_{\pi} \mathbb{E} \sum_{1 \le k \le K} \sum_{1 \le i < 2I} R(\mathbf{S}_{k,i}, X^{\pi}(\mathbf{S}_{k,i})). \tag{30}$$

### *4.1.2. Constraints*

The constraints represented by action $\mathbf{x}_i^k \in \mathbb{X}$ need to capture the basic requirements in a rescheduling process. We next reformulate constraints in Eq. (19) as follows in order to propose the ADP model.

$$0 \le \sum_{i'=1}^{i} t_{k,i'}^{d} + \sum_{i'=1}^{i} t_{k,i'}^{r} - t_{k,i}^{r} + \bar{d}_{k,i} \le \alpha_{d,k}, \quad \forall 1 \le i \le 2I, k \in \mathcal{K} \tag{C1}$$

$$0 \le \sum_{i'=1}^{i-1} t_{k,i'}^{d} + \sum_{i'=1}^{i-1} t_{k,i'}^{r} - \bar{a}_{k,i} \le \alpha_{d,k}, \quad \forall 1 < i \le 2I, k \in \mathcal{K} \tag{C2}$$

$$t_{k,i}^{r} \ge t_{r,i}^{min}, \quad \forall (1 \le i < I) \cup (I < i < 2I), k \in \mathcal{K} \tag{C3}$$

$$t_{k,i}^{d} \ge t_{d,i}^{min}, \quad \forall 1 \le i \le 2I, k \in \mathcal{K} \tag{C4}$$

$$t_{k,I}^{r} = t_{turn}, \quad \forall k \in \mathcal{K} \tag{C5}$$

$$N_k^{v}(t) \le C_k, \quad \forall k \in \mathcal{K}, t \in [t_0, t_{end}] \tag{C6}$$

$$\sum_{i'=1}^{i} (t_{k,i'}^{d} + t_{k,i'}^{r}) - \sum_{i'=1}^{i} (t_{k+1,i'}^{d} + t_{k+1,i'}^{r}) \le -h_{min}, \quad \forall 1 \le i \le 2I, 1 \le k < K \tag{C7}$$

$$\sum_{i'=1}^{i} (t_{k,i'}^{d} + t_{k,i'}^{r}) - \sum_{i'=1}^{i} (t_{k+1,i'}^{d} + t_{k+1,i'}^{r}) \le -h_{min} + t_{k,i}^{r} - t_{k+1,i}^{r}, \quad \forall 1 \le i \le 2I, 1 \le k < K \tag{C8}$$

$$t_{k,i}^{d} \in \mathbb{N}^{+}, \quad \forall 1 \le i \le 2I, k \in \mathcal{K} \tag{C9}$$

$$t_{k,i}^{r} \in \mathbb{N}^{+}, \quad \forall 1 \le i \le 2I, k \in \mathcal{K} \tag{C10}$$

These constraints (i.e., C1 to C10) are consistent with those given by Eq. (19), and they essentially define the feasible region $\mathbb{X}$ for all possible actions that can be taken by trains at each decision stage.

## *4.2. Solution methodology*

Based on the ADP theory, this section focuses on designing an integrated efficient train rescheduling algorithm with fast computing speed, in order to improve the real-time performances of metro operations.

*4.2.1. Calculation of the potentially affected trains*

The first problem for the metro dispatchers in case of a disturbance is to estimate the number of trains to be rescheduled. Ideally, it is better to reschedule less trains as much as possible to minimize the severity of the disturbance. For example, due to an equipment fault in a segment $i$ $(1 \le i \le I-1)$, we have to postpone the departure time of the first impacted train at station $i$ for at least $t_d$ seconds. If this incident duration $t_d$ exceeds the timetable-specified headway between adjacent trains, we usually have to reschedule the next several successive trains to guarantee the safe operations. By these methods, the perturbation can be expectedly absorbed by the rescheduled timetable as soon as possible and the remainder trains can be operated without disturbances. In reality, rescheduling a part of directly impacted trains instead of all the following trains is practically desirable. Typically, the number of affected trains can be estimated by the duration of disturbance $t_d$, operation headway $h_k$ and minimum headway $h_{min}$. One example is that, a train is informed that the forward segment is disabled when it arrives at a station, its departure time from the current station is postponed for 150 s. Then, the following train has to change its arrival time at this station in order to satisfy the headway constraints, which means that the duration time can be absorbed by $h_k - h_{min}$ at most. This process can be followed by the following successive trains, and finally the duration time can be completely absorbed by a finite number of trains. Here, we can use a heuristic method to calculate the number of minimum affected trains by $K = \max\{1, \lceil \frac{t_d}{|h_k - h_{min}|} \rceil\}$. In real-world applications, the number of rescheduled trains should be larger than $K$ due to the uncertainty of the driving behaviors. Then the fluctuant number on the basis of the minimum number $K$ can be given to well address this issue.

**Algorithm 1** Train energy consumption calculation algorithm

**Step 1.** Initialize the parameters: set $m = 1$; initial train state $s_{ki}(t_m) = 0$, $v_{ki}(t_m) = 0$, and $t_m = 0$;
**Step 2.** If $s_{ki}(t_m) > s_{p,i+1}$ (i.e., the start point of parking area in station $i+1$), go to Step 5; otherwise, input the train state through the knowledge base, and obtain the feasible solution set $U$ for train controller output;
**Step 3.** Improve the policy and value function, and then calculate the optimal train controller output by

$$u^*(t_m) = \min_{u_m \in U}\{r(x_m, u) + \gamma J^*(x_{m+1})\}; \tag{31}$$

**Step 4.** Let $m \leftarrow m+1$ and update train state $s_{ki}$ and $v_{ki}$ by Eq. (6) to Eq. (12). Then, go to Step 2;
**Step 5.** Brake the train at the next station;
**Step 6.** Output the energy-consumption and regenerativeenergy by Eqs. (13) and (15).

After the number of affected trains is determined, we can reschedule the timetable of the affected trains in order to make a trade-off between the negative impacts on the traveling passengers and energy-consumption.

*4.2.2. Energy-consumption calculation for train operations*

Next, we introduce the energy-consumption calculation method in order to compute the objective function. In the previous studies by Yin et al. (2014, 2015), a real-time train operation (RTO) algorithm is derived on the basis of a knowledge-based system, where the controller outputs for a train can be obtained in real-time with the given segment running time. Here, we use the RTO algorithm to calculate the segment energy-consumption and regenerative energy for train $k \in \mathcal{K}$ in segment $i$, described in Algorithm 1. In this algorithmic procedure, $x_m = (s_{ki}(t_m), v_{ki}(t_m))$ is defined as the train state, and $r(x_m, u)$ is the reward function that represents the energy-consumption with state $x_m$ and action $u$.

*4.2.3. Lookahead policy for timetable rescheduling*

In the metro train rescheduling problem, we need to make decisions over the planning stages for the affected trains, in order to minimize the expected passenger traveling time and energy consumption. The objective function defined in Eq. (30) can be formulated into an optimal policy that is given by using the function

$$X^*_{k,i}(\mathbf{S}_{k,i}) = \arg\min_{\mathbf{x}_{k,i}} \left\{ R(\mathbf{S}_{k,i}, \mathbf{x}_{k,i}) + \min_{\pi} \mathbb{E}\left[ \sum_{k\in\mathcal{K}} \sum_{i'=i+1}^{2I} R(\mathbf{S}_{k,i'}, X^{\pi}_{i'}(\mathbf{S}_{k,i'})) | \mathbf{S}_{k,i} \right] \right\} \tag{32}$$

where $\mathbf{S}_{k,i+1} = S^M(\mathbf{S}_{k,i}, \mathbf{x}_{k,i})$. This function shows that we aim to find the best rescheduled timetable for the affected trains through considering the impacts of actions on the rest planning stages. However, this function cannot be solved directly due to the stochastic characteristics and complexity of the transition function $S^M$. Therefore, we first derive the Bellman's equation for solving the stochastic train rescheduling problem according to the Bellman's principle of optimality (Bellman, 1954), given below.

$$\begin{aligned} V_{k,i}(\mathbf{S}_{k,i}) &= \min_{\mathbf{x}\in\mathbb{X}} [R(\mathbf{S}_{k,i}, \mathbf{x}_{k,i}) + \eta \mathbb{E}\{V_{k,i+1}(\mathbf{S}') | \mathbf{S}_{k,i}\}] \\ &= \min_{\mathbf{x}\in\mathbb{X}} \left[ R(\mathbf{S}_{k,i}, \mathbf{x}_{k,i}) + \eta \sum_{\mathbf{S}'\in\mathbb{S}} P(\mathbf{S}' | \mathbf{S}_{k,i}, \mathbf{x}_{k,i}) V_{k,i+1}(\mathbf{S}') \right], \quad \forall \mathbf{S}_{k,i} \in \mathbb{S}, \mathbf{x}_{k,i} \in \mathbb{X}, i \in \mathcal{I}, k \in \mathcal{K}, \end{aligned} \tag{33}$$

where $V_{k,\,i}$ is the value of being in state $\mathbf{S}_{k,i}$, $\mathbf{S}'$ represents all the possible states if we are in state $\mathbf{S} = \mathbf{S}_{k,i}$ and take action $\mathbf{x}_{k,i}$, and $\eta$ is the discount factor with $0 < \eta \le 1$ that indicates the discounted value of future rewards. Then, the optimal lookahead policy can be calculated by using the function

$$\mathbf{x}_{k,i}(\mathbf{S}_{k,i}) = \arg\min_{\mathbf{x}\in\mathbb{X}} \left[ R(\mathbf{S}_{k,i}, \mathbf{x}_{k,i}) + \eta \sum_{\mathbf{S}'\in\mathbb{S}} P(\mathbf{S}'|\mathbf{S}_{k,i}, \mathbf{x}_{k,i}) V_{k,i+1}(\mathbf{S}') \right], \forall \mathbf{S}_{k,i} \in \mathbb{S}, \mathbf{x}_{k,i} \in \mathbb{X}, i \in \mathcal{I}, k \in \mathcal{K}. \tag{34}$$

This dynamic programming problem can be solved theoretically by applying a lookup table that represents the policies and value functions in Eq. (33) to optimize the reward function over the planning horizon. More specifically, we illustrate the metro train rescheduling procedure by a lookahead policy in Algorithm 2.

**Algorithm 2** Metro train rescheduling by ADP with lookahead policy

**Step 1.** (Initialization) Set $V_{k,i}(\mathbf{S}_{k,i}) = 0$, $\forall \mathbf{S}_{k,i}$, $i \in \mathcal{I}$, $k \in \mathcal{K}$, choose initial state of trains, and set $n = 1$.
**Step 2.** If $n \le N$ (the maximum iteration time), go to Step 3; otherwise, go to step 5.
**Step 3.** Do for $k = 1, \dots, K$;
**Step 3.1** Do for $i = 1, \dots, 2I$;
(a) if there is no disturbance for train $k$, keep the original timetable and set $a_{k,i} = \bar{a}_{ki}$ and $d_{k,i} = \bar{d}_{ki}$;
(b) else if $i = I$ that indicates that the train is at the turnaround station, set $a_{k,i+1} = d_{k,i} + t_{turn}$;
(c) else, calculate the lookahead policy by Eq. (34).
**Step 3.2** If $i > 1$ and $i \neq I$, update the value function by using

$$V_{k,i}(\mathbf{S}_{k,i}) \leftarrow \min_{\mathbf{x}\in\mathbb{X}} \left[ R(\mathbf{S}_{k,i}, \mathbf{x}_{k,i}) + \eta \sum_{\mathbf{S}'\in\mathbb{S}} P(\mathbf{S}'|\mathbf{S}_{k,i}, \mathbf{x}_{k,i}) V_{k,i+1}(\mathbf{S}') \right], \tag{35}$$

where the reward function can be calculated by Algorithm 1 and Eq. (29).
**Step 3.3** Update the post-decision state by

$$\mathbf{S}_{k,i+1} = S^M(\mathbf{S}_{k,i}, \mathbf{x}_{k,i}). \tag{36}$$

**Step 4.** Set $n \leftarrow n + 1$ and go to step 2.
**Step 5.** Output the final value function that is represented by a lookup table.

In this algorithm, we try to approximate the value function with a lookup table, and update the lookup table in each iteration according to Eq. (34). In general, this algorithm with a lookup table can work well when the dimension of our problem is relatively small, and the value function approximation can quickly converge to a good result. However, as pointed out by Powell (2007), this traditional dynamic programming formulation faces the well-known three curses of dimensionality, i.e., states, outcomes and actions. For example, when we consider the real-world case of Beijing metro Yizhuang line, which is a bi-directional urban rail line consisting 13 stations (i.e., $2I = 26$ stops) with cycle time 4254 s, the number of possible states can be as much as $4254^{26}$. In addition, if we consider the state vector $\mathbf{c}_{k,i}$, the established lookup table can become extremely large, which consumes a lot of computer's memory to solve the train rescheduling problem with dynamic programming. To overcome this drawback in the solution process, we shall introduce the value function approximations by replacing the lookup table with a regression model for the formulated train rescheduling problem in the following discussion.

*4.2.4. Value function approximations*

The approximate dynamic programming provides an elegant framework to overcome the curse of dimensionality, and it is also a powerful tool in solving complex multi-stage decision problems with stochastic characteristics (Fang et al., 2013; Zhang and Adelman, 2009). In the problem of interest, we use a $Q$-function $Q(\mathbf{S}_{k,i}, \mathbf{x}_{k,i})$ to represent the cost-to-go function, which is calculated by

$$Q(\mathbf{S}_{k,i}, \mathbf{x}_{k,i}) = \sum_{\mathbf{S}'\in\mathbb{S}} P(\mathbf{S}'|\mathbf{S}_{k,i}, \mathbf{x}_{k,i}) V_{k,i+1}(\mathbf{S}'), \tag{37}$$

where $\mathbf{S}'$ expresses the next possible state. The $Q$-function directly captures the optimal expected future reward with respect to train state $\mathbf{S}_{k,i}$ (disturbance information, arrival time at station $i$, number of in-vehicle passengers and waiting passengers) and action $\mathbf{x}_{k,i}$ (train rescheduling indicator, rescheduled dwelling and running time). Then, the train rescheduling problem is to derive an effective method to approximate the $Q$-function in order to make the representation of the $Q$-function scalable to improve the computational efficiency and memory allocation. According to (Medury and Madanat, 2013), we use $\overline{Q}$, which is summed up by a set of linear, separable basis functions, to estimate the $Q$-function for train $k$:

$$\overline{Q}(\mathbf{S}_{k,i}, \mathbf{x}_{k,i}) = \sum_{f\in\mathcal{F}} \theta_f(\mathbf{S}_{k,i}, \mathbf{x}_{k,i}) \phi_f(\mathbf{S}_{k,i}, \mathbf{x}_{k,i}), \tag{38}$$

where $f$ is a positive integer that represents the number of basis functions, $\phi$ is the basis function that specifies the attributes of the $Q$-function (i.e., train state vector $\mathbf{S}_{k,i}$ and action vector $\mathbf{x}_{k,i}$), and $\theta_f$ is the regression coefficient with respect to the train state and action, which is iteratively updated for calculating the optimal value function. The sequence of

$\theta_f(\mathbf{S}_{k,i}, \mathbf{x}_{k,i})$ for every $f \in \mathcal{F}$ can be defined as a parameter vector $\vec{\theta}_{k,i}$. Note that we are approximating trains at the aggregation level represented by $\mathcal{F}$, which captures train arrival time, numbers of in-vehicle passengers and waiting passengers, and number of metro stops.

Next, we use an example to demonstrate the use of value function approximations to derive a decision $\mathbf{x}_{k,i}$ at a planning stage, in which we assume that train $k$ is at state $\mathbf{S}_{k,i}$ and we want to determine its rescheduled dwelling time $t^d_{k,i}$ at station $i$ and traveling time $t^r_{k,i}$ in segment $i$. First, using constraints in Eq. (C1) to Eq. (C10), we can derive an action set $\mathbb{X}$ that defines the feasible dwelling and traveling time $t^d_{k,i}$ and $t^r_{k,i}$, respectively. Then, we compute all the value function approximations $\overline{Q}(\mathbf{S}_{k,i}, \mathbf{x}_{k,i})$ with respect to the possible actions that belong to $\mathbb{X}$ based on the regression coefficient $\theta_f$ in the former iteration. We can thus select the optimal value function that has the minimum future rewards (traveling time, delay time and energy consumption), which corresponds to an optimal action $\mathbf{x}^*_{k,i}$. In other words, the optimal policy for train $k$ at station $i$ in the metro train rescheduling can be obtained by

$$\mathbf{x}^*_{k,i}(\mathbf{S}_{k,i}) = \arg\min_{\mathbf{x}\in\mathbb{X}} \left[ R(\mathbf{S}_{k,i}, \mathbf{x}_{k,i}) + \sum_{f\in\mathcal{F}} \theta_f(\mathbf{S}_{k,i}, \mathbf{x}_{k,i})\phi_f(\mathbf{S}_{k,i}, \mathbf{x}_{k,i}) \right], \forall \mathbf{S}_{k,i} \in \mathbb{S}, \mathbf{x}_{k,i} \in \mathbb{X}, \boldsymbol{i} \in \mathcal{I}, k \in \mathcal{K}.$$

After choosing a policy, we let the state $\mathbf{S}_{k,i}$ transfer to the next state $\mathbf{S}'$ with the given action $\mathbf{x}^*_{k,i}$ and state transition function $S^M$, and then we update the value function approximations, i.e., the regression coefficient $\theta_f$ (see the content below). After all the affected trains arrive at the destination following the above rules, we begin a new iteration until reaching the maximum iteration time.

### 4.2.5. Value function updating

Obviously, we expect that the approximated $\overline{Q}$ function in Eq. (38) is close to the target function $Q$ on the planning horizon, which means to minimize the error between $\overline{Q}$ and $Q$. Here, we use a mean-squared error (MSE) to evaluate the function approximations. This MSE-based value function updating method was given by Sutton and Barto (1998), and we briefly describe its main procedure for the completeness of this paper. First, the MSE between $Q$-function and $\overline{Q}$ is given as

$$\mathrm{MSE}(\vec{\theta}) = \sum_{\mathbf{S}\in\mathbb{S}, \mathbf{x}\in\mathbb{X}} P(\mathbf{S}, \mathbf{x})[Q(\mathbf{S}, \mathbf{x}) - \overline{Q}(\mathbf{S}, \mathbf{x})]^2, \tag{39}$$

where $P(\mathbf{S}, \mathbf{x})$ is a distribution that weights the errors of different status and $\vec{\theta}$ is essentially the parameter vectors given by each $\vec{\theta}_{k,i}$. In terms of the MSE, the ideal goal is to find the optimal parameter vector $\vec{\theta}^*$ such that $MSE(\vec{\theta}^*) \le MSE(\vec{\theta})$ for all possible $\vec{\theta}$. Here, there are two main tasks in order to obtain the approximated value function $\overline{Q}$: (1) we need to find an effective parameter updating method to find the parameter vector $\vec{\theta}^*$; (2) the value of $Q$ is unknown in most circumstances, and we must replace it with some approximation techniques.

In this study, we apply a gradient descent method to update the weights associated with the $Q$ function approximation, which is based on the assumption that the possible states are with the same distribution. Then, the updating rule with the gradient descent method is given as:

$$\begin{aligned} \vec{\theta}^{n+1}_{k,i} &= \vec{\theta}^n_{k,i} - \frac{1}{2}\alpha_n \nabla_{\vec{\theta}_{k,i}} [Q(\mathbf{S}_{k,i}, \mathbf{x}_{k,i}) - \overline{Q}^n(\mathbf{S}_{k,i}, \mathbf{x}_{k,i})]^2 \\ &= \vec{\theta}^n_{k,i} + \alpha_n [Q(\mathbf{S}_{k,i}, \mathbf{x}_{k,i}) - \overline{Q}^n(\mathbf{S}_{k,i}, \mathbf{x}_{k,i})] \nabla_{\vec{\theta}_{k,i}} \overline{Q}^n(\mathbf{S}_{k,i}, \mathbf{x}_{k,i}), \end{aligned} \tag{40}$$

where

$$\overline{Q}^n(\mathbf{S}_{k,i}, \mathbf{x}_{k,i}) = R(\mathbf{S}_{k,i}, \mathbf{x}_{k,i}) + \eta \overline{Q}^{n-1}(\mathbf{S}_{k,i}, \mathbf{x}_{k,i}), \tag{41}$$

$n$ represents the ADP iteration, $\nabla_{\vec{\theta}_{k,i}}$ is the vector of partial derivations (i.e., *gradient*) with respect to $\vec{\theta}_{k,i}$, and $\alpha_n$ is the step size of the gradient algorithm (also called *learning rate*).

Now, the remaining problem for the value function updating is the calculation of $Q$. Note that we can use a TD (temporal difference) error to approximate $Q$ by one-period look-ahead policy. We use $\delta_n = Q(\mathbf{S}_{k,i}, \mathbf{x}_{k,i}) - \overline{Q}^n(\mathbf{S}_{k,i}, \mathbf{x}_{k,i})$ to represent the return of one-step of TD learning, and its updating method is shown as follows:

$$\delta_n = R(\mathbf{S}_{k,i+1}, \mathbf{x}_{k,i+1}) + \eta \overline{Q}^{n-1}(\mathbf{S}_{k,i+1}, \mathbf{x}_{k,i+1}) - \overline{Q}^{n-1}(\mathbf{S}_{k,i}, \mathbf{x}_{k,i}). \tag{42}$$

Then, the complete process for linear function approximation updating is described in Algorithm 3.

### 4.2.6. Procedure of ADP-based algorithm

Based on the introductions of the ADP framework for metro train rescheduling, lookahead policies and value function approximation, an integrated rescheduling algorithm for metro trains is developed in Algorithm 4. The procedure given in Fig. 9 illustrates the overall flowcharts of the ADP-based algorithm.

**Algorithm 3** Updating the value function approximations

**Step 1.** Initialization:
**Step 1.1** Input the iteration time $n$, station number $i$, train number $k$, state $\mathbf{S}_{k,i}$, action $\mathbf{x}_{k,i}$ and the next state $\mathbf{S}'$;
**Step 1.2** Set $\delta_n \leftarrow 0$.
**Step 2.** Do:
**Step 2.1** Update the value of $\delta_n$ by using Eq. (42);
**Step 2.2** Update the weight factor by Eq. (40).
**Step 3.** Update the post-decision state by

$$\mathbf{S}_{k,i+1} = S^M(\mathbf{S}_{k,i}, \mathbf{x}_{k,i}). \tag{43}$$

**Step 4.** Set $n \leftarrow n+1$ and go to step 2.
**Step 5.** Output the updated linear value function approximation.

**Algorithm 4** ADP-based algorithm for metro train rescheduling problem

**Step 1.** Input initial information, including the current time, detailed information of the disturbance, status of trains, and number of waiting and in-vehicle passengers, etc.
**Step 2.** Calculate the value function by applying the lookahead policy in Algorithm 2.
**Step 3.** Update the weight factors of the approximated function according to Algorithm 3.
**Step 4.** Repeat the lookahead policy and the weight factor updating. If the value function satisfies the termination condition, go to step 5; otherwise go to step 2.
**Step 5.** Obtain the best value function and output the rescheduled train timetable.

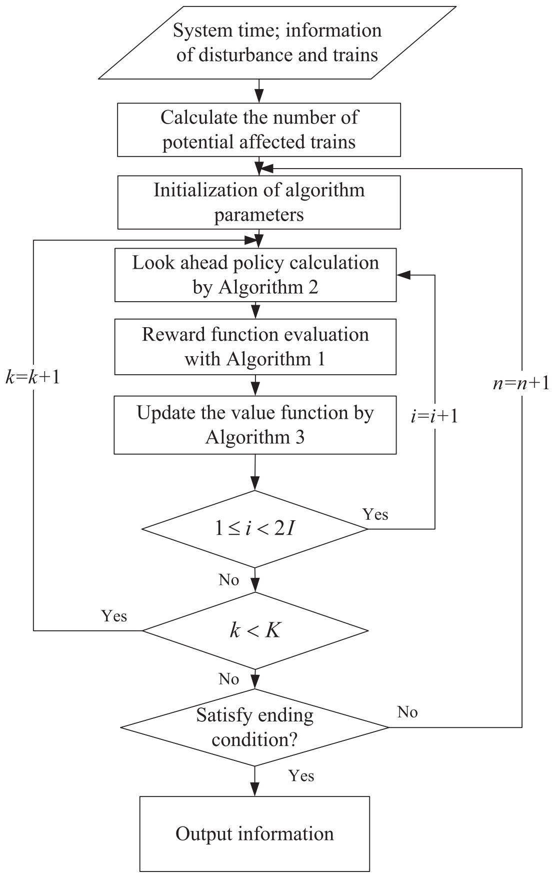



**Fig. 9.** The framework of ADP-based algorithm for metro train rescheduling.

**Table 2**
Parameters of DKZ32 train in the Beijing metro system.

| Parameters | Symbol | Value |
|---|---|---|
| Train mass (*Kg*) | $M$ | $1.99 \times 10^5$ |
| Average passenger mass (*Kg*) | $m_p$ | 60 |
| Train capacity (Number) | $C_k$ | 1468 |
| Davis parameter | $\alpha$ | $1.36 \times 10^{-4}$ |
| Davis parameter | $\beta$ | $1.45 \times 10^{-2}$ |
| Davis parameter | $\gamma$ | 1.244 |
| Regenerative energy-usage coefficient (%) | $c_r$ | 0.75 |
| Available energy percentage (%) | $c_a$ | 0.8 |
| Maximum accelerating rate ($m/s^2$) | $u_a^{max}$ | 0.8 |
| Minimum braking rate ($m/s^2$) | $u_b^{max}$ | −1 |

**Table 3**
The original timetable.

| Train index | Station index (arrival time, departure time) (second) | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 4 | 3 | 2 | 1 |
| 1 | 0, 40 | 160, 200 | 320, 360 | 480, 520 | 640, 680 | 800, 840 | 960, 1000 | 1120, 1160 |
| 2 | 140, 180 | 300, 340 | 460, 500 | 620, 660 | 780, 820 | 940, 980 | 1100, 1140 | 1260, 1300 |
| 3 | 280, 320 | 440, 480 | 600, 640 | 760, 800 | 920, 960 | 1080, 1120 | 1240, 1280 | 1400, 1440 |

**Table 4**
The TOD matrix that represents passenger arrival rates (passengers/second).

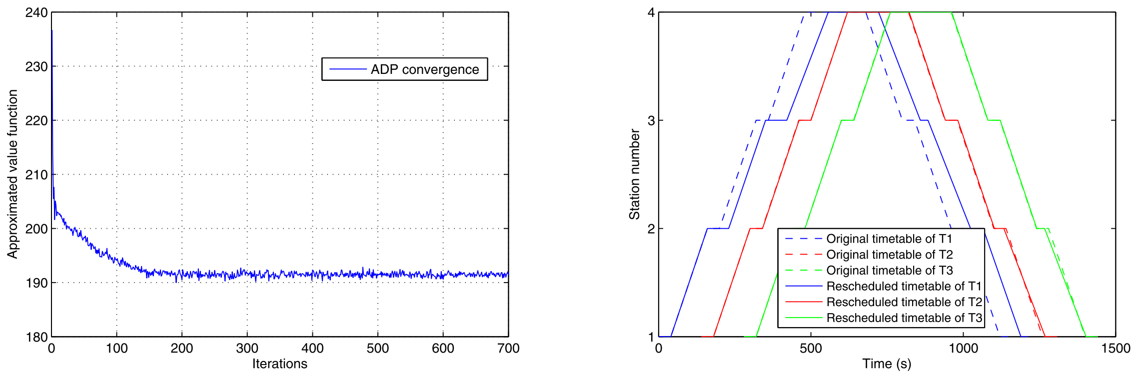
| Station index | 1 | 2 | 3 | 4 | 4 | 3 | 2 | 1 |
|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0.12 | 0.60 | 0.84 | 0 | 0 | 0 | 0 |
| 2 | 0 | 0 | 0.10 | 0.12 | 0 | 0 | 0 | 0 |
| 3 | 0 | 0 | 0 | 0.60 | 0 | 0 | 0 | 0 |
| 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | 0 | 0 | 0 | 0 | 0 | 0.30 | 0.40 | 0.40 |
| 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0.43 | 0.56 |
| 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.80 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

## 5. Numerical examples

In this section, we implement two sets of numerical examples to demonstrate the efficiency and effectiveness of the ADP-based approach for the metro train rescheduling problem. In the first example, we consider a simplified deterministic model by setting all the origin-destination oriented passenger demands as constants. For this case, a comparison among the results by CPLEX solver, a heuristic method and the proposed ADP approach is given to verify the performance of the proposed algorithm. Then, in Example 2, we demonstrate how to use the proposed approaches to solve the stochastic optimization model of this metro train rescheduling problem on the real-world Beijing Yizhuang metro line with practically detected datasets.

### 5.1. A small case study

This example considers the railway structure described in Fig. 2 as the experimental environment, which is a typically bi-directional urban rail line. In order to test the effectiveness of the ADP-based solution approach, we derive a simplified deterministic model (described in Appendix A), which is demonstrated to be a mixed-integer quadratic programming (MIQP) problem that can be solved by both CPLEX solver and ADP. Clearly, in this case study, there are four stations with one turnaround terminus and three running segments. The distances of different segments are set as 1600 m, 1500 m and 1300 m, respectively. The type of service trains is called DKZ32 electric multiple unit (EMU), which is commonly used in the Beijing metro system, and the parameters of DKZ32 EMU are illustrated in Table 2. We consider the scenario that a disturbance occurs at time $t = 140$ s, and it disables segment 2 for 100 s. The first train is delayed at station 2 and the following two consecutive trains are also affected during their operations. The original timetable for these three affected trains is illustrated in Table 3, in which we assume that the time horizon starts from time $t = 0$ *s*. In this case, train 1 arrives at station 1 at time $t = 0$ s and departs at time $t = 40$ s according to the original timetable. The time when train 1 arrives at station 2 is $t_0 = 160$ s, at which the numbers of passengers in train 1 with destinations of stations 2, 3 and 4 are 150, 250 and 200, respectively. The second-specified average origin-to-destination passenger demands are shown in Table 4.




(a) Convergence of the value function approximation (b) The original timetable and rescheduled timetable

**Fig. 10.** Computational results with ADP approach.

**Table 5**
The rescheduled timetable for the affected trains.

| Train index | Station index (arrival time, departure time) (second) | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 4 | 3 | 2 | 1 |
| 1 | 0, 40 | 160, 230 | 344, 414 | 538, 563 | 663, 703 | 843, 873 | 1013, 1055 | 1175, 1210 |
| 2 | 140, 180 | 300, 340 | 460, 500 | 620, 660 | 780, 820 | 940, 980 | 1100, 1138 | 1263, 1300 |
| 3 | 280, 320 | 440, 480 | 600, 640 | 760, 800 | 920, 960 | 1080, 1120 | 1240, 1275 | 1400, 1440 |

The parameters corresponding to ADP are set as follows: the discounting factor $\lambda$ is taken as 0.98; the step size is set as $1/N_{ite}$ where $N_{ite}$ is the iteration times; the maximum iteration time is assumed to be 700. Now, we solve the deterministic model by implementing the proposed ADP-based rescheduling algorithm in this study. As shown in Fig. 10(a), the approximated value function converges after about 200 iterations. We illustrate the best rescheduled plan for the affected trains in Table 5. In addition, a comparison between the best rescheduled timetable and the original timetable is shown in Fig. 10(b). It can be seen that, train 1 is delayed at station 2 due to the occurrence of disturbance, causing the deviation from the original timetable. With the rescheduled timetable, both train 2 and train 3 finally arrive at the destination on time. This indicates that, after rescheduling the affected trains by the proposed ADP-based algorithm, we can reduce the negative affects of some disturbances on the metro line.

In addition, we derive a heuristic method (HEM), which is commonly used by the metro dispatchers in practice to deal with these kinds of small delays by simply postponing the arrival times and departure times for all the affected trains (Schmocher et al., 2005). For example, in this case, since train 1 is delayed at station 2 for 30 s, we also need to delay the departure and arrival times of train 2 and train 3 for 30 s, respectively. Typically, this heuristic method is effective to guarantee the safe headway between trains, while it is not economical enough because it will inevitably increase both the delay time and traveling time of involved passengers. Thus, this method is available only when the disturbance duration time is very short (within about 2 min). Moreover, we also adopt IBM ILOG CPLEX 12.5.1 to solve this MIQP problem for solution comparisons, where parameters in CPLEX solver are all set as their default values.

In this set of experiments, a total of five instances are tested with different disturbance occurrence times and durations. Table 6 shows the computational results with respect to HEM, ADP and CPLEX for the different instances, and the weight values for $w_d$, $w_t$ and $w_e$ are set as 10.0, 1.0 and 1.0, respectively. In this table, the first column denotes the tested disturbance scenarios, the second column denotes the solution methods, the third to sixth columns represent the produced best objective value, delay time of passengers, their traveling time and energy-consumption, and the seventh and eighth columns show the computational time and gap, respectively.

It follows from Table 6 that, for these five instances, the time delay and traveling time are obviously reduced by using the generated solutions of ADP and CPLEX. The objective values with both ADP and CPLEX are improved by about 15% than those of the HEM, while the energy consumptions of these three approaches are very close. This is essentially because that the rescheduled model by ADP and CPLEX imposes constraints for trains' arrival time delay by retrenching the original timetable. Meanwhile, since the rescheduling solution is a trade-off among three indicators, the energy-consumption may be increased a little to meet the reduction of time delays. In addition, the returned gap in CPLEX solver nearly approaches 0 when it terminates, and the best objective value of ADP is very close to that of CPLEX, which demonstrates that the ADP is very effective to obtain a solution with good quality. Besides, we can see that the computational times for these five instances by using ADP (about 10 seconds) are all about halves of those by CPLEX, which typically satisfy the real-time requirements of metro train rescheduling in practical applications.

**Table 6**
Computational results by HEM, ADP and CPLEX for different instances.

| Instance index | Method | Best objective value ($\cdot\ 10^4$) | Total delay time (s) | Traveling time (s) | Operation costs (KJ) | Computation time(s) | Gap (%) |
|---|---|---|---|---|---|---|---|
| 1 | HEM | 224.79 | $1.42 \cdot 10^5$ | $5.16 \cdot 10^5$ | $2.79 \cdot 10^5$ | – | – |
| $t_d = 70$ | ADP | 190.47 | $1.12 \cdot 10^5$ | $5.03 \cdot 10^5$ | $2.83 \cdot 10^5$ | 9.2 | – |
| $t^* = 140$ | CPLEX | 189.2 | $1.15 \cdot 10^5$ | $4.71 \cdot 10^5$ | $2.78 \cdot 10^5$ | 19.4 | 0% |
| 2 | HEM | 243.46 | $1.53 \cdot 10^5$ | $6.23 \cdot 10^5$ | $2.81 \cdot 10^5$ | – | – |
| $t_d = 80$ | ADP | 214.33 | $1.32 \cdot 10^5$ | $5.40 \cdot 10^5$ | $2.82 \cdot 10^5$ | 9.4 | – |
| $t^* = 140$ | CPLEX | 212.90 | $1.34 \cdot 10^5$ | $5.12 \cdot 10^5$ | $2.77 \cdot 10^5$ | 21.5 | 0% |
| 3 | HEM | 300.46 | $1.89 \cdot 10^5$ | $8.32 \cdot 10^5$ | $2.83 \cdot 10^5$ | – | – |
| $t_d = 90$ | ADP | 278.65 | $1.65 \cdot 10^5$ | $8.52 \cdot 10^5$ | $2.84 \cdot 10^5$ | 9.8 | – |
| $t^* = 140$ | CPLEX | 269.88 | $1.64 \cdot 10^5$ | $7.76 \cdot 10^5$ | $2.82 \cdot 10^5$ | 27.8 | 0.8% |
| 4 | HEM | 174.11 | $1.01 \cdot 10^5$ | $4.64 \cdot 10^5$ | $2.67 \cdot 10^5$ | – | – |
| $t_d = 70$ | ADP | 161.74 | $0.92 \cdot 10^5$ | $4.27 \cdot 10^5$ | $2.70 \cdot 10^5$ | 8.90 | – |
| $t^* = 400$ | CPLEX | 154.91 | $0.89 \cdot 10^5$ | $3.91 \cdot 10^5$ | $2.68 \cdot 10^5$ | 18.2 | 0% |
| 5 | HEM | 195.82 | $1.25 \cdot 10^5$ | $4.39 \cdot 10^5$ | $2.69 \cdot 10^5$ | – | – |
| $t_d = 90$ | ADP | 172.39 | $0.99 \cdot 10^5$ | $4.67 \cdot 10^5$ | $2.67 \cdot 10^5$ | 10.2 | – |
| $t^* = 400$ | CPLEX | 172.20 | $1.00 \cdot 10^5$ | $4.53 \cdot 10^5$ | $2.69 \cdot 10^5$ | 17.7 | 0% |


**Fig. 11.** Illustration of Beijing metro Yizhuang line.

## 5.2. A real-world case study on Beijing metro Yizhuang line

In this example, we consider a real-world case study over the Beijing metro Yizhuang line. In the implementations, the uncertain and dynamic parameters (i.e., passengers' time-dependent origan-destination demands) are all taken from historical detected operation data. We next execute this set of experiments with different disturbance durations and weight values to verify the effectiveness and robustness of the proposed ADP-based approach for metro train rescheduling problem.

### 5.2.1. Experiment description and parameter settings

The Yizhuang line is a bi-directional metro line consisting of 13 stations and 12 running segments with a total length of 23.3 km (see Fig. 11). In daily operations of Yizhuang line, the planned cycle time is 4254 s, the minimal headway is $h_{min} = 90$ s and the turnaround time is $t_{turn} = 110$ s(see Wang et al. (2014)). Some basic operation data of Yizhuang line, including the distances of each segment, the planned dwelling time and running time, the speed limit and the maximal passenger arriving rate (MPAR) at each station, are all illustrated in Table 10 (see Appendix B). In the numerical experiments, we consider the planning horizon from 8: 30 to 9: 15 in morning peak hours, during which a total of 21 trains are operated. In this study, we use the real-world passenger demand data at each station collected on a weekend of October 2014. Due to page limitations, we only show the average demand profiles for different stations in Fig. 12. It is obvious that the passenger demand varies throughout a day, which is typically related to the different stations and service times. For example, the passenger demands at stations in the middle of the line, e.g., Jiugong, Jinghai, are much higher than those at other stations.

In the following, we derive different disturbance scenarios that usually happen in Yizhuang line to test the proposed approaches. Based on the practical experiences, the segment between Jiugong and Yizhuang qiao (up direction) is sensitive to the disturbances due to the high density of traveling passengers. Another sample segment is from Wenhua yuan to

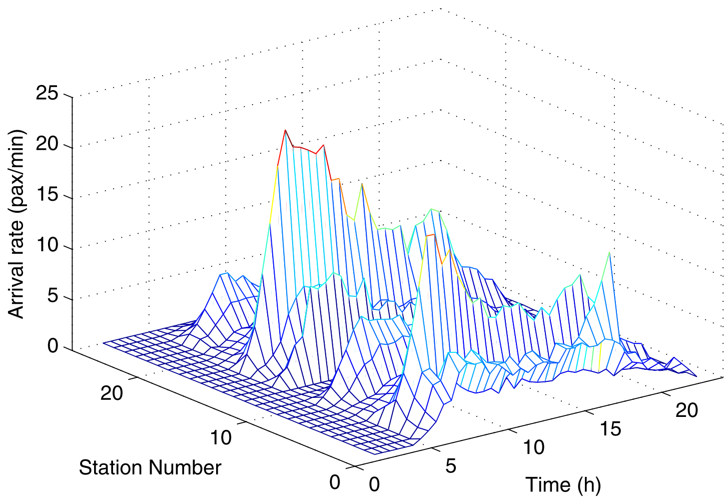



**Fig. 12.** Passenger demand variations of Yizhuang line on a weekend.

**Table 7**
Settings of four disturbance scenarios.

| Scenario index | Initial time | Occurrence segment | Duration time (s) |
|---|---|---|---|
| N1 | 8: 30 | Jiugong and Yizhuang qiao | 100.0 |
| N2 | 8: 30 | Jiugong and Yizhuang qiao | 150.0 |
| N3 | 8: 30 | Wenhua yuan - Wanyuan | 100.0 |
| N4 | 8: 30 | Wenhua yuan - Wanyuan | 150.0 |

Wanyuan (up direction), where the amount of transfer passengers are too huge so that trains are always delayed at Wenhua yuan station. Then, we give four disturbance scenarios in the implementations, and the detailed settings are listed in Table 7. In this table, for instance, scenario $N2$ means that, when the first affected train arrives at Jiugong station, it is informed that the segment ahead is unavailable until 100 seconds later. During this time period, we can use the rescheduling methods to update the timetable of all the affected trains in order to reduce the negative influences on passengers and metro trains.

*5.2.2. Computational results*

We note that the departure interval of Yizhuang Line in rush hours varies from 140 s to 160 s in the experiments, and thus we can specify that at least three trains are impacted by the disturbances given above. Then, we use the ADP-based algorithm to calculate the rescheduled timetable for aforementioned four scenarios, in which the discount factor and learning rate are set as the same values as those in the first example, and the maximum iteration time is set as 800. We show the convergence plots of the four scenarios in Fig. 13. The weight coefficients for passenger time delay, their traveling time and energy consumption are pre-given as 8.0, 1.0 and 3.0, respectively. The initial solutions are obtained by HEM, i.e., a commonly used method in real-world operations. It can be seen from Fig. 13 that the objective values for scenarios $N1$, $N2$, $N3$ and $N4$ converge to the optimal solutions after about 200 iterations. In addition, the performance improvement is about 10% for all these four scenarios in comparison to the results by HEM. Moreover, it is interesting to see that the near-optimal objective values for $N2$ and $N4$ are much larger than those of $N1$ and $N3$. This is essentially because that we consider much longer disturbance durations for scenarios $N2$ and $N4$, which directly causes larger delays for passengers and then leads to larger optimal objective values.

For clarity, we give a detailed performance comparison between the results of HEM and ADP for scenarios $N1$-$N4$. As illustrated in Table 8, the values of the objective function, the passenger delay time, their traveling time, the energy consumption, and the computational time are clearly specified. It is shown that the time delays of passengers are greatly reduced by about 25%–30% by using the ADP-based approaches, which in turn reduces the passenger traveling time. The energy-consumption of ADP is a little higher than that of HEM, since ADP needs to shorten the running and dwelling times for the affected trains in order to recover from the original timetable quickly, which will inevitably increase the energy consumption. In addition, the near-optimal objective value of ADP outperforms that of HEM in all these four cases by about 10%. The computational results illustrate that ADP achieves better than HEM in reducing passengers' delay and traveling time, while maintaining nearly the same energy-consumption. Additionally, we can see that the computational time of ADP algorithm is only about 45–60 s, which can satisfy the requirements for metro train managements, making it appropriate for the real-time implementations. Besides, Fig. 14(a) to Fig. 14(d) illustrate the original and rescheduled timetables by ADP algorithm for these four instances. It can be seen that a disturbance delays train 1 (T1) at a station for 100 or 150 s, which also affects

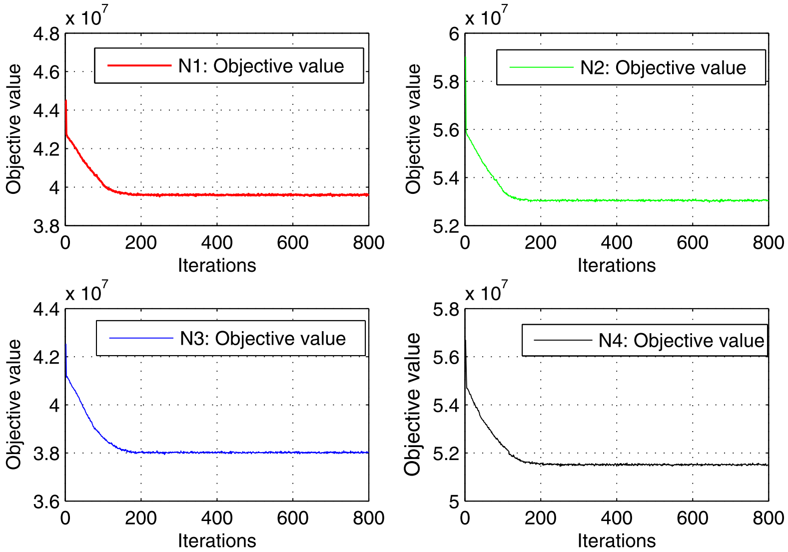



**Fig. 13.** The convergence process of ADP for train timetable rescheduling.

**Table 8**
Result comparisons for different approaches.

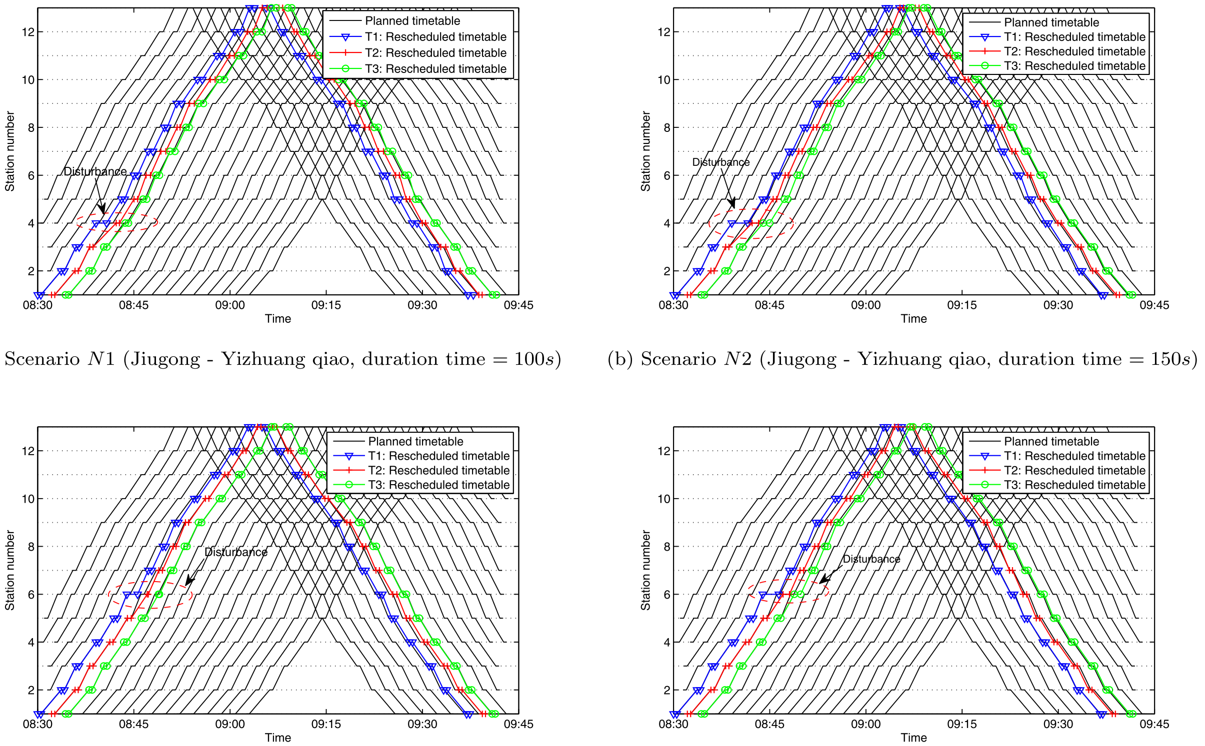
| Scenario index | Method | Best objective value | Total delay time (s) | Traveling time (s) | Operation costs (KJ) | Computation time (s) |
|---|---|---|---|---|---|---|
| *N*1 | HEM | $4.47 \cdot 10^7$ | $1.79 \cdot 10^6$ | $1.99 \cdot 10^7$ | $3.50 \cdot 10^6$ | – |
| | ADP | $\mathbf{3.94 \cdot 10^7}$ | $1.15 \cdot 10^6$ | $1.91 \cdot 10^7$ | $3.69 \cdot 10^6$ | 58.3 |
| *N*2 | HEM | $5.94 \cdot 10^7$ | $3.59 \cdot 10^6$ | $2.01 \cdot 10^7$ | $3.53 \cdot 10^6$ | – |
| | ADP | $\mathbf{5.29 \cdot 10^7}$ | $2.81 \cdot 10^6$ | $1.92 \cdot 10^7$ | $3.73 \cdot 10^6$ | 61.4 |
| *N*3 | HEM | $4.25 \cdot 10^7$ | $1.77 \cdot 10^6$ | $1.87 \cdot 10^7$ | $3.21 \cdot 10^6$ | – |
| | ADP | $\mathbf{3.79 \cdot 10^7}$ | $1.17 \cdot 10^6$ | $1.80 \cdot 10^7$ | $3.50 \cdot 10^6$ | 56.5 |
| *N*4 | HEM | $5.67 \cdot 10^7$ | $3.52 \cdot 10^6$ | $1.89 \cdot 10^7$ | $3.21 \cdot 10^6$ | – |
| | ADP | $\mathbf{5.15 \cdot 10^7}$ | $2.85 \cdot 10^6$ | $1.81 \cdot 10^7$ | $3.53 \cdot 10^6$ | 48.6 |

the following two trains. By rescheduling the timetables for T1, T2 and T3 with ADP algorithm, the traveling delays are reduced, and more importantly, the last train, namely T3, can be finally recovered to the original timetable, which enables the following trains to operate normally.

### *5.2.3. More experiments with different weight coefficients*

To investigate the performance of rescheduled timetables with different weight coefficients, we intend to implement more case studies in the following discussion. We take the instance of *N*2 as an example, for which the disturbance happens in segment Jiugong to Yizhuang qiao and its duration time is 150 s, and the basic weights are the same as the experiments in Section 5.2.2. We adjust the weigh coefficients for the delay time, traveling time and energy consumption orderly, where the weight ranges are set as $w_d \in [1, 21]$, $w_t \in [0.6, 2.6]$ and $w_e \in [1, 11]$. Note that the range of $w_d$, $w_t$ and $w_e$ are defined according to their order of magnitude. In what follows, we present the performance criteria variation curves by ADP and HEM with different values of $w_d$, $w_t$ and $w_e$ in Figs. 15, 16(a) to Fig. 16(d), and Fig. 16(e) to Fig. 16(h), respectively.

It can be seen from Fig. 15(a) and 15(b) that, the values of both passengers' time delay and traveling time by ADP are less than the results by HEM. This means that, by using the rescheduled timetable of ADP, the time delay and traveling time can be decreased in comparison to those of HEM in practice. And both the delay time and traveling time of ADP decrease obviously when $w_d$ is smaller than 7, and then keep relatively constant when $w_d$ is larger. This indicates that, the weight coefficents can influence the performance of the rescheduled timetable. Besides, for a given disturbance scenario, the rescheduled timetables might have a *"lower bound solution"*, which already achieves the minimum delay and traveling time subject to the rescheduling constraints. It can be seen from Fig. 15(c) that the variation of energy consumption takes an opposite tendency in comparison to that of the traveling time. This is just consistent with the practical experiences that the shorter traveling time will lead to higher energy consumption. Also, Fig. 15(d) shows the objective values of HEM and ADP with different $w_d$, where we can find an obvious phenomenon that the performance of ADP is better than that of HEM, which indicates its effectiveness. Correspondingly, the performance criteria variation with different value of $w_t$ follows the similar pattern in general from Fig. 16(a) to Fig. 16(d). That is, the two indicators, i.e., traveling time and time delay have a




(a) Scenario $N1$ (Jiugong - Yizhuang qiao, duration time = $100s$)

(b) Scenario $N2$ (Jiugong - Yizhuang qiao, duration time = $150s$)

(c) Scenario $N3$ (Wenhua yuan - Wanyuan, duration time = $100s$)

(d) Scenario $N4$ (Wenhua yuan - Wanyuan, duration time = $150s$)

**Fig. 14.** Rescheduled timetable for the four scenarios.

strong correlation. In essence, the traveling time contains the delay time, and the shorter delay time will potentially reduce the traveling time.

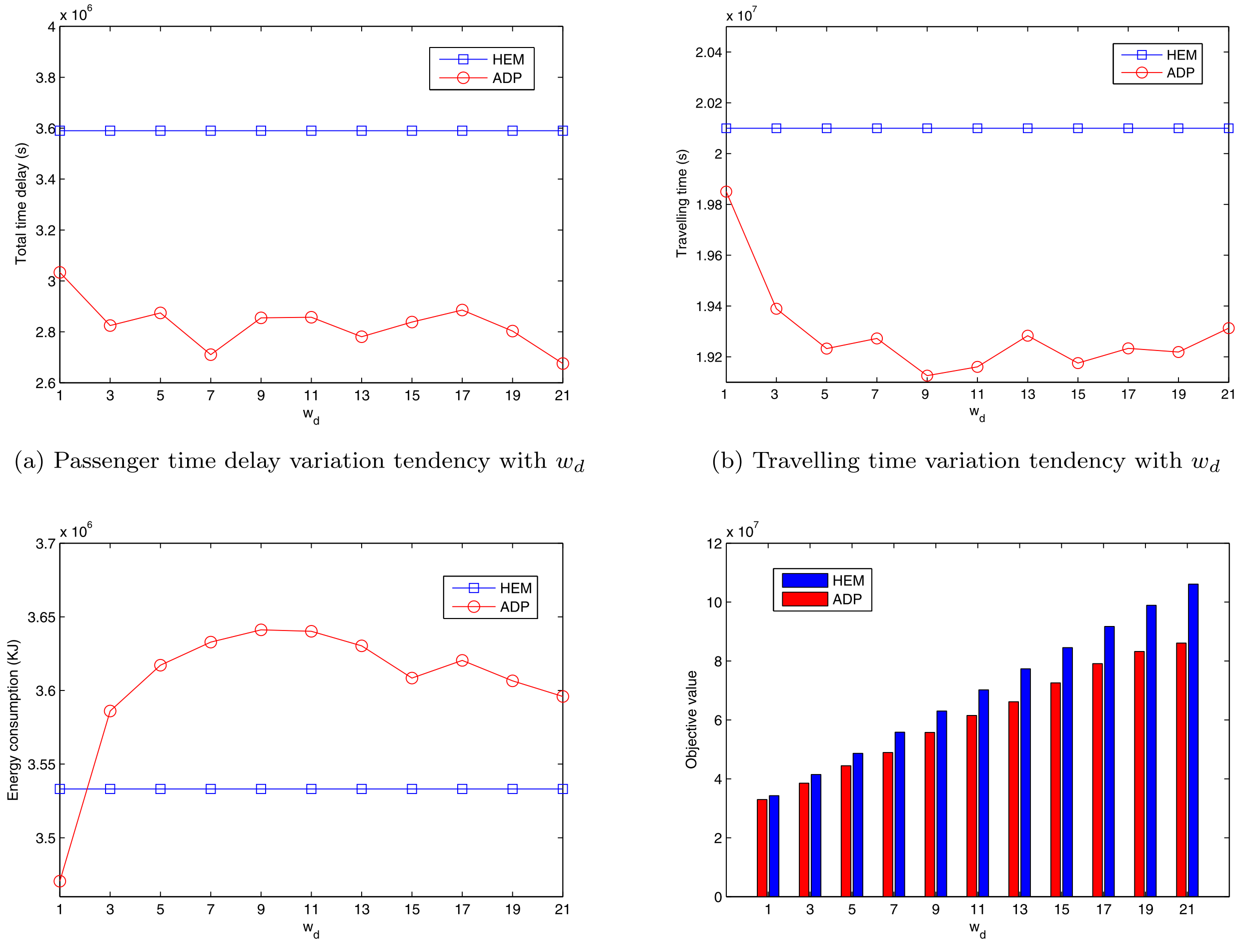
Fig. 16(e) to Fig. 16(h) show the performance criteria variation with different values of $w_e$. Different from the former two sets of experiments, the increase of $w_e$ in Fig. 16(e) and Fig. 16(f) makes the traveling time and delay time a little larger, even though they are still better than the performance of HEM. The energy-consumption decreases with the increase of weight factor $w_e$, as shown in Fig. 16(g). In addition, Fig. 16(h) depicts once again that the performance of rescheduled timetable by ADP is more effective than that of HEM in each case.

*5.2.4. More experiments with respect to larger disturbances*

In this case study, we shall consider more complex scenarios of metro train rescheduling with larger disturbances and more affected trains. In the experiments, the disturbance incident is derived from scenario $N2$, in which the duration times vary from 150 s to 360 s. In the following analysis, we use the metro operation data which are the same to those adopted in Example 5.2.1. The weight coefficients of $w_d$, $w_t$ and $w_e$ are set as 8.0, 1.0 and 3.0, respectively.

Since the disturbance duration times in the experiments are much larger, we note that, some other practical rescheduling strategies (e.g., increasing train running speed, adding a gap train or train station skipping, etc.) can be used besides of HEM to release transport pressure and reduce the travel time delay of passengers in metro systems (Schmocher et al., 2005). The rescheduling strategy by increasing train running speed is in essence distributing the delay time into each segment, which can reduce passenger delay time and traveling time, but might dramatically increase the energy consumption. This strategy can be only implemented in metro lines where the train control system has a slack trip time (e.g., in Moscow metro, Beijing subway) that enables the trains to run faster with higher speed and arrive at the next station earlier for about 2–8 s. Since this study assumes that the rolling stock plan is not changed and stop-skipping strategy is prohibited, we further add the method by decreasing the train running time averagely on each segment, which is termed as ETR in the following discussion. Thus, we use both HEM and ETR as the benchmarks in order to test the robustness of ADP algorithm for eight cases with increased disturbances. The computational results are listed in Table 9.

It is easy to see from the computational results that, when the delay duration time $t_d$ is increased, we have to add more rescheduled trains to recover from the disturbances for both ETR and ADP, while HEM needs to reschedule all the following trains. To constitute the same benchmark for comparison, we note that HEM only calculates the performance indicators of the same number of trains that are to be rescheduled by ETR and ADP. And it can be seen that, as train number and delay duration time increase, the performance indicators, i.e., delay time, traveling time and energy consumption increase for all




(a) Passenger time delay variation tendency with $w_d$

(b) Travelling time variation tendency with $w_d$

(c) Energy-consumption variation tendency with $w_d$

(d) Objective value variation tendency with $w_d$

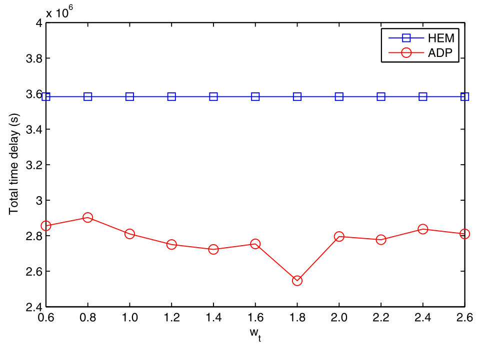
**Fig. 15.** Performance criteria variations with different $w_d$.

these three methods. This indicates that, larger disturbances will inevitably cause more affected passengers and longer time delay. The performance indicators of these three methods in the eight instances show that, ETR and ADP evidently reduce passengers' time delay and their traveling time compared with HEM, although both ETR and ADP slightly increase the energy consumption. This is consistent with practical experiences since these two methods decrease the train running time in each segment, which will inevitably increase energy consumption. Moreover, ADP and ETR achieve the similar performances in passengers' delay time and traveling time, while ADP consumes less energy than ETR, leading to the better objective value of ADP. This demonstrates that, ADP is able to reduce the delay for passengers while keeping the initial energy consumption as much as possible. In addition, the experiments with more rescheduled trains show that, the computational time of ADP is short enough (even when we reschedule 8 trains) to be applied in real-world applications.

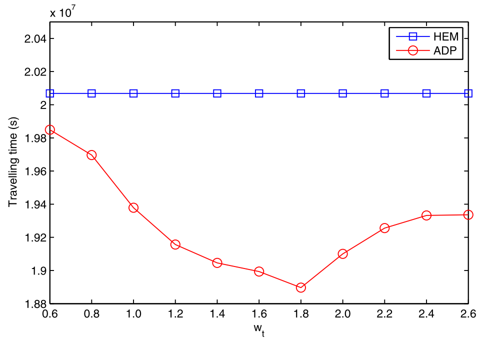
## 6. Conclusion

In this paper, we studied the real-time metro train rescheduling problem in an uncertain environment with disturbances to improve management service level and energy efficiency. Considering the uncertain and dynamic characteristics of passenger demands, we developed a stochastic programming model for train rescheduling that jointly optimizes the time delay of passengers, their traveling time and energy consumption. To capture the complexity and real-time requirements for the rescheduling problem, this paper proposed an efficient train rescheduling algorithm based on approximate dynamic programming, for which the expected future costs of trains delayed at stations are associated with the objective values, forming into an aggregation of approximated linear functions.

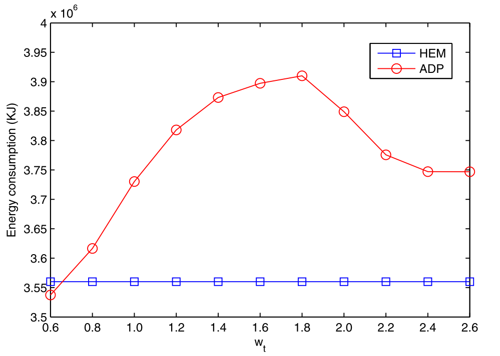
Extensive numerical experiments were implemented on two different cases, i.e., a small simulated metro line and a real-world instance of Beijing Metro Yizhuang Line to demonstrate the performance of the proposed approaches. The computational results showed that, by real-time rescheduling the timetable for affected trains, the ADP approach can effectively



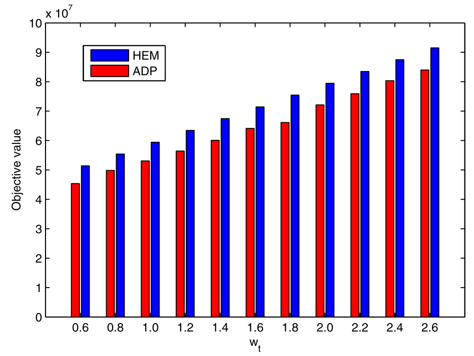

(a) Passenger time delay variation tendency with $w_t$



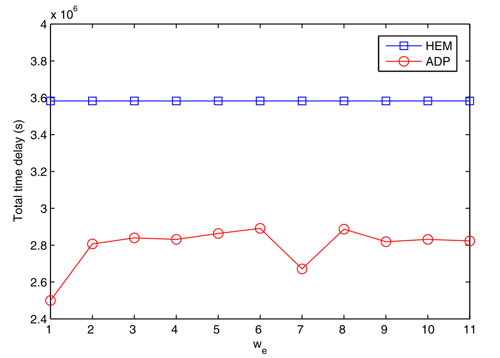

(b) Travelling time variation tendency with $w_t$



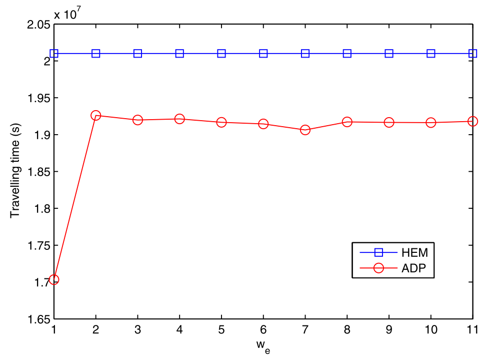

(c) Energy-consumption variation tendency with $w_t$



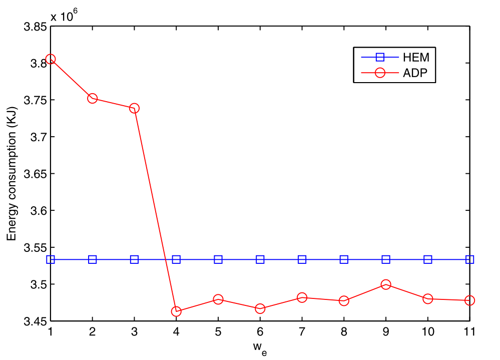

(d) Objective value variation tendency with $w_t$



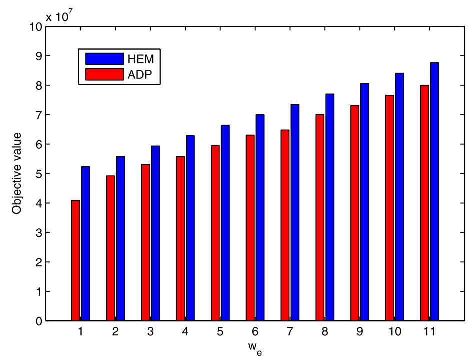

(e) Passenger time delay variation tendency with $w_e$




(f) Travelling time variation tendency with $w_e$




(g) Energy-consumption variation tendency with $w_e$




(h) Objective value variation tendency with $w_e$

**Fig. 16.** Performance criteria variation with different value of $w_t$ and $w_e$.

**Table 9**
Computational results by HEM, ETR and ADP with larger disturbances.

| Instance index | Number of affected trains | Method | Best objective value | Total delay time ($s$) | Traveling time (s) | Operation costs (KJ) | Computation time(s) |
|---|---|---|---|---|---|---|---|
| 1 | All | HEM | $5.94 \cdot 10^7$ | $3.59 \cdot 10^6$ | $2.01 \cdot 10^7$ | $3.53 \cdot 10^6$ | – |
| $t_d = 150$ | 3 | ETR | $5.45 \cdot 10^7$ | $2.83 \cdot 10^6$ | $1.97 \cdot 10^7$ | $4.07 \cdot 10^6$ | – |
| | 3 | ADP | $\mathbf{5.29 \cdot 10^7}$ | $2.81 \cdot 10^6$ | $1.92 \cdot 10^7$ | $3.73 \cdot 10^6$ | 61.4 |
| 2 | All | HEM | $7.44 \cdot 10^7$ | $4.22 \cdot 10^6$ | $2.68 \cdot 10^7$ | $4.60 \cdot 10^6$ | – |
| $t_d = 180$ | 4 | ETR | $7.06 \cdot 10^7$ | $3.54 \cdot 10^6$ | $2.64 \cdot 10^7$ | $5.30 \cdot 10^6$ | – |
| | 4 | ADP | $\mathbf{6.87 \cdot 10^7}$ | $3.53 \cdot 10^6$ | $2.56 \cdot 10^7$ | $4.97 \cdot 10^6$ | 87.5 |
| 3 | All | HEM | $8.82 \cdot 10^7$ | $4.74 \cdot 10^6$ | $3.33 \cdot 10^7$ | $5.67 \cdot 10^6$ | – |
| $t_d = 210$ | 5 | ETR | $8.70 \cdot 10^7$ | $4.31 \cdot 10^6$ | $3.28 \cdot 10^7$ | $6.57 \cdot 10^6$ | – |
| | 5 | ADP | $\mathbf{8.40 \cdot 10^7}$ | $4.19 \cdot 10^6$ | $3.19 \cdot 10^7$ | $6.20 \cdot 10^6$ | 100.6 |
| 4 | All | HEM | $9.54 \cdot 10^7$ | $5.62 \cdot 10^6$ | $3.34 \cdot 10^7$ | $5.67 \cdot 10^6$ | – |
| $t_d = 240$ | 5 | ETR | $9.40 \cdot 10^7$ | $5.18 \cdot 10^6$ | $3.29 \cdot 10^7$ | $6.57 \cdot 10^6$ | – |
| | 5 | ADP | $\mathbf{9.08 \cdot 10^7}$ | $5.01 \cdot 10^6$ | $3.20 \cdot 10^7$ | $6.23 \cdot 10^6$ | 96.6 |
| 5 | All | HEM | $12.1 \cdot 10^7$ | $7.62 \cdot 10^6$ | $3.98 \cdot 10^7$ | $6.73 \cdot 10^6$ | – |
| $t_d = 270$ | 6 | ETR | $11.9 \cdot 10^7$ | $7.02 \cdot 10^6$ | $3.92 \cdot 10^7$ | $7.83 \cdot 10^6$ | - |
| | 6 | ADP | $\mathbf{11.5 \cdot 10^7}$ | $6.89 \cdot 10^6$ | $3.82 \cdot 10^7$ | $7.20 \cdot 10^6$ | 101.8 |
| 6 | All | HEM | $14.9 \cdot 10^7$ | $9.92 \cdot 10^6$ | $4.60 \cdot 10^7$ | $7.77 \cdot 10^6$ | – |
| $t_d = 300$ | 7 | ETR | $14.6 \cdot 10^7$ | $9.21 \cdot 10^6$ | $4.53 \cdot 10^7$ | $8.97 \cdot 10^6$ | – |
| | 7 | ADP | $\mathbf{14.2 \cdot 10^7}$ | $9.14 \cdot 10^6$ | $4.42 \cdot 10^7$ | $8.40 \cdot 10^6$ | 107.9 |
| 7 | All | HEM | $15.8 \cdot 10^7$ | $11.1 \cdot 10^6$ | $4.62 \cdot 10^7$ | $7.77 \cdot 10^6$ | - |
| $t_d = 330$ | 7 | ETR | $15.5 \cdot 10^7$ | $10.2 \cdot 10^6$ | $4.54 \cdot 10^7$ | $9.17 \cdot 10^6$ | - |
| | 7 | ADP | $\mathbf{15.2 \cdot 10^7}$ | $10.3 \cdot 10^6$ | $4.45 \cdot 10^7$ | $8.43 \cdot 10^6$ | 109.2 |
| 8 | All | HEM | $19.0 \cdot 10^7$ | $13.9 \cdot 10^6$ | $5.23 \cdot 10^7$ | $8.80 \cdot 10^6$ | – |
| $t_d = 360$ | 8 | ETR | $18.5 \cdot 10^7$ | $12.9 \cdot 10^6$ | $5.15 \cdot 10^7$ | $1.02 \cdot 10^7$ | - |
| | 8 | ADP | $\mathbf{18.1 \cdot 10^7}$ | $12.9 \cdot 10^6$ | $5.05 \cdot 10^7$ | $9.07 \cdot 10^6$ | 114.7 |

**Table 10**
Numerical characteristics of different segments on Yizhuang metro line.

| Station number & segment | Length (m) | Dwelling time (s) | Running time (s) | MPAR (per minute) | Speed limit (m/s) |
|---|---|---|---|---|---|
| 1 Songjia zhuang - Xiao cun | 2641 | 30 | 190 | 6.83 | 22.22 |
| 2 Xiao cun - Xiaohong men | 1337 | 30 | 108 | 13.57 | 20.33 |
| 3 Xiaohong men - Jiugong | 2377 | 30 | 157 | 5.70 | 22.22 |
| 4 Jiugong - Yizhuang qiao | 1993 | 30 | 135 | 17.70 | 22.22 |
| 5 Yizhuang qiao - Wenhua yuan | 998 | 35 | 90 | 7.97 | 22.22 |
| 6 Wenhua yuan - Wanyuan | 1543 | 30 | 114 | 4.97 | 22.22 |
| 7 Wanyuan - Rongjing | 1285 | 30 | 103 | 4.90 | 22.22 |
| 8 Rongjing - Rongchang | 1360 | 30 | 104 | 3.50 | 22.20 |
| 9 Rongchang - Tongjinan | 2348 | 30 | 164 | 4.47 | 22.20 |
| 10 Tongjinan - Jinghai | 2274 | 30 | 150 | 5.60 | 22.22 |
| 11 Jinghai - Ciqunan | 2096 | 30 | 140 | 0.86 | 22.22 |
| 12 Ciqunan - Ciqu | 1290 | 35 | 102 | 0 | 22.20 |
| 13 Turnaround | – | 30 | – | – | – |
| 14 Ciqu - Ciqunan | 1290 | 35 | 102 | 2.00 | 22.20 |
| 15 Ciqunan - Jinghai | 2096 | 30 | 140 | 8.53 | 22.22 |
| 16 Jinghai - Tongjinan | 2274 | 30 | 150 | 25.13 | 22.22 |
| 17 Tongjinan - Rongchang | 2348 | 30 | 164 | 11.93 | 22.20 |
| 18 Rongchang - Rongjing | 1360 | 30 | 104 | 6.10 | 22.20 |
| 19 Rongjing - Wanyuan | 1285 | 30 | 103 | 5.90 | 22.22 |
| 20 Wanyuan - Wenhua yuan | 1543 | 30 | 114 | 4.16 | 22.22 |
| 21 Wenhua yuan - Yizhuang qiao | 998 | 35 | 90 | 4.53 | 22.22 |
| 22 Yizhuang qiao - Jiugong | 1993 | 30 | 135 | 6.63 | 22.22 |
| 23 Jiugong - Xiaohong men | 2377 | 30 | 157 | 1.26 | 22.22 |
| 24 Xiaohong men - Xiao cun | 1337 | 30 | 108 | 1.37 | 20.33 |
| 25 Xiao cun - Songjia zhuang | 2641 | 30 | 190 | 0 | 22.22 |
| 26 Terminal | – | 30 | – | – | – |

reduce the passengers' time delay around 10%-30% compared with a commonly used heuristic method. Moreover, the computational time for generating a near-optimal solution by the ADP method was only about $1-2$ min, which satisfies the time requirement for metro train rescheduling, making it appropriate for real-time implementations.

Our future research will focus on the following two major aspects. (1) This research mainly deals with the metro rescheduling problem with some relatively small disturbances that can be handled by adjusting the arrival and departure times of affected trains. Nevertheless, the large disruptions (e.g., fire disaster, etc.) may also occur in a metro system, which will cause a complete blockage and cancellation of seriously affected trains. Thus, one of our further research directions is to reschedule trains in more complex situations by considering the rerouting and cancelling of trains. (2) In this paper, we do not consider the stop-skip strategies or passenger interventions, which are both alternative approaches to recover from disturbances in urban metro systems. In the future study, we shall also focus on an integrated approach for the real-time metro rescheduling problem.

**Acknowledgement**

This research was supported by the National Natural Science Foundation of China (Nos. 71422002, U1434209, 71271020), and National Basic Research Program of China (No. 2012CB725400).

## Appendix A. The simplified deterministic train rescheduling model

We illustrate that the proposed stochastic model in Eq. (19) can be simplified into a mix-integer quadratic programming (MIQP) model under deterministic situations (i.e., the origin-to-destination passenger demands are considered to be deterministic). In the rescheduling problem, the objective function implies optimizing the delay time of passengers, their traveling time, and energy-consumption. Now we show that the objective function is a quadratic equation in condition that trains have unlimited capacity.

First, according to the literature (Cucala et al., 2012), the energy consumption is directly related to the train running time and the length of each segment. We assume that the regenerative braking energy is not considered, then the energy-consumption can be approximated as a linear piecewise curve between two adjacent stations for train $k$ in segment $i$, i.e.,

$$E_{k,i} = \big[\chi_x(\widehat{a}_{k,i+1} - \widehat{d}_{k,i}) + \chi_y\big]s_i, \quad 1 \le i < 2I, \quad i \ne I, \quad 1 \le k \le K, \tag{44}$$

where $\chi_x$ and $\chi_y$ are the corresponding parameters of the linear function. Consequently, the total energy consumption can be approximately calculated by

$$E_{total} = E_{kI} + E_{k(2I)} + \sum_{1\le k\le K}\sum_{1\le i<2I, i\ne I}\{[\chi_x(\widehat{a}_{k,i+1} - \widehat{d}_{k,i}) + \chi_y]s_i\}, \tag{45}$$

which is a linear function with given values of parameters $\chi_x$ and $\chi_y$. Then, we consider the passengers' delay time, which is denoted by

$$\begin{aligned} T_{delay} &= \sum_{1\le k\le K}\sum_{1\le i\le 2I} N_{a,k,i}(\widehat{a}_{k,i} - \bar{a}_{k,i}) \\ &= \sum_{1\le k\le K}\sum_{1<i\le 2I}\left[(\widehat{a}_{k,i} - \bar{a}_{k,i})\sum_{r=1}^{i-1} N_{b,k,r,i}\right]. \end{aligned} \tag{46}$$

Since the train boarding capacity is not considered in this simplified model, indicating that all the waiting passengers can get on train $k$ for $1 \le k \le K$, we have $T^c_{k,i} = \widehat{d}_{k,i}$ at each station $i$ and

$$\begin{aligned} N_{b,k,r,i} &= N^{r,i}_{\widehat{d}_{k-1,r},\widehat{d}_{k,r}} \\ &= \tau_{r,i}(\widehat{d}_{k,r} - \widehat{d}_{k-1,r}), \end{aligned} \tag{47}$$

in which $N^{r,i}_{\widehat{d}_{k-1,r},\widehat{d}_{k,r}}$ represents the number of arriving passengers at station $r$ with destination $i$ in time window $[\widehat{d}_{k-1,r}, \widehat{d}_{k,r}]$, and we denote $\widehat{d}_{0,i}$ to represent the departure time of the former train that is not affected by the disturbance, and thus, $\widehat{d}_{0,i}$ for each $i \in \mathcal{I}$ is a constant value that can be given by the original timetable. Based on Eqs. (46) and (47), the delay time of passengers can be further formulated as

$$T_{delay} = \sum_{k\in\mathcal{K}}\sum_{1<i\le 2I}\left\{(\widehat{a}_{k,i} - \bar{a}_{k,i})\sum_{r=1}^{i-1}\tau_{r,i}(\widehat{d}_{k,r} - \widehat{d}_{k-1,r})\right\}, \tag{48}$$

which implies that the time delay of passengers can be expressed as a quadratic equation, and the variables are the departure and arrival times of the rescheduled trains. Finally, we consider the representation of the traveling time of passengers, which is initially given as

$$T_{travel} = \sum_{k\in\mathcal{K}}\sum_{1\le i<2I}\{[N^v_k(\widehat{a}_{k,i}) - N_{a,k,i}](\widehat{d}_{k,i} - \widehat{a}_{k,i}) + N^v_k(\widehat{d}_{k,i})(\widehat{a}_{k,i+1} - \widehat{d}_{k,i})\}, \tag{49}$$

where $N^v_k(\widehat{a}_{k,i})$ and $N^v_k(\widehat{d}_{k,i})$ are the numbers of in-vehicle passengers of train $k$ when it arrives and departs from station $i$, respectively. Note that, for $\forall i > 1$, $N^v_k(\widehat{a}_{k,i})$ can be expressed as follows

$$\begin{aligned} N^v_k(\widehat{a}_{k,i}) &= N^v_k(\widehat{a}_{k,i-1}) + N_{b,k,i-1} - N_{a,k,i-1} \\ &= \cdots \end{aligned}$$

$$= N_k^{\nu}(\widehat{a}_{k,1}) + \sum_{1\le r<i}(N_{b,k,r} - N_{a,k,r}). \tag{50}$$

Since there is no passengers in the train when it arrives at the first station, i.e., $N_k^{\nu}(\widehat{a}_{k,1}) = 0$, we have

$$N_k^{\nu}(\widehat{a}_{k,i}) = \sum_{1\le r<i}(N_{b,k,r} - N_{a,k,r}), \tag{51}$$

for each $i > 1$. According to the derivations for passenger alighting and boarding processes, for each $1 \le k \le K$ and $1 < i < 2I$, we have

$$\begin{aligned} N_k^{\nu}(\widehat{a}_{k,i}) &= \sum_{1\le r<i}(N_{b,k,r} - N_{a,k,r}) \\ &= \sum_{1\le r<i}\left(N_{b,k,r} - \sum_{\phi=0}^{r-1} N_{b,k,\phi,r}\right) \\ &= \sum_{1\le r<i}\left\{\lambda_r(\widehat{d}_{k,r} - \widehat{d}_{k-1,r}) - \sum_{\phi=0}^{r-1}\left[\tau_{\phi,r}(\widehat{d}_{k,\phi} - \widehat{d}_{k-1,\phi})\right]\right\}, \end{aligned} \tag{52}$$

where $\tau_{\phi=0,r}$ and $\widehat{d}_{k,\phi=0}$ are given as 0. Correspondingly, for each $1 \le k \le K$ and $1 \le i < 2I$, we have

$$\begin{aligned} N_k^{\nu}(\widehat{d}_{k,i}) &= N_k^{\nu}(\widehat{a}_{k,i}) + N_{b,k,i} - N_{a,k,i} \\ &= \sum_{1\le r<i+1}\left\{\lambda_r(\widehat{d}_{k,r} - \widehat{d}_{k-1,r}) - \sum_{\phi=0}^{r-1}\left[\tau_{\phi,r}(\widehat{d}_{k,\phi} - \widehat{d}_{k-1,\phi})\right]\right\}. \end{aligned} \tag{53}$$

Therefore, the total traveling time of passengers can be formulated as

$$\begin{aligned} T_{travel} &= \sum_{k\in\mathcal{K}}\sum_{1\le i<2I}\left\{(\widehat{d}_{k,i} - \widehat{a}_{k,i})\left[N_k^{\nu}(\widehat{a}_{k,i}) - N_{a,k,i}\right] + (\widehat{a}_{k,i+1} - \widehat{d}_{k,i})N_k^{\nu}(\widehat{d}_{k,i})\right\} \\ &= \sum_{k\in\mathcal{K}}\sum_{1\le i<2I}\left\{(\widehat{d}_{k,i} - \widehat{a}_{k,i})\left[N_k^{\nu}(\widehat{d}_{k,i}) - N_{b,k,i}\right] + (\widehat{a}_{k,i+1} - \widehat{d}_{k,i})N_k^{\nu}(\widehat{d}_{k,i})\right\} \\ &= \sum_{k\in\mathcal{K}}\sum_{1\le i<2I}\left[(\widehat{a}_{k,i+1} - \widehat{a}_{k,i})N_k^{\nu}(\widehat{d}_{k,i}) - N_{b,k,i}(\widehat{d}_{k,i} - \widehat{a}_{k,i})\right] \\ &= \sum_{k\in\mathcal{K}}\sum_{1\le i<2I}\left\{(\widehat{a}_{k,i+1} - \widehat{a}_{k,i})\sum_{1\le r<i+1}\left\{\lambda_r(\widehat{d}_{k,r} - \widehat{d}_{k-1,r}) - \sum_{\phi=0}^{r-1}\left[\tau_{\phi,r}(\widehat{d}_{k,\phi} - \widehat{d}_{k-1,\phi})\right]\right\}\right. \\ &\qquad \left. - \lambda_i(\widehat{d}_{k,i} - \widehat{d}_{k-1,i})(\widehat{d}_{k,i} - \widehat{a}_{k,i})\right\}. \end{aligned} \tag{54}$$

Note that both $\tau_{\phi,\,r}$ and $\lambda_i$ are constant values with the assumption that the passenger arriving rates are deterministic and constant. Therefore, the objective function of the simplified model is a quadratic equation. Since the constraints are all linear inequations, we conclude that the model described in Eq. (19) is simplified into an MIQP model with deterministic and constant passenger demands.

## Appendix B. Basic operation data of Beijing Metro Yizhuang Line

See Table 10.

## References

Albrecht, A., Howlett, P., Peter, P., Vu, X., Zhou, P., 2015. The key principals of optimal train control-part 2: Existence of an optimal strategy, the local energy minimization principle, uniqueness, computational techniques. Transp. Res. Part B. In press, http://dx.doi.org/10.1016/j.trb.2015.07.024
Albrecht, A., Howlett, P., Peter, P., Vu, X., Zhou, P., 2015. The key principles of optimal train control-part 1: Formulation of the model, strategies of optimal type, evolutionary lines, location of optimal switching points. Transp. Res. Part B. In press, http://dx.doi.org/10.1016/j.trb.2015.07.023
Barrena, E., Canca, D., Coelho, L.C., Laporte, G., 2014. Single-line rail transit timetabling under dynamic passenger demand. Trans. Res. Part B 70, 134–150.
Bellman, R., 1954. The theory of dynamic programming. Bulletin of Am. Math. Soc. 60 (6), 503–516.
Bertsekas, D., 1995. Dynamic programming and optimal control. MA: Athena Scientific.
Bouzaiene-Ayari, B., Cheng, C., Das, S., Fiorillo, R., Powell, W., 2016. From single commodity to multiattribute models for locomotive optimization: a comparison of optimal integer programming and approximate dynamic programming. Transp. Sci. 50 (2), 366–389.
Cacchiani, V., Furini, F., Kidd, M.P., 2016. Approaches to a real-world train timetabling problem in a railway node. Omega 58, 97–110.
Cacchiani, V., Huisman, D., Kidd, M., Kroon, L., Toth, P., Veelenturf, L., Wagenaar, J., 2014. An overview of recovery models and algorithms for real-time railway rescheduling. Transp. Res. Part B 63, 15–37.
Cacchiani, V., Toth, P., 2012. Nominal and robust train timetabling problem. Eur. J. Operat. Res. 219, 727–737.
Caprara, A., Fischetti, M., Toth, P., 2002. Modeling and solving the train timetabling problem. Operat. Res. 50 (5), 851–861.

Corman, F., D'Ariano, A., Pacciarelli, D., Pranzo, M., 2012. Optimal inter-area coordination of train rescheduling decisions. Transp. Res. Part E 48, 71–88.
Corman, F., Quaglietta, E., 2014. Closing the loop in real-time railway control: framework design and impacts on operations. Transp. Res. Part C 54, 15–39.
Cucala, A.P., Fernandez, A., Sicre, C., Dominguez, M., 2012. Fuzzy optimal schedule of high speed train operation to minimize energy consumption with uncertain delays and driver's behavioral response. Eng. Appl. Artificial Intell. 25 (8), 1548–1557.
D'Ariano, A., 2009. Innovative decision support system for railway traffic control. IEEE Trans. Intell. Transp. Syst. Mag. 1 (4), 8–16.
D'Ariano, A., Pacciarelli, D., Pranzo, M., 2007. A branch and bound algorithm for scheduling trains in a railway network. Transp. Res. Part B 184 (2), 643–657.
Dorfman, M.J., Medanic, J., 2004. Scheduling trains on a railway network using a discrete event model of railway traffic. Transp. Res. Part B 38, 81–98.
Dundar, S., Sahin, I., 2013. Train re-scheduling with genetic algorithms and artifical neural networks for single-track railways. Transp. Res. Part C 27, 1–15.
Fang, J., Zhao, L., Fransoo, J.C., Woensel, T.V., 2013. Sourcing strategies in supply risk management: an approximate dynamic programming approach. Comput. Operat. Res. 40, 1371–1382.
Hadas, Y., Ceder, A., 2010. Optimal coordination of public-transit vehicles using operational tactics examined by simulation. Transp. Res. Part C 18, 879–895.
Howlett, P.G., Pudney, P.J., 1995. Energy-efficient train control. Springer-Verlag, Berlin.
Howlett, P.G., Pudney, P.J., Vu, X., 2009. Local energy minimization in optimal train control. Automatica 45 (11), 2692–2698.
Huang, Y., Yang, L., Tang, T., Cao, F., Gao, Z., 2016. Saving energy and improving service quality: bicriteria train scheduling in urban rail transit systems. IEEE Trans. Intell. Transp. Syst.. In press, http://dx.doi.org/10.1109/TITS.2016.2549282
Jiang, D.R., Powell, W.B., 2015. An approximate dynamic programming algorithm for monotone value functions. Operat. Res. 63 (6), 1526–5463.
Karvonen, H., Aaltonen, I., Wahlstrom, M., Salo, L., Savioja, P., Norros, L., 2011. Hidden roles of the train driver: a challenge for metro automation. Interacting Comput. 23, 289–298.
Khmelnitsky, E., 2000. On an optimal control problem of train operation. IEEE Trans. on Autom. Control 45 (7), 1257–1266.
Lee, Y., Chen, C.Y., 2009. A heuristic for the train pathing and timetabling problem. Transp. Res. Part B 43, 837–851.
Li, S., Schutter, B.D., Yang, L., Gao, Z., 2016. Robust model predictive control for train regulation in underground railway transportation. IEEE Trans. Contr. Syst. Technol. 24 (3), 1075–1083.
Li, X., Lo, H., 2014. An energy-efficient scheduling and speed control approach for metro rail operations. Transp. Res. Part B 64, 73–89.
Liu, R., Golovitcher, I.M., 2003. Energy-efficient operation of rail vehicles. Transp. Res. Part A 37 (10), 917–932.
Louwerse, I., Huisman, D., 2014. Adjusting a railway timetable in case of partial or complete blockades. Eur. J. Operat. Res. 235, 583–593.
Mahmoudi, M., Zhou, X., 2016. Finding optimal solutions for vehicle routing problem with pickup and delivery services with time windows: a dynamic programming approach based on state-space-time network representations. Transp. Res. Part B 89, 19–42.
Medury, A., Madanat, S., 2013. Incorporating network considerations into pavement management systems: a case for approximate dynamic programming. Transp. Res. Part C 33, 134–150.
Meng, L., Zhou, X., 2011. Robust single-track train dispatching model under a dynamic and stochastic environment: a scenario-based rolling horizon solution approach. Transp. Res. Part B 45, 1080–1102.
Meng, L., Zhou, X., 2014. Simultaneous train rerouting and rescheduling on an n-track network: a model reformulation with network-based cumulative flow variables. Transp. Res. Part B 67, 208–234.
Mu, S., Dessouky, M., 2013. Efficient dispatching rules on double tracks with heterogeneous train traffic. Transp. Res. Part B 51, 45–64.
Niu, H., Tian, X., Zhou, X., 2015. Demand-driven train schedule synchronization for high-speed rail lines. IEEE Trans. Intell. Transp. Syst. 16 (5), 2642–2652.
Niu, H., Zhou, X., 2013. Optimizing urban rail timetable under time-dependent demand and oversaturated conditions. Transp. Res. Part C 36, 212–230.
Niu, H., Zhou, X., Gao, R., 2015. Train scheduling for minimizing passenger waiting time with time-dependent demand and skip-stop patterns: nonlinear integer programming models with linear constraints. Transp. Res. Part B 76, 117–135.
Papagergiou, D.J., Cheon, M., Nemhauser, G., Sokol, J., 2015. Approximate dynamic programming for a class of long-horizon maritime inventory routing problems. Transp. Sci. 49 (4), 870–885.
Powell, W.B., 2007. Approximate dynamic programming: solving the curses of dimensionality. John Wiley & Sons, Hoboken, NJ, USA.
Powell, W.B., 2014. Clearing the jungle of stochastic optimization. INFORMS Tutorials in Operat. Res. 109–137.
Ross, S.M., 2014. Introduction to probability models, 10th Academic Press, Inc., Orlando, FL, USA.
Schmid, V., 2012. Solving the dynamic ambulance relocation and dispatching problem using approximate dynamic programming. Eur. J. Operat. Res. 219, 611–621.
Schmocher, J., Cooper, S., Adeney, W., 2005. Metro service delay recovery, comparison of strategies and constraints across systems. Trans. Res. Record 1930, 30–37.
Schutz, H., Kolisch, R., 2012. Approximate dynamic programming for capacity allocation in the service industry. Eur. J. Operat. Res. 218, 239–250.
Sun, L., Jin, J.G., Lee, D., Axhausen, K.W., Erath, A., 2014. Demand-driven timetable design for metro services. Trans. Res. Part C 46, 284–299.
Sutton, R.S., Barto, A.G., 1998. Reinforcement learning: An introduction. MIT Press.
Tornquist, J., 2012. Design of an effective algorithm for fast response to the re-scheduling of railway traffic during disturbances. Trans. Res. Part C 20, 62–78.
Veelenturf, L., Kidd, M., Cacchiani, V., Kroon, L., Toth, P., 2015. A railway timetable rescheduling approach for handling large scale disruptions. Transp. Sci.. In press, http://dx.doi.org/10.1287/trsc.2015.0618
Wang, Y., De Schutter, B., van den, B.T.J.J., Ning, B., Tang, T., 2014. Efficient bilevel approach for urban rail transit operation with stop-skipping. IEEE Trans. Intell. Transp. Syst. 15 (6), 2658–2670.
Wang, Y., Tang, T., Ning, B., van den, B.T.J.J., De Schutter, B., 2015. Passenger-demands-oriented train scheduling for an urban rail transit network. Transp. Res. Part C 60, 1–23.
Wong, R., Yuen, T., Fung, K., Leung, J., 2008. Optimizing timetable synchronization for rail mass transit. Transp. Sci. 42 (1), 57–69.
Xu, X., Li, K., Yang, L., 2015. Scheduling heterogeneous train traffic on double tracks with efficient dispatching rules. Transp. Res. Part B 78, 364–384.
Xu, X., Li, K., Yang, L., 2016. Rescheduling metro trains by a discrete event model considering service balance performance. Appl. Math. Model. 40 (2), 1446–1466.
Xu, X., Liu, J., Li, H., Hu, J., 2014. Analysis of metro station capacity with the use of queueing theory. Transp. Res. Part C 38, 28–43.
Xu, X., Liu, J., Li, H., Jiang, M., 2016. Capacity-oriented passenger flow control under uncertain demand: algorithm development and real-world case study. Transp. Res. Part E 87, 130–148.
Yalcinkaya, O., Bayhan, G.M., 2009. Modelling and optimization of average travel time for a metro line by simulation and response surface methodology. Eur. J. Operat. Res. 196, 225–233.
Yang, L., Qi, J., Li, S., Gao, Y., 2015. Collaborative optimization for train scheduling and train stop planning on high-speed railways. Omega. In press, http://dx.doi.org/10.1016/j.omega.2015.11.003
Yang, L., Zhang, Y., Li, S., Gao, Y., 2016. A two-stage stochastic optimization model for the transfer activity choice in metro networks. Transp. Res. Part B 83, 271–297.
Yang, L., Zhou, X., Gao, Z., 2013. Rescheduling trains with scenario-based fuzzy recovery time representation of two-way double-track railways. Soft Comput. 17 (4), 605–616.
Yang, L., Zhou, X., Gao, Z., 2014. Credibility-based rescheduling model in a double-track railway network: a fuzzy reliable optimization approach. Omega 48, 75–93.
Yang, X., Ning, B., Li, X., Tang, T., 2013. A cooperative scheduling model for timetable optimization in subway systems. IEEE Trans. Intell. Transp. Syst. 14, 438–447.
Yin, J., Chen, D., Li, L., 2014. Intelligent train operation algorithms for subway by expert system and reinforcement learning. IEEE Trans. Intell. Transp. Syst. 15 (6), 1561–1571.

Yin, J., Chen, D., Yang, L., Tang, T., Ran, B., 2015. Efficient real-time train operation algorithms under uncertain passenger demand. IEEE Trans. Intell. Transp. Syst.. In press, http://dx.doi.org/10.1109/TITS.2015.2478403

Zhang, D., Adelman, D., 2009. An approximate dynamic programming approach to network revenue management with customer choice. Transp. Sci. 43 (3), 381–394.

Zhou, X., Zhong, M., 2007. Single-track train timetabling with guaranteed optimality: branch-and-bound algorithms with enhanced lower bounds. Transp. Res. Part B 41, 320–341.