Article:

Yuan, Y. orcid.org/0000-0002-9933-3926, Li, S., Liu, R. orcid.org/0000-0003-0627-3184 et al. (2 more authors) (2023) Decomposition and approximate dynamic programming approach to optimization of train timetable and skip-stop plan for metro networks. Transportation Research Part C: Emerging Technologies, 157. 104393. ISSN 0968-090X

https://doi.org/10.1016/j.trc.2023.104393

Decomposition and approximate dynamic programming approach to train timetable coordination with skip-stop strategies of metro networks

Shukai Li[a], Yin Yuan[a], Ronghui Liu[b], Lixing Yang[a], Ziyou Gao[a]

[a]*State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing, 100044, China*
[b]*Institute for Transport Studies, University of Leeds, Leeds LS2 9JT, U.K.*

Abstract

With increased service lines and stations in large urban rail networks, there are invariably large passenger flows that involve transfers between lines, and the passenger demand can vary significantly between stations and over time of the day. Carefully coordinating train timetables of different operating lines can help reduce transfer delays, which in turn reduces station crowding and improves overall service quality. Separately, skip-stop strategies are often deployed during the train operations in order to balance the train capacity according to passenger demand distribution. This paper explores the joint optimization problem timetable coordination and skip-stop strategies in the timetable design that aims to minimize the passenger waiting, transfer time and station crowding. The combined problem is formulated as a mixed integer nonlinear programming model. To effectively address the complexity of our model, a decomposition and approximate dynamic programming approach is designed to reformulate the original network-level problem into many small-scale subproblems, one for each operating line, to be solved quickly in a distributed manner. The effectiveness and practicability of the model and method are demonstrated on two case networks: a simple synthetic network of three metro lines and a real network based on Beijing Subway. The computational results illustrate that our proposed joint timetable coordination with skip-stop strategies method can effectively reduce passenger waiting time and station crowing, our proposed decomposition and approximate dynamic programming approach is also shown to perform more efficiently than traditional centralized heuristic algorithms, such as genetic algorithm and simulated annealing algorithm, especially for larger-scale networks.

Keywords: Urban metro networks; Timetable coordination optimization; Skip-stop pattern; Approximate dynamic programming

1. Introduction

Rail-based urban metro transit has become an efficient and sustainable form of urban public transport due to its large capacity, high speed, and low energy consumption (Mannino and Mascis, 2009; Cadarso and Marín, 2012; Lamorgese and Mannino, 2015; Xu and Ng, 2020). As metro networks expand, and more lines and stations are added to the system, so are growing transfer demands where passengers have to change lines to reach their destination stations. Coordinating arrival times of trains from different lines at transfer stations can greatly enhance the transfer experience and improve the service level (Dessouky et al., 2003;

*Corresponding author.
Email address: R.Liu@its.leeds.ac.uk (Ronghui Liu)

Fonseca et al., 2018). Reducing passengers waiting time for the transfer, also helps reduce station and platform crowding, which is a common problem in many large metro systems. Therefore, how to design an optimized timetable to ensure good coordination of transfers is essential for metro systems to improve operation efficiency and service quality.

Separately, as the number of lines and stations increase, so will be the variability in passenger demand distributions, in that the passenger flows between origin-destination stations are not evenly distributed across the network and lines. The traditional all-stop strategy, widely adopted by the majority of metro operators, is shown to be ineffective to meet the requirements of such variable demand (Wang et al., 2014; Jamili and Aghaee, 2015a). For instance, when a train arrives at a station with high passenger demand, some passengers may not be able to get on because of train capacity constraints, and have to wait for the next train. This happens regularly in large metro networks, such as Beijing Subway, especially during rush hours. The skip-stop strategy that allows trains to selectively skip some stations has been shown to be able to balance the capacity resources of trains and passenger demand (Freyss et al., 2013; Jamili and Aghaee, 2015a). Typically, the train would skip stations with relatively low passenger demand, in order to retain its capacity to accommodate passengers gathering at busy stations, therefore balance the distribution of waiting passengers and alleviate the overcrowding of those busy stations. There is scope, therefore, to incorporate skip-stop strategies into the train timetable coordination in order to further improve the level of service, safety and reliability of metro systems.

Joint optimization of transfer coordination and skip-stop strategies in metro networks is challenging, not only owing to the complex coupling interaction among transfer passengers, non-transfer passengers, and train capacities, but also the high computational complexity in dealing with realistic large-scale metro networks. Focusing on these challenges, this paper investigates the timetable coordination optimization with skip-stop strategies of metro networks, and develops a decomposition and approximate dynamic programming (ADP) approach for the purpose of efficiently obtaining reasonable train timetables and skip-stop strategies for multiple connected operating lines.

1.1. Literature review

Coordinating the arrival times of trains from different lines at a transfer station would reduce passenger transfer waiting time and improve the overall service level of a metro network. Given the predetermined number of transfer passengers, Wong et al. (2008) constructed a mixed-integer programming model for the timetable synchronization problem for rail transit networks to minimize the transfer waiting time. Similarly, Hu et al. (2022) also considered the predetermined transfer passengers to study the timetable coordination problem, and formulated a mixed-integer quadratic programming problem. To improve the worst transfers, Wu et al. (2015) studied the equity-based scheduling synchronization problem to reduce transfer waiting. The train scheduling problem for transitional periods was investigated by Guo et al. (2017) to increase the number of synchronization events. Li et al. (2018a) incorporated the transfer coordination constraints into their network-level train regulation model to improve the passenger transfer services. The above studies explored network-level train scheduling problems to improve the transfer performances, with unlimited train capacity and pre-given transfer demands in formulations. To adapt to a wider range of application scenarios, the interactions between passengers and the train capacity have been considered in the literature. Considering the train capacity, Wang et al. (2020) constructed timetable synchronization optimization problems, where their objectives are minimizing the waiting time and numbers of those failing in transferring. Similarly, the train capacity limitation and the dynamicity of passenger flows were also considered by Yin et al. (2021). They studied train timetable coordination problems to minimize station crowding. Han et al. (2021) additionally took into account the uncertainty of passenger demands, and constructed multi-scenario scheduling models from the network perspective.

As the metro networks expand and the number of lines and stations increase, the passenger demand distributions across the network become increasingly varied. To address the such imbalanced spatial distribution of passenger flows, train scheduling considering skip-stop patterns has been widely studied. However, most of the studies focus on single lines, and ignore the wider network impact of skip-stop on individual lines. With predetermined skip-stop strategies for metro lines, Niu et al. (2015) formulated a quadratic integer programming model to explore train timetable optimization issues for a metro line, with the objective of minimizing passenger waiting time. Considering flexible skip-stop patterns, Wang et al. (2014) constructed a mixed-integer nonlinear programming model to provide the optimized train timetables and skip-stop strategies for a metro loop line, aiming to simultaneously reduce passenger travel time and energy consumption. Focusing on congested railway double-track line, a integer linear programming model was formulated by Jiang et al. (2017) to address the collaborative optimization of train scheduling and skip-stop patterns, aiming to increase scheduled trains. Shang et al. (2018) investigated the equity-oriented skip-stop strategies to ensure that passengers have identical shares of train capacity for the oversaturated urban rail transit system. By combining the passenger flow control with skip-stopping strategies, Jiang et al. (2019) developed a mixed-integer nonlinear programming model to reduce the penalty value of stranded passengers for a congested metro line. Zhu and Goverde (2020) considered flexible skip-stop patterns and short-turning strategies for train rescheduling problems. Focusing on improving the service quality during a quick recovery of trains from disturbances, Chen et al. (2022) investigated the real-time skip-stop strategy to reduce the number of stranded passengers.

Compared to optimization problems for single lines, models developed to address large-scale network problems are usually highly complex, involving multiple lines and the interaction between them. The existing studies on network-level problems tend to adopt centralized approaches (Corman et al., 2012; Wu et al., 2015; Kang et al., 2015; Li et al., 2016; Han et al., 2021; Hu et al., 2022), which inevitably bring heavy computational burden. To reduce the computational complexity, decomposed and distributed approaches have been designed. For timetable coordination optimization problems, a decomposition based algorithm was shown in studies of Yin et al. (2021), performing well in realistic numerical examples. Based on a dual decomposition, Li et al. (2018b) designed a distributed approach for network-level train regulation problems. Besides, some effective methods are also used in other large-scale systems. Frey et al. (2017) designed a novel decomposition approach with column generation to deal with the planning of outbound baggage handling. Liu et al. (2020) adopted the Lagrangian decomposition approach for collaborative train scheduling optimization problems. Lamorgese and Mannino (2015) designed an exact decomposition method to realize the efficient dispatch of trains. Taherkhani et al. (2020) applied Benders decomposition approach to solve the capacitated hub location problems. To cope with the systems with complex characteristics, some ADP-based methods were adopted to quickly obtain high-quality solutions. Liu et al. (2018) utilized ADP method to generate energy-efficient train scheduling strategies. Papageorgiou et al. (2015) coped with maritime inventory routing problems. He et al. (2022) applied multi-stage look-ahead strategies to bus holding control problems.

1.2. The proposed approach

Table 1 summarizes the above relevant studies, and highlights the distinctive contributions of the proposed approach.

Table 1: Summary of different publications on train timetable optimization.

Publication	Infrastructure	Stop strategy	Train capacity limitation	Objective function	Methodology	Solution way
Wang et al. (2014)	Single line	Skip-stop	Yes	Passenger travel time, energy consumption	Bilevel approach	Centralized
Niu et al. (2015)	Single line	Skip-stop	Yes	Total waiting time	GAMS	Centralized
Jiang et al. (2019)	Single line	Skip-stop	Yes	Passengers being stranded	Q-learning algorithm	Centralized
Wong et al. (2008)	Network	All stop	No	Transfer waiting time	Optimization-based heuristic approach	Centralized
Yin et al. (2019)	Network	All stop	No	Waiting passengers	CPLEX	Centralized
Hu et al. (2022)	Network	All stop	No	Transfer waiting time	Benders decomposition	Centralized
Wang et al. (2020)	Network	All stop	Yes	Waiting time, passengers failing to transfer	GA & GWO	Centralized
Yin et al. (2021)	Network	All stop	Yes	Crowdedness of stations	CPLEX/VLNS	Decomposed
Han et al. (2021)	Network	All stop	Yes	Train service cost, waiting time, final stranded penalty	Genetic algorithm	Centralized
This paper	Network	Skip-stop	Yes	Total waiting time, station crowding	Decomposition and ADP approach	Distributed

This paper makes a meaningful investigation into the complicated issue concerning a distributed approach to timetable coordination for an entire metro network with skip-stop strategies. Specifically, two-fold contributions are highlighted:

(1) At present, existing studies on train timetable coordination problems for networks focus mainly on all-stop patterns (Wang et al., 2020; Yin et al., 2021; Han et al., 2021; Hu et al., 2022), and research on skip-stop strategies have only been solved as line-level problems (Wang et al., 2014; Niu et al., 2015; Jiang et al., 2019). Studies that jointly optimize skip-stop strategies and transfer coordination at a network level are found to be scarce. Besides, to avoid addressing the complex interaction between passengers and train capacity, existing network-level studies tend to adopt idealized assumptions, e.g., the train capacity is infinite and the number of transfer passengers is predetermined (Wong et al., 2008; Guo et al., 2017; Hu et al., 2022), which limits their practical applications. As a novel approach, this paper proposes a nonlinear programming formulation for train timetable coordination optimization problems with skip-stop strategies of metro networks, characterizing coupling interactions among transfer passengers, non-transfer passengers and train capacity.

(2) In existing research, the timetable optimization problem for networks is usually solved by centralized approaches (Yin et al., 2019; Wang et al., 2020; Han et al., 2021; Hu et al., 2022) with high computational complexity. In this study, a decomposition and ADP algorithm is designed to address this problem. Specifically, under a distributed coordinate descent scheme, the original complex nonconvex and nonlinear problem can be transformed into many small-scale line-level subproblems and solved using dynamic programming (DP) to achieve efficient solutions. In this way, not only parallelization can be realized, the subproblems are also equipped with desirable properties suitable to apply ADP methods to obtain efficient and high-quality solutions. Overall, we show that our proposed algorithm can address large-scale network-level timetable and operational optimization problems.

The remainder of our paper is organized as follows. Section 2 formulates a timetable coordination optimization model for metro networks. Section 3 presents the decomposition and ADP algorithm. Section

4 illustrates two sets of numerical experiments to show the effectiveness of our approaches. Then conclusions are given in Section 5.

2. Problem Description

We consider a metro network consisting of f uni-directional lines represented by the set $L = \{ l | \, l = 1, 2, ..., f\}$. There are m_l stations represented by the set $U^l = \{1, 2, ..., m_l\}$, and n_l operating trains denoted by the set $Q^l = \{q = 1, 2, ..., n_l\}$ on line l. We focus on the timetable coordination optimization problem with skip-stop strategies for the metro network, and take into account explicitly the following characteristics typical of large metro networks: (1) capacity constraints on the trains and at stations (and platforms), and the interactions between passenger demand and train services; (2) coordinating the arrival times and departure times of trains from different lines, so as to reduce passenger transfer delay and station crowding; (3) network-wide skip-stop strategies to balance the demand with capacity; and (4) balancing the needs of transfer and non-transfer passengers to maximize travel experience to all passengers. The focus is on a short-term during-the-operation scheduling strategy, devising dynamic train dispatching applications (on departure/arrival times at stations, and stopping patterns) over a horizon of one or two hours ahead, that adapts to predicted passenger demands in this time horizon. Such a dynamic operation optimization allows the metro operators to best respond to short-term changes due for example to emergency situations or to unusually weather conditions, whereby a computationally efficient solution is crucial.

To facilitate the analysis, some assumptions are listed as follows.

Assumption 2.1. *A train is not allowed to skip two adjacent stations, and a station is not skipped by two adjacent trains. Terminus and transfer stations are not to be skipped.*

Assumption 2.2. *Transfer walking times between different lines are known and fixed.*

Assumption 2.3. *The two directions of a physical line are considered as independent, where trains on the two lines are considered to be operating separately.*

Assumption 2.1 sets some certain constraints to the skip-stop pattern to ensure that the metro service will not be impacted too much, which has been also considered by Jiang et al. (2019). Assumption 2.2 is a widely used assumption for network-level train scheduling problems, and similarly to the existing researches (Wong et al., 2008; Wang et al., 2020), transfer walking times between lines could be obtained from the history data. Finally, for simplicity, the process of turnarounds for trains is not considered as Assumption 2.3, which also appears in studies of (Li et al., 2018b; Wang et al., 2020).

2.1. Notations and parameters

Some symbols and parameters in formulating the problem are given in Table 2.

2.2. Mathematical Formulations

2.2.1. Modelling the skip-stop strategies

In this paper, we design the skip-stop strategy dynamically and in cooperation with timetable optimization to serve a set of optimization objectives described in Section 2.2.4. We first introduce a binary variable $x_u^{l,q}$ to describe whether train q skips station u of line l, i.e.,

$$x_u^{l,q} = \begin{cases} 0 \text{ if train } q \text{ stops at station } u \text{ on line } l \\ 1 \text{ if train } q \text{ skips station } u \text{ on line } l \end{cases} , \quad \forall l \in L, u \in U^l, q \in Q^l$$

Table 2: Main symbols and parameters used in the model.

Notations	Definition
Sets	
L	the set of lines, $L = \{1, 2, ..., f\}$;
Q^l	the set of trains of line l, $Q^l = \{1, 2, ..., n_l\}$;
U^l	the set of stations of line l, $U^l = \{1, 2, ..., m_l\}$;
K^l	the set of transfer stations of line l;
$X_{l,u}$	the set of connecting lines with line l at station u;
System parameters	
$p_u^l(t)$	the number of passengers arriving at station u at time interval t on line l;
$e_l^{l'}$	the average walking time of transferring from line l' to line l;
C_l	the capacity of trains of line l;
$R_u^{l,q}$	the running time of train q from station u to $(u+1)$ of line l;
$H_{\min}^l$	the predetermined minimum value of headway on line l;
$S_{\min}^l$, $S_{\max}^l$	the lower and upper limit to dwell times of line l;
$\beta_u^{l',l}$	the proportion of the number of passengers who will transfer to line l, among the total number of those alighting from line l' at transfer station u;
System variables	
$A_u^{l,q}$	the arrival time of train q at station u of line l;
$D_u^{l,q}$	the departure time of train q from station u of line l;
$S_u^{l,q}$	the dwell time of train q at station u of line l;
$x_u^{l,q}$	the binary variable, if train q skip station u, $x_u^{l,q} = 1$; otherwise, it is 0;
$o_u^{l,q}$	the remaining capacity of train q at station u of line l;
$n_{l,u,q}^{wt}$	the number of passengers waiting for train q at station u of line l;
$n_{l,u,q}^{bt}$	the number of passengers actually getting on train q at station u of line l;
$n_{l,u,q}^{al}$	the number of alighting passengers from train q at station u of line l;
$n_{l,u,q}^{st}$	the number of stranded passengers at station u after departure of train q on line l;
$g_u^{l',l}$	the number of passengers transferring from line l' to l at transfer station u;
$\theta_{l,u,q}^{l',q'}$	the binary variable, if train q of line l and train q' of line l' constitute a possible connection at transfer station u, $\theta_{l,u,q}^{l',q'} = 1$; otherwise, it is 0.

Following Assumption 1, the following limits are set to the skip-stop strategy.

$$x_u^{l,q} + x_{u+1}^{l,q} \le 1, \quad \forall l \in L, u \in U^l, q \in Q^l \tag{1}$$

$$x_u^{l,q} + x_u^{l,q+1} \le 1, \quad \forall l \in L, u \in U^l, q \in Q^l \tag{2}$$

$$x_1^{l,q} + x_{m_l}^{l,q} + \sum_{u \in K^l} x_u^{l,q} = 0, \quad \forall l \in L, q \in Q^l \tag{3}$$

Constraint (1) claims that a train is not allowed to skip two adjacent stations, while constraint (2) indicates that a station will not be skipped by two adjacent trains. In addition, constraint (3) suggests that terminals and transfer stations are considered not to be skipped generally owing to the high passenger density.

2.2.2. Modelling the train traffic dynamics

Trains travel at a fixed speed (and with fixed travel time) between stations. Their dwell time is influenced by skip-stop strategies. If a station is skipped, the corresponding dwell time equals zero; otherwise, the dwell time is viewed as a dynamic decision variable constrained within a given range, which shall not be too long due to the operation efficiency, nor too short to ensure the safety of passengers getting on and off. Besides, the train departure headways are also constrained between a minimum and maximum range for the line owing to service and safety requirements. Constraints (4)-(7) illustrate the train traffic dynamics with skip-stop strategies.

$$D_u^{l,q} = A_u^{l,q} + S_u^{l,q}\left(1 - x_u^{l,q}\right), \quad \forall l \in L, u \in U^l, q \in Q^l \tag{4}$$

$$A_u^{l,q} = D_{u-1}^{l,q} + R_{u-1}^{l,q}, \quad \forall l \in L, u \in U^l, q \in Q^l \tag{5}$$

$$D_u^{l,q} - D_u^{l,q-1} \ge H_{\min}^l, \quad \forall l \in L, u \in U^l, q \in Q^l \tag{6}$$

$$S_{\min}^l \le S_u^{l,q} \le S_{\max}^l, \quad \forall l \in L, u \in U^l, q \in Q^l \tag{7}$$

Equation (4) defines the departure time $D_u^{l,q}$ of train q at station u of line l, where $S_u^{l,q}$ means the dwell time. Equation (5) formulates the dynamic transition equation of the arrival time, where $R_u^{l,q}$ presents the running time of train q from station u to $(u+1)$. Constraints (6) and (7) provide bound limitation to departure headway and dwell time for operational safety and service requirements, where $H_{\min}^l$ is the predetermined minimum headway, and $S_{\min}^l$ and $S_{\max}^l$ are the minimum and maximum dwell times of line l.

2.2.3. Modelling the passenger behaviours

In metro networks, passenger behaviours include arriving, waiting, boarding and alighting, where transferring is embodied in the process of alighting and arriving. Specifically, passengers begin to wait for trains after arriving at the station from the outside. If the train remaining capacity is sufficient, waiting passengers can board the train when it departures; otherwise, some of them have to be stranded and continue to wait for the next. For passengers on the train, they will alight when the train arrive at their destinations on the line. Among those alighting passengers, non-transfer passengers will leave the station, while transfer passengers will walk to the platform of connecting lines to wait again as the arriving passenger flow and continue to participate in the interaction with trains. Equations (8)-(12) describe the relationship between passengers and trains for metro networks.

$$n_{l,u,q}^{wt} = \sum_{t \in \left[D_u^{l,q-1}, D_u^{l,q}\right]} p_u^l(t) + \sum_{l' \in X_{l,u}, q' \in Q^{l'}} g_{l,u,q}^{l',q'} + n_{l,u,q-1}^{st}, \quad \forall l \in L, u \in U^l, q \in Q^l \tag{8}$$

$$n_{l,u,q}^{bt} = \min\left\{\left(1 - x_u^{l,q}\right) n_{l,u,q}^{wt}, o_u^{l,q}\right\}, \quad \forall l \in L, u \in U^l, q \in Q^l \tag{9}$$

$$n_{l,u,q}^{st} = n_{l,u,q}^{wt} - n_{l,u,q}^{bt}, \quad \forall l \in L, u \in S^l, q \in Q^l \tag{10}$$

$$o_u^{l,q} = o_{u-1}^{l,q} + n_{l,u,q}^{al} - n_{l,u-1,q}^{bt}, \quad \forall l \in L, u \in U^l, q \in Q^l \tag{11}$$

$$n_{l,u,q}^{al} = \alpha_u^{l,q}\left(1 - x_u^{l,q}\right)\left(C_l - o_u^{l,q} + n_{l,u,q}^{bt}\right), \quad \forall l \in L, u \in U^l, q \in Q^l \tag{12}$$

Equation (8) defines the number of waiting passengers. For non-transfer stations, $n_{l,u,q}^{wt}$ includes the passengers arriving at stations from the outside during the two consecutive departures of trains, and the stranded passengers left by the previous trains. For transfer stations, $n_{l,u,q}^{wt}$ also comprises the passengers transferring from other lines. In this equation, $p_u^l(t)$ indicates the number of the arriving passengers at station u at time interval t from the outside. $X_{l,u}$ denotes the set of connecting lines with line l at station u. If station u do not provide transfer services, $X_{l,u} = \emptyset$. $g_{l,u,q}^{l',q'}$ is the number of passengers transferring from train q' of line l' to train q of l at transfer station u, which is defined in (16). Equation (9) calculates the number of passengers $n_{l,u,q}^{bt}$ actually boarding train q at station u, which depends on the number of waiting passengers, whether the train skips the station, and the train remaining capacity. Equation (10) formulates the number of stranded passengers, which is equal to the number of waiting passengers minus the number of those getting on the train. Equation (11) specifies the dynamic transition of the remaining capacity $o_u^{l,q}$ for train q at station u. $o_u^{l,q}$ equals $o_{u-1}^{l,q}$ plus the alighting passengers at station u, and minus the boarding passengers at station $u-1$. To ensure the integrity of the formulation, we set $o_1^{l,q} = C_l$, for $l \in L, q \in Q^l$. The number of alighting passengers $n_{l,u,q}^{al}$ from train q at station u, is expressed in Equation (12), where $\alpha_u^{l,q}$ denotes the alighting ratio.

In the metro network, the transfer passenger flows depend on the transfer coordination between trains of connecting lines. Constraints (13)-(16) stipulate the train coordination and transfer passenger flows.

$$D_u^{l,q} - \left(A_u^{l',q'} + e_l^{l'}\right) > M\left(\theta_{l,u,q}^{l',q'} - 1\right), \quad \forall l \in L, u \in K^l, q \in Q^l, l' \in X_{l,u}, q' \in Q^{l'} \tag{13}$$

$$D_u^{l,q} - \left(A_u^{l',q'} + e_l^{l'}\right) \leq M\theta_{l,u,q}^{l',q'}, \quad \forall l \in L, u \in K^l, q \in Q^l, l' \in X_{l,u}, q' \in Q^{l'} \tag{14}$$

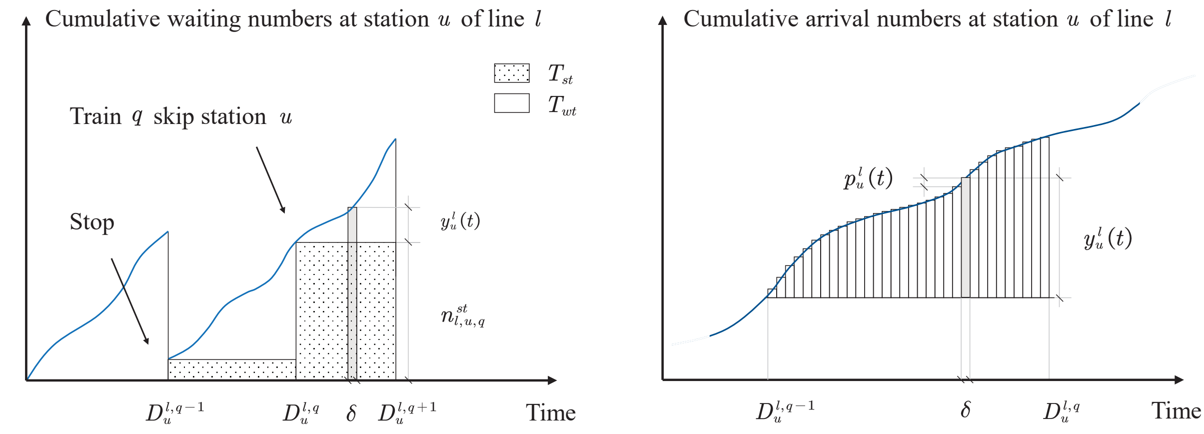
$$y_{l,u,q}^{l',q'} = \theta_{l,u,q}^{l',q'} - \theta_{l,u,q-1}^{l',q'}, \quad \forall l \in L, u \in K^l, q \in Q^l, l' \in X_{l,u}, q' \in Q^{l'} \tag{15}$$

$$g_{l,u,q}^{l',q'} = \beta_u^{l,q} y_{l,u,q}^{l',q'} n_{l',u,q'}^{al}, \quad \forall l \in L, u \in K^l, q \in Q^l, l' \in X_{l,u}, q' \in Q^{l'} \tag{16}$$

Constraints (13) and (14) define a binary variable $\theta_{l,u,q}^{l',q'}$ to indicate possible connections. M is a large positive number. If the departure time $D_u^{l,q}$ is later than the transfer passengers' entering time (i.e., $A_u^{l',q'} + e_l^{l'}$), these passengers from train q' of line l' will have opportunities to get on train q of line l, and thus the two trains constitute a possible connection. For transfer passengers, they will want to board the coming train nearest their entering time among all the connecting trains with possible connections with the train they get off. Equation (15) introduces $y_{l,u,q}^{l',q'}$ to indicate whether train q' and q constitute an effective connection, namely whether train q is the first train those transfer passengers alighting from train q' wait for. Based on the transfer connection relationships between trains of connecting lines, equation (16) is formulated to represent the number of passengers transferring from line l' to l at transfer station u, where $\beta_u^{l,q}$ denotes the transfer ratio.

2.2.4. Objective functions

Concerning both the efficiency and safety of metro operations, our objectives consist of three aspects: minimizing passenger waiting time, alleviating station crowding, and meanwhile reducing the penalty of skipping operations.



(a) Total waiting time at station u

(b) Cumulative arrival passengers

Figure 1: Illustration of the computation for the passenger waiting time for non-transfer stations.

Firstly, the passenger waiting time includes three parts: the waiting time for the new arrivals before the arrival of trains, the transfer waiting time if the station provides transfer services, and the waiting time for those who are left by previous trains and have to wait for the next.

Figure 1 (a) illustrates the two parts of the passenger waiting time at non-transfer stations, i.e., the waiting time T_{wt} for the new arrivals and the waiting time T_{st} for those passengers who failed to get on an previous train. Figure 1 (b) illustrates the cumulative number of those entering station u, through which, the cumulative number of the new arrivals at time interval t ($t \in \left[D_u^{l,q-1}, D_u^{l,q}\right]$) can be presented by

$$y_u^l(t) = \sum_{t' \in \left[D_u^{l,q-1}, t\right]} p_{l,u}(t'), \quad \forall l \in L, u \in U^l, t \in \left[D_u^{l,q-1}, D_u^{l,q}\right] \tag{17}$$

Thus, the passenger waiting time of the new arrivals for train q at station u of line l can be formulated by

$$w_u^{l,q} = \sum_{t \in \left[D_u^{l,q-1}, D_u^{l,q}\right]} \delta y_u^l(t), \quad \forall l \in L, u \in U^l, q \in Q^l \tag{18}$$

where δ denotes the length of the time interval t. Then the total waiting time T_{wt} for the new arrivals can be formulated as

$$T_{wt} = \sum_{l \in L, u \in U^l, q \in Q^l} w_u^{l,q} \tag{19}$$

The waiting time T_{st} for those stranded at stations due to train capacity limitation or the train skipping operation is formulated as

$$T_{st} = \sum_{l \in L, u \in U^l, q \in Q^l} n_{l,u,q}^{st} \left(D_u^{l,q+1} - D_u^{l,q} \right) \tag{20}$$

For transfer stations, the transfer waiting time is expressed as

$$T_{tf} = \sum_{l \in L, u \in K^l, q \in Q^l} \sum_{l' \in X_{l,u}, q' \in Q^{l'}} g_{l,u,q}^{l',q'} \left(D_u^{l,q} - A_u^{l',q'} - e_l^{l'} \right) \tag{21}$$

Notably, if trains q' of line l' and q of line l do not constitute an effective transfer connection, T_{tf} will be enforced to be zeros, since $g_{l,u,q}^{l',q'} = 0$, according to the definition of the number of transfer passengers $g_{l,u,q}^{l',q'}$ in equation (16). Besides, transfer passengers might fail to board the first coming train after they enter the transfer platform, due to the limited capacity of trains. In this case, they will be a part of $n_{l,u,q}^{st}$. Hence, the total transfer passenger waiting time can be comprised in the calculation of T_{tf} and T_{st}.

Secondly, waiting for 1 minute at a highly crowded station equals 1.7–2.5 minutes under medium-crowding conditions (Li and Hensher, 2011), and with the increase of station crowding, the probability of stampede or other accidents will increase rapidly (Jiang et al., 2019). Therefore, minimizing the station crowding is considered, which is expressed by the number of passengers F under crowding conditions, i.e.,

$$F = \sum_{l \in L, u \in U^l, q \in Q^l} z_{l,u,q} n_{l,u,q}^{wt} \tag{22}$$

where $z_{l,u,q}$ is the passenger accumulation risk value. Specifically, when the passenger accumulation is small and the station is in a safe state, $z_{l,u,q}$ should be zeros. When the state of the station becomes unsafe or even dangerous with the increase in passenger accumulation, the value of $z_{l,u,q}$ should be intensified to strengthen the penalty for passenger accumulation. A set $\{1, 2, ..., \varphi\}$ comprising different passenger accumulation levels is introduced to represent $z_{l,u,q}$. The value of $z_{l,u,q}$ increases as the level increases, which can be determined by

$$z_{l,u,q} = \begin{cases} 0, \text{ if } 0 \leqslant n_{l,u,q}^{wt} \leqslant \mu_1 \\ \rho_1, \text{ if } \mu_1 < n_{l,u,q}^{wt} \leqslant \mu_2 \\ ..., ... \\ \rho_{\varphi-1}, \text{ if } \mu_{\varphi-1} < n_{l,u,q}^{wt} \leqslant \mu_\varphi \\ \rho_\varphi, \text{ if } n_{l,u,q}^{wt} > \mu_\varphi \end{cases}, \forall \quad l \in L, u \in U^l, q \in Q^l \tag{23}$$

where $\rho_1 \sim \rho_\varphi$ denote the passenger accumulation risk values corresponding to accumulation levels $1 \sim \varphi$, and $\mu_1 \sim \mu_\varphi$ are the critical boundaries of the number of accumulated waiting passengers between two accumulation levels.

Finally, considering that the skipping operations inevitably bring the dissatisfaction of passengers that want to alight at a certain station but the train decides to skip it, it is necessary to introduce the corresponding penalty term to minimize the skipped stations. The penalty term is formulated as

$$P = \sum_{l \in L, u \in U^l, q \in Q^l} x_u^{l,q} \tag{24}$$

Therefore, the objective function is expressed as

$$J = \xi_1(T_{wt} + T_{tf} + T_{st}) + \xi_2 F + \xi_3 P \tag{25}$$

where ξ_1, ξ_2, and ξ_3 are positive weights of total waiting time, station crowding, and the penalty of skipped stations respectively.

2.2.5. The optimization model

According to the above descriptions, the timetable coordination optimization problem with skip-stop strategies of metro networks will be modelled as

$$\begin{aligned} &\min J = \xi_1(T_{wt} + T_{tf} + T_{st}) + \xi_2 F + \xi_3 P \\ &s.t. \begin{cases} (1)-(6), (8)-(24) \\ x_u^{l,q} \in \{0,1\}, \quad \forall l \in L, u \in U^l, q \in Q^l \\ S_{\min}^{l,u,q} \leq S_u^{l,q} \leq S_{\max}^{l,u,q}, \quad \forall l \in L, u \in U^l, q \in Q^l \end{cases} \end{aligned} \tag{26}$$

The above model is in fact a nonlinear and nonconvex programming model, aiming to reduce the passenger waiting time and alleviate station crowding. Constraints (1)-(3) illustrate skip-stop patterns. Constraints (4)-(6) describe the train operations. Constraints (8)-(16) model the passenger behaviours. Constraints (17)-(24) define the total passenger waiting time, station crowding, and the penalty values of skipping operations of trains for the metro network. The complexity of our formulation is analyzed in Table 3 and specific scales of our realistic numerical cases are also given.

Table 3: Numbers of significant variables and constraints.

Variables or constraints	Total number at most	Realistic cases
Binary variable $x_u^{l,q}$	$\sum_{l\in L} m_l \cdot n_l$	2040
Binary variable $\theta_{l,u,q}^{l',q'}$	$\sum_{l\in L, u\in K^l} \sum_{l'\in X_{l,u}} n_l \cdot n_{l'}$	4600
Integer variable $A_u^{l,q}$	$\sum_{l\in L} m_l \cdot n_l$	2040
Integer variable $S_u^{l,q}$	$\sum_{l\in L} m_l \cdot n_l$	2040
Train skip-stop constraints (1)-(3)	$3 \cdot \sum_{l\in L} m_l \cdot n_l$	6120
Train operation constraints (4), (7)	$3 \cdot \sum_{l\in L} m_l \cdot n_l$	10920
Train operation constraints (5)	$\sum_{l\in L} (m_l - 1) \cdot n_l$	1940
Train operation constraints (6)	$\sum_{l\in L} m_l \cdot (n_l - 1)$	1836
Passenger interaction constraints (8)-(12)	$5 \cdot \sum_{l\in L} m_l \cdot n_l$	10200
Transfer coordination constraints (13)-(16)	$4 \cdot \sum_{l\in L, u\in K^l} \sum_{l'\in X_{l,u}} n_l \cdot n_{l'}$	18400

3. Algorithm design

The formulation (26) is a large-scale complex mixed integer nonlinear programming model, involving the coupling constraints between different lines and between skipped and not-skipped stations, and the complex interactive process of passengers and trains. It constitutes a high-dimensional nonlinear and nonconvex optimization problem, and is hard to solve quickly, especially when the network scale is large. In this paper, a novel decomposition and ADP approach is proposed. Under a tailored parallel coordinate descent scheme, our original complex network-level problem is transformed into line-level subproblems with DP formulations, suitable for the efficient ADP method. The overall approach can be considered decentralized, computationally inexpensive and highly desirable in practical applications.

3.1. Model decomposition in a distributed manner

For our formulation, the coupling relationship between connecting lines exists in (13), (14) and (16). For the system, e.g., metro networks, with necessary communication among subsystems, the idea of decomposition and parallel implementation is well worth considering to address the obstacle of high computational complexity to efficient solutions. However, many common processing methods, such as dual decomposition, are difficult to utilize owing to the nonconvexity of subproblems.

Based on the above considerations, a tailored parallel coordinate descent scheme is introduced, the idea of which is widely adopted in dealing with large-scale problems with high computational complexity (Necoara and Clipici, 2013; Wright, 2015; Richtárik and Takáč, 2016; Wu et al., 2018). Specifically, all the variables in problem (26) are partitioned according to operating lines. For each $l \in L$, the passenger waiting time and station crowding can be optimized by solving the l-th line-level subproblem (27) as below, where the variables concerning other lines in coupling constraints, i.e., $n^{al}_{l',u,q'}$ and $A^{l',q'}_{u}$, for $u \in K^l, l' \in X_{l,u}, q' \in Q^{l'}$, are considered as fixed, which are denoted as $\hat{n}^{al}_{l',u,q'}$ and $\hat{A}^{l',q'}_{u}$. Note that in each iteration, all subproblems are solved simultaneously, and the fixed values in each subproblem are derived from the update in the previous iteration to ensure the independence of each subproblem and realize parallelism.

$$\min J^l = \xi_1(T^l_{wt} + T^l_{pf} + T^l_{st}) + \xi_2 F^l + \xi_3 P^l$$

$$
s.t.\begin{cases}
(1)-(6),(8)-(12),(15),(17),(18) \\
T_{wt}^{l}=\sum\limits_{u\in U^{l},q\in Q^{l}} w_{u,v}^{l,q} \\
T_{st}^{l}=\sum\limits_{u\in U^{l},q\in Q^{l}} n_{l,u,q}^{st}\left(D_{u}^{l,q+1}-D_{u}^{l,q}\right) \\
T_{tf}^{l}=\sum\limits_{u\in K^{l},q\in Q^{l}}\sum\limits_{l'\in X_{l,u},q'\in Q^{l'}} g_{l,u,q}^{l',q'}\left(D_{u}^{l,q}-\hat{A}_{u}^{l',q'}-e_{l}^{l'}\right) \\
F^{l}=\sum\limits_{u\in U^{l},q\in Q^{l}} z_{l,u,q}n_{l,u,q}^{wt} \\
P^{l}=\sum\limits_{u\in U^{l},q\in Q^{l}} x_{u}^{l,q} \\
D_{u}^{l,q}-\left(\hat{A}_{u}^{l',q'}+e_{l}^{l'}\right)>M\left(\theta_{l,u,q}^{l',q'}-1\right), \quad \forall u\in K^{l},q\in Q^{l},l'\in X_{l,u},q'\in Q^{l'} \\
D_{u}^{l,q}-\left(\hat{A}_{u}^{l',q'}+e_{l}^{l'}\right)\leq M\theta_{l,u,q}^{l',q'}, \quad \forall u\in K^{l},q\in Q^{l},l'\in X_{l,u},q'\in Q^{l'} \\
g_{l,u,q}^{l',q'}=\beta_{u}^{l,q}y_{l,u,q}^{l',q'}\hat{n}_{l',u,q'}^{al}, \quad \forall u\in K^{l},q\in Q^{l},l'\in X_{l,u},q'\in Q^{l'} \\
x_{u}^{l,q}\in\{0,1\}, \quad \forall u\in U^{l},q\in Q^{l} \\
S_{\min}^{l,u,q}\leq S_{u}^{l,q}\leq S_{\max}^{l,u,q}, \quad \forall u\in U^{l},q\in Q^{l}
\end{cases} \tag{27}
$$

The line-level subproblem (27) obtained through our decomposition technique is a nonlinear dynamic programming (DP) problem, which can not only greatly reduce the scale of the original network-level problem, but also adapt to the application of approximate dynamic programming algorithm to obtain effective solutions. Specifically, under the DP framework, the skip-stop strategies $x_u^{l,q}$ and dwell times $S_u^{l,q}$ are decision variables, the arrival times $A_u^{l,q}$ and the remaining capacity $o_u^{l,q}$ of trains are the state variables, and the corresponding state transition equations can be deduced through equations (4), (5) and (11).

It is noteworthy that, to realize the parallelism of line-level subproblems (27), the variables involving other connecting lines in coupling constraints are fixed, and the fixed values are derived from the previous iteration. In this case, the values of state variables and objectives in each line-level subproblem are local, and slightly biased from the perspective of the whole metro network. Specifically, constraints (13),(14) and equation (16) in the original problem (26) are reformulated and decomposed in subproblem (27) with the fixed values $\hat{n}_{l',u,q'}^{al}$ and $\hat{A}_u^{l',q'}$, for $u \in K^l, l' \in X_{l,u}, q' \in Q^{l'}$, which results in the infeasibility of solutions with contradictions among the values of timetabling variables, connection variables and passenger flow variables. Accordingly, we design an event-based algorithm to present procedures of feasibility recovery. Note that (1)-(7) keep intact in subproblems owing to their independence for each line, so the timetables obtained by solving all subproblems are feasible for train operations in metro networks. Therefore, we utilize this special property to readjust the values of connection and passenger flow variables with the fixed feasible timetables. Hence, the fixed values and objectives globally update with the timetables obtained after the completion of all subproblems in each iteration, as shown in Algorithm 1.

In Algorithm 1, for each iteration of our parallel scheme, the values of dwell times $S_u^{l,q}$ and skip-stop strategies $x_u^{l,q}$ from solving subproblems (27) are set as inputs, the updated values of objective function J, and the fixed values $\hat{n}_u^{l,q}, \hat{A}_u^{l,q}$ are outputs, for $l \in L, u \in U^l, q \in Q^l$. In the process, the study horizon is divided into multiple intervals based on transfer arrival events. During each interval, the corresponding target train operation events are determined, the passenger behaviours are recalculated, and the number

Algorithm 1 Feasibility recovery in the tailored parallel coordinate descent scheme.

Step 1: Initialize the number of alighting passengers $\hat{n}^{al}_{l',u,q'} = 0$ at transfer stations, for $l \in L$, $u \in K^l, q \in Q^l$.
Step 2: Divide the study horizon T into multiple intervals, i.e., $T = [1, 2, ..., c_{\max}]$, according to transfer arrival events $(A^{l,q}_u + e^{l'}_l, \forall l \in L, u \in K^l, q \in Q^l, l' \in X_{l,u})$ determined from solutions of subproblems (27).
Step 3: Determine the sets of target lines L_c, target stations $U_{c,l}$ and target trains $Q_{c,l,u}$ at each interval based on departure times determined from solving subproblems (27). Specifically, if the departure time $D^{l,q}_u$ is between two transfer arrival events, the line-station-train sequence l-u-q is considered to be located in the corresponding interval.
Step 4: Do for interval $c = 1, ..., c_{\max}$.
Step 4a: Do for each target train $q \in Q_{c,l,u}$ at station $u \in U_{c,l}$ of line $l \in L_c$ in interval c.
Step 4b: Calculate transfer connection relationship $\theta^{l',q'}_{l,u,q}$, for $l' \in X_{l,u}, q' \in Q^{l'}$, using constraints (13) and (14).
Step 4c: calculate the number of transfer passengers $g^{l',q'}_{l,u,q}$ using equation (16) with $\theta^{l',q'}_{l,u,q}$ from Step 4b, and the number of alighting passengers $\hat{n}^{al}_{l',u,q'}$ from the previous intervals, for $l' \in X_{l,u}$, $q' \in Q^{l'}$.
Step 4d: Calculate passenger behaviours using (8)-(12), and pass updated values of the number of alighting passengers at transfer stations to the next interval.
Step 5: Update the objective value J using (17)-(25). Set $\hat{A}^{l,q}_u = A^{l,q}_u$, for $l \in L, u \in K^l, q \in Q^l$.
Step 6: Output the updated objective value J and the updated fixed values $\hat{A}^{l,q}_u$ and $\hat{n}^{al}_{l',u,q'}$ for $l \in L, u \in K^l, q \in Q^l$.

of alighting passengers at transfer stations are passed to the next interval to participate in the calculation of the number transfer passengers. Note that the generation of transfer flows occurs at the bound of two intervals, namely the event used for dividing horizons. Hence, it will not affect the target train events and corresponding passenger behaviours belonging to the previous intervals, ensuring the accuracy of transfer flows for the overall metro network in the process.

Note that through our tailored parallel scheme, the network-level complex problem is converted into line-level subproblems (27) with desirable DP property. Although the subproblem still maintains the high computational challenge of nonconvexity and nonlinearity, it is suitable for an ADP approach to realize fast solution.

3.2. ADP approach to subproblems

Approximate dynamic programming (ADP) is a useful method to deal with problems of maximizing or minimizing rewards through learning strategies in the process of interaction between agents and environments, which effectively avoids the curse of dimensionality of traditional DP (Papageorgiou et al., 2015; Liu et al., 2018; He et al., 2022). For our subproblem (27), the corresponding framework and solution method are introduced as follows.

3.2.1. ADP framework.

To apply the ADP method, our subproblem (27) is transformed into a multi-stage decision optimization problem. Decisions (i.e., dwell times and skip-stop strategies) made at a certain stage (i.e., for a station) have impacts on future results (i.e., train arrival times, remaining capacity of trains, and the waiting passengers). The basic elements of ADP, i.e., the action, state, and reward function, are described as follows.

Firstly, for our line-level timetabling subproblem, the action vector $a_{u,q}^{l} \in \mathbb{A}_{u,q}^{l}$ is specified as the dwell times, and whether trains skip stations. $\mathbb{A}_{u,q}^{l}$ is the set of possible actions for train q at station u of line l. At each stage, i.e., each station, the train will take actions according to the current state, i.e., $a_{u,q}^{l} = [S_u^{l,q}, x_u^{l,q}]$, $\forall u \in U^l, q \in Q^l$.

Secondly, in the interaction between trains and passengers, the arrival times of trains will influence the distribution of waiting and loading passengers, and meanwhile, the passenger behaviours will also affect adjustments of the arrival time through optimizing the objective function. Therefore, we define the state vector $z_{u,q}^{l} \in \mathbb{Z}_{u,q}^{l}$ as the arrival time, train remaining capacity and waiting passengers, i.e., $z_{u,q}^{l} = [A_u^{l,q}, o_u^{l,q}, n_{l,u,q}^{wt'}]$, $\forall u \in U^l, q \in Q^l$ where $\mathbb{Z}_{u,q}^{l}$ is the set of possible states for train q at station u of line l. $n_{l,u,q}^{wt'}$ is the number of waiting passengers when train q arrives.

When train q arrives at station u of line l at state $z_{u,q}^{l}$, it makes decisions $a_{u,q}^{l} = \mathscr{A}^{\pi}\left(z_{u,q}^{l}\right)$ with policy π to determine the skip-stop strategies and dwell times, where policy π is a mapping from the state to the action. Actually, with a policy, a unique decision will be determined at a state. Then the train leaves for station $(u+1)$ at state $z_{u+1,q}^{l}$, which is derived from the state transition functions. Specifically, the state transition of the remaining capacity $o_{u+1}^{l,q}$ of trains can be expressed as equation (11). Based on (4) and (5), the state transition of the arrival time $A_{u+1}^{l,q}$ is formulated as

$$A_u^{l,q} = A_{u-1}^{l,q} + R_{u-1}^{l,q} + S_{u-1}^{l,q}\left(1 - x_u^{l,q}\right), \quad \forall u \in U^l, q \in Q^l \tag{28}$$

The number of waiting passengers $n_{l,u,q}^{wt'}$, when train q arrives, is calculated by

$$n_{l,u,q}^{wt'} = \sum_{t \in \left[D_u^{l,q-1}, A_u^{l,q}\right]} p_u^l(t) + \sum_{l' \in X_{l,u}, q' \in Q^{l'}} {g'}_{l,u,q}^{l',q'} + n_{l,u,q-1}^{st}, \quad \forall u \in U^l, q \in Q^l \tag{29}$$

where ${g'}_{l,u,q}^{l',q'}$ represents the number of waiting transfer passengers when train q arrives. Based on the idea of replacing $D_u^{l,q}$ in constraints (13) and (14) with $A_u^{l,q}$, ${g'}_{l,u,q}^{l',q'}$ can be calculated by

$$A_u^{l,q} - \left(A_u^{l',q'} + e_l^{l'}\right) > M\left({\theta'}_{l,u,q}^{l',q'} - 1\right), \quad \forall u \in K^l, q \in Q^l, l' \in X_{l,u}, q' \in Q^{l'} \tag{30}$$

$$A_u^{l,q} - \left(A_u^{l',q'} + e_l^{l'}\right) \le M{\theta'}_{l,u,q}^{l',q'}, \quad \forall u \in K^l, q \in Q^l, l' \in X_{l,u}, q' \in Q^{l'} \tag{31}$$

$${y'}_{l,u,q}^{l',q'} = \max\left\{0, {\theta'}_{l,u,q}^{l',q'} - \theta_{l,u,q-1}^{l',q'}\right\}, \quad \forall u \in K^l, q \in Q^l, l' \in X_{l,u}, q' \in Q^{l'} \tag{32}$$

$${g'}_{l,u,q}^{l',q'} = \beta_u^{l,q}{y'}_{l,u,q}^{l',q'}\hat{n}_{l',u,q'}^{al}, \quad \forall u \in K^l, q \in Q^l, l' \in X_{l,u}, q' \in Q^{l'} \tag{33}$$

Based on the objective function (25), we can decompose it to construct the reward function for each train $q \in Q^l$, at each station $u \in U^l$, as:

$$\begin{aligned}\mathit{\Gamma}\left(z_{u,q}^{l}, a_{u,q}^{l}\right) =& \mathit{\Gamma}\left(A_u^{l,q}, o_u^{l,q}, n_{l,u,q}^{wt'}, S_u^{l,q}, x_u^{l,q}\right) \\ =& \xi_1\left[w_{u,v}^{l,q} + n_{l,u,q}^{st}\left(D_u^{l,q+1} - D_u^{l,q}\right)\right] + \xi_2 z_{l,u,q} n_{l,u,q}^{wt} + \xi_3 x_u^{l,q}, \quad \forall z_{u,q}^{l} \in \mathbb{Z}_{u,q}^{l}, a_{u,q}^{l} \in \mathbb{A}_{u,q}^{l}\end{aligned} \tag{34}$$

where the first term denotes the passenger waiting time for the train q at station u of line l, the second term means the corresponding station crowding, the third term involves the penalty of train q skipping station u. Hence, our goal is converted into finding a policy π minimizing the total rewards for all trains at all stages, i.e., $\min_{\pi} \sum_{q \in Q^l} \sum_{u \in U^l} \mathit{\Gamma}\left(z_{u,q}^{l}, \mathscr{A}^{\pi}\left(z_{u,q}^{l}\right)\right)$.

3.2.2. Solution method.

Based on the ADP framework, an efficient method for line-level subproblems based on lookahead policy and approximation value iteration is introduced to quickly obtain an approximate optimal solution. For the train timetable optimization problem, we need to determine the dwell times and whether to skip stations for each train. In this process, the idea of the lookahead policy is adopted to combine an approximation of future information with future actions (Powell, 2007), and the approximate optimal timetables will be attained by considering the influence of actions on subsequent stages. The goal of our line-level problem can be represented as an optimal policy given by

$$\mathscr{A}^*\left(z_{u,q}^l\right)=\underset{a_{u,q}^l\in\mathbb{A}_{u,q}^l}{\arg\min}\left\{\varGamma\left(z_{u,q}^l,a_{u,q}^l\right)+\min_{\pi}\mathbb{E}\left[\sum_{q\in Q^l}\sum_{u'=u+1}^{m_l}\varGamma\left(z_{u',q}^l,\mathscr{A}^{\pi}\left(z_{u',q}^l\right)\right)|z_{u,q}^l\right]\right\} \tag{35}$$

which shows that the impacts of subsequent stages are considered in the decision process of train timetables. Besides, the Bellman's equation is derived since (35) is usually difficult to solve directly, i.e.,

$$V_{u,q}^l\left(z_{u,q}^l\right)=\min_{a_{u,q}^l\in\mathbb{A}_{u,q}^l}\left\{\varGamma\left(z_{u,q}^l,a_{u,q}^l\right)+\gamma\mathbb{E}\left[V_{u+1,q}^l\left(z'\right)|z_{u,q}^l\right]\right\},\quad\forall z_{u,q}^l\in\mathbb{Z}_{u,q}^l,a_{u,q}^l\in\mathbb{A}_{u,q}^l,q\in Q^l,u\in U^l \tag{36}$$

where $V_{u,q}^l\left(z_{u,q}^l\right)$ means the value of state $z_{u,q}^l$, γ denotes the discount factor and $\gamma\in(0,1]$, while z' denotes all the possible states at state $z_{u,q}^l$ with action $a_{u,q}^l$. Hence, the lookahead policy at state $z_{u,q}^l$ can be realized using

$$a_{u,q}^l=\underset{a_{u,q}^l\in\mathbb{A}_{u,q}^l}{\arg\min}\left\{\varGamma\left(z_{u,q}^l,a_{u,q}^l\right)+\gamma\mathbb{E}\left[V_{u+1,q}^l\left(z'\right)|z_{u,q}^l\right]\right\},\quad\forall z_{u,q}^l\in\mathbb{Z}_{u,q}^l,a_{u,q}^l\in\mathbb{A}_{u,q}^l,q\in Q^l,u\in U^l \tag{37}$$

In our approach, the value function approximation is realized by the lookup table strategy, and the key point of the approach is how to choose actions, i.e., dwell times of train $S_u^{l,q}$ and skip-stop strategies $x_u^{l,q}$, where the adopted function is

$$\hat{v}_{u,q}^{l,\omega}=\min_{a_{u,q}^l\in\mathbb{A}_{u,q}^l}\left[\varGamma\left(A_u^{l,q},o_u^{l,q},n_{l,u,q}^{wt'},S_u^{l,q},x_u^{l,q}\right)+\gamma\bar{V}_{u+1,q}^{l,\omega-1}\left(A_{u+1}^{l,q},o_{u+1}^{l,q},n_{l,u+1,q}^{wt'}\right)\right] \tag{38}$$

where ω represents the number of iterations, the action $a_{u,q}^l=[S_u^{l,q},x_u^{l,q}]$, $\bar{V}_{u,q}^{l,\omega}\left(A_u^{l,q},o_u^{l,q},n_{l,u,q}^{wt'}\right)$ denotes the value approximation of $V_{u,q}^l\left(A_u^{l,q},o_u^{l,q},n_{l,u,q}^{wt'}\right)$ at state $[A_u^{l,q},o_u^{l,q},n_{l,u,q}^{wt'}]$ in the ω-th iteration, which is derived from recurrence equation (39). In the decision-making process, the actions concerning dwell times and skip-stop strategies are obtained from the above minimization problem. Actually, solving the minimization problem can be realized by calculating the value of $\hat{v}_{u,q}^{l,\omega}$ using each discrete action. This process is independent for each action, thus supporting a parallel implementation, which contributes to the acceleration of the algorithm. Meanwhile, $\hat{v}_{u,q}^{l,\omega}$ is used to realize the update of value function approximation, using

$$\bar{V}_{u,q}^{l,\omega}\left(A_u^{l,q},o_u^{l,q},n_{l,u,q}^{wt'}\right)=\left(1-\eta_{\omega-1}\right)\bar{V}_{u,q}^{l,\omega-1}\left(A_u^{l,q},o_u^{l,q},n_{l,u,q}^{wt'}\right)+\eta_{\omega-1}\hat{v}_{u,q}^{l,\omega} \tag{39}$$

where $\eta_\omega\in(0,1]$. Furthermore, we adopt the time-varying $\epsilon-$greedy strategy in our decision-making phase, which can endure more "exploration" during earlier periods and more "exploitation" during later periods by defining ϵ as a function decreasing with iteration, so as to improve the final solution quality. Algorithm

2 illustrates the process of solving subproblems, where for each iteration of our parallel scheme, the value of action $a^l_{u,q}$ concerning timetabling variables are outputs of solving the l-th subproblem, for $u \in U^l, q \in Q^l$.

Algorithm 2 Procedure of ADP approach for solving the l-th subproblem (27).

Step 1: Initialization.

Step 1a: Initialize the value approximation $\bar{V}^{l,0}_{u,q}\left(A^{l,q}_u, o^{l,q}_u, n^{wt'}_{l,u,q}\right)$.

Step 1b: Initialize the states, i.e., the arrival time $TA^{l,q}_1$, the train remaining capacity $o^{l,q}_1$ and waiting passengers $n^{wt'}_{l,1,q}$ at the first station.

Step 1b: Initialize the policy $\mathscr{A}^\pi\left(A^{l,q}_u, o^{l,q}_u, n^{wt'}_{l,u,q}\right)$ used for determining train timetables.

Step 1c: Set iteration number $\omega = 1$.

Step 2: Do for $q = 1, 2, ..., n_l$.

Step 2a: Do for $u = 1, 2, ..., m_l - 1$.

Step 2b: Determine the timetabling action $a^l_{u,q} = [S^{l,q}_u, x^{l,q}_u]$, namely the dwell times and skip-stop strategies at the ω-th iteration using $[S^{l,q}_u, x^{l,q}_u] = \mathscr{A}^\pi\left(A^{l,q}_u, o^{l,q}_u, n^{wt'}_{l,u,q}\right)$ at the $(\omega - 1)$-th iteration.

Step 2c: Update the subsequent state $z^l_{u+1,q} = [A^{l,q}_{u+1}, o^{l,q}_{u+1}, n^{wt'}_{l,u+1,q}]$. Specifically, calculate the arrival time $A^{l,q}_{u+1}$ using (28), the remaining capacity $o^{l,q}_{u+1}$ using (11), and the waiting passengers $n^{wt'}_{l,u+1,q}$ using (29), at station $(u + 1)$, with $A^{l,q}_u$, $o^{l,q}_u, n^{wt'}_{l,u,q}$ and timetabling action $a^l_{u,q}$ at station u from Step 2b.

Step 3: Update value function approximation using

$$\bar{V}^{l,\omega}_{u,q}\left(A^{l,q}_u, o^{l,q}_u, n^{wt'}_{l,u,q}\right) = (1 - \eta_{\omega-1})\,\bar{V}^{l,\omega-1}_{u,q}\left(A^{l,q}_u, o^{l,q}_u, n^{wt'}_{l,u,q}\right) + \eta_{\omega-1}\hat{v}^{l,\omega}_{u,q}$$

Step 4: Update the policy using

$$\mathscr{A}^\pi\left(A^{l,q}_u, o^{l,q}_u, n^{wt'}_{l,u,q}\right) = \begin{cases} \text{Choose an action } a^l_{u,q} = [A^{l,q}_1, S^{l,q}_u, x^{l,q}_u] \text{ randomly from the set } \mathbb{A}^l_{u,q}, \text{ if } P^l_{u,q} \leqslant \epsilon \\ \arg\min\limits_{a^l_{u,q} \in \mathbb{A}^l_{u,q}} \left\{ \begin{array}{c} \varGamma\left(A^{l,q}_u, o^{l,q}_u, n^{wt'}_{l,u,q}, S^{l,q}_u, x^{l,q}_u\right) \\ \gamma\bar{V}^{l,\omega}_{u+1,q}\left(A^{l,q}_{u+1}, o^{l,q}_{u+1}, n^{wt'}_{l,u+1,q}\right) \end{array} \right\}, \text{ otherwise} \end{cases}$$

where $P^l_{u,q} \leqslant \epsilon$ means that the timetabling action $a^l_{u,q} = [A^{l,q}_1, S^{l,q}_u, x^{l,q}_u]$ will be randomly chosen from $\mathbb{A}^l_{u,q}$ with probability ϵ.

Step 5: Increment ω. If $\omega < \omega_{max}$, go to Step 2.

Step 6: Output the final timetabling action $a^l_{u,q} = [S^{l,q}_u, x^{l,q}_u]$, for $u \in U^l, q \in Q^l$.

According to literatures (Mohri et al., 2018; Powell, 2007), for Algorithm 2, the approximate value function sequence is convergent, which is presented as the following proposition:

Proposition 3.1. *For Algorithm 2, given by the update rule* (39)*, the approximate value function sequence* $\left\{\bar{V}^{l,\omega}_{u,q}\right\}$ *converges to the optimal value* $V^l_{u,q}$ *as* $\omega \to \infty$*, i.e., each state-action pair is visited infinitely many times, for* $l \in L, u \in U^l, q \in Q^l$.

According to Proposition 3.1, when all state-action pairs are visited infinitely often, ADP algorithm will converges to the optimal value $V^l_{u,q}$. Based on this, in our impure exploitation algorithms, we use the best action that appears so far to provide high-quality train timetables. Concerning the stopping condition, reaching the maximum iteration ω_{max} marks the completion of the ADP method to the l-th subproblem.

3.3. Overall procedure of decomposition and ADP algorithm

Overall, with the tailored parallel coordinate descent scheme and approximate dynamic programming approach, we design an efficient decomposition and ADP approach to provide the network-level train timetabling strategies. In our approach, the original complex problem is converted into many line-level subproblems. It ensures parallelism and DP characteristics of subproblms suitable for ADP method to quickly solve, so as to lighten the computation burden. The complete solution procedure of decomposition and ADP algorithm is summarized in Algorithm 3, and illustrated as flowchart in Figure 2.

Algorithm 3 Procedure of decomposition and ADP algorithm

Step 1: Initialization.

Step 1a: Initialize the values of arrival times of trains $\hat{A}_u^{l,q}(0)$, and the number of alighting passengers $\hat{n}_{l,u,q}^{al}(0)$, for $l \in L, u \in K^l, q \in Q^l$.

Step 1b: Set iteration number $\tau = 1$.

Step 2: Solve subproblems in parallel using Algorithm 2 to obtain actions concerning train timetables $\mathcal{A}^\tau = \left\{ S_u^{l,q}, x_u^{l,q} \,\middle|\, l \in L, u \in U^l, q \in Q^l \right\}$ at the τ-th iteration, with the fixed values of arrival times $\hat{A}_u^{l',q'}(\tau - 1)$ of trains and the number of alighting passengers $\hat{n}_{l,u,q}^{al}(\tau - 1)$ at the $(\tau - 1)$-th iteration.

Step 3: Update the value of objective J^τ, the fix values of $\hat{A}_u^{l',q'}(\tau)$ and $\hat{n}_{l,u,q}^{al}(\tau)$, for $l \in L, u \in U^l, q \in Q^l$, using Algorithm 1 at the τ-th iteration.

Step 4: Update the timetabling action sequence by $\mathcal{A}^\tau = \begin{cases} \mathcal{A}^\tau, \text{if } J^\tau \leqslant J^{\tau-1} \\ \mathcal{A}^{\tau-1}, \text{otherwise} \end{cases}$, and the objective value by $J^\tau = \min \left\{ J^\tau, J^{\tau-1} \right\}$ at the τ-th iteration.

Step 5: Increment the iteration number τ. If $\tau < \tau_{max}$, go to Step 2, where τ_{max} denotes the maximum iterations.

Step 6: Output the final timetabling action sequence $\mathcal{A}^\tau$ and the final objective value J^τ.

The proposed approach effectively lightens the computation burden to obtain high-quality solutions quickly. Besides, compared with centralized implementation, the parallel structure allows higher reliability and flexibility. The non-increasing property of Algorithm 3 is given as the following theorem:

Theorem 3.1. *For Algorithm 3, let J^τ and $\mathcal{A}^\tau$ be obtained by our distributed coordinated descent scheme. Then, for $\tau = 1, 2, ...$, the iterative objective value sequence $\{J^\tau\}$ is a monotonically non-increasing and convergent sequence.*

Proof. For Algorithm 3, the objective value is updated by $J^\tau = \min \left\{ J^\tau, J^{\tau-1} \right\} \leqslant J^{\tau-1}$ We can conclude that $\{J^\tau\}$ is monotonically non-increasing. Since $\{\mathcal{A}^\tau\}$ is the sequence of feasible solutions of (26), the values of J^τ for $\tau = 1, 2, ...$ are bounded below by the optimal value J^* of (26). Hence, the below-bounded non-increasing sequence $\{J^\tau\}$ for $\tau = 1, 2, ...$ can be proved as a convergent sequence. □

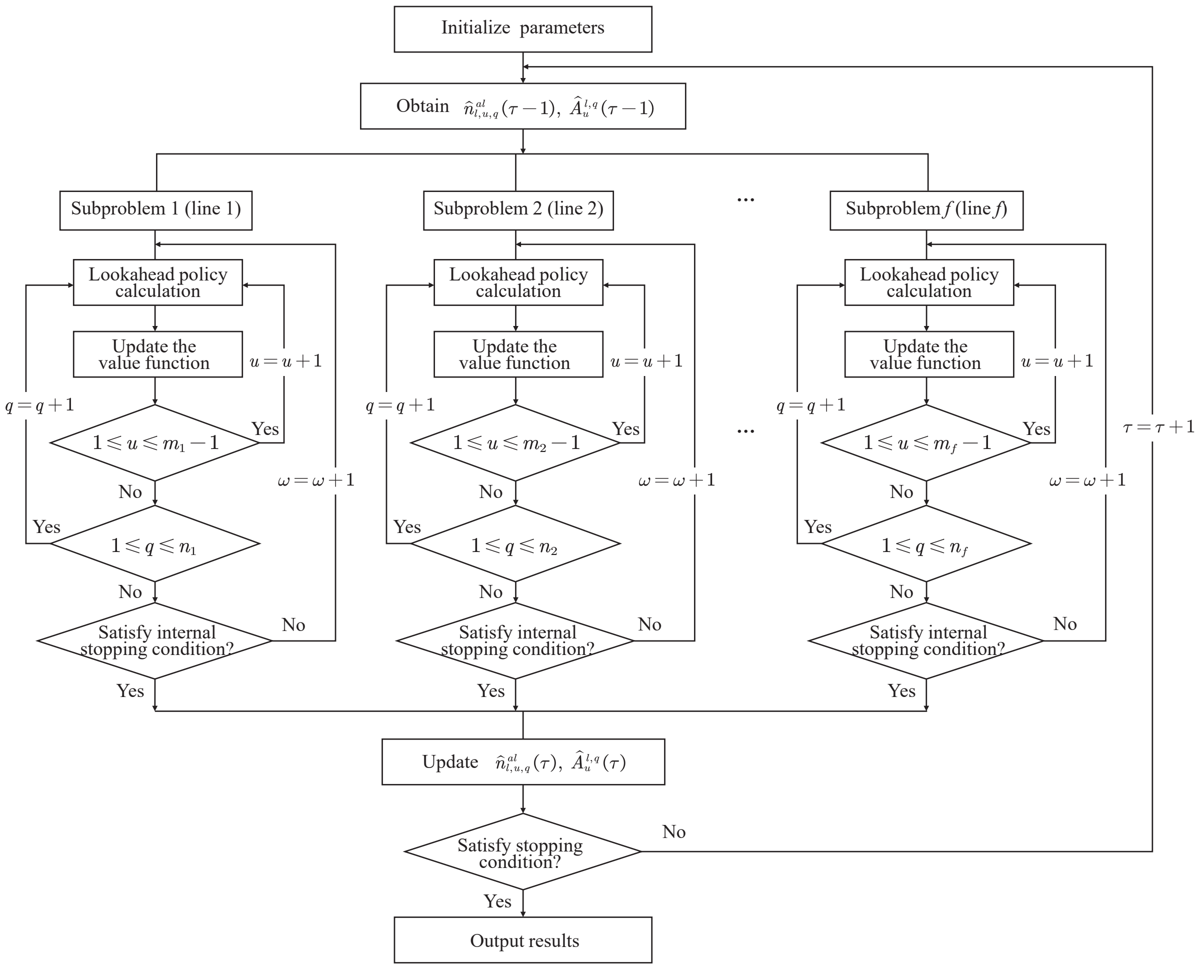


Figure 2: Illustration of decomposition and ADP approach for timetable coordination optimization.

Based on the above, the original large-scale network-level problem is decomposed into multiple line-level subproblems to compute in parallel in Algorithm 3. It effectively lightens the computation burden to obtain high-quality solutions quickly. Compared with centralized implementation, the distributed structure allows higher reliability, fault tolerance and flexibility. When new metro lines are introduced to the study network, only those lines connecting with them are affected in the algorithm. Besides, according to Theorem 1, the non-increasing property of Algorithm 3 is proved, $\{J^\tau\}$ is a convergent sequence for $\tau = 1, 2, ...$ It ensures to a certain extent that a high-quality approximate optimal solution can be found after a certain number of iterations, which can provide reliable and feasible train timetables for metro networks when the stopping condition is satisfied. Concerning the stopping condition, we can consider the number of iterations, the best objective value remaining unchanged for several consecutive steps, and so on. The actual performance of our algorithm is tested in Section 4.

4. Numerical Examples

The performance of our optimization model and solution algorithm are tested in two metro networks: a simplified network with three metro lines, and a real-world metro network based on Beijing Subway. Both are implemented by by MATLAB R2021a with Gurobi 9.1.2 on the Windows 10 PC (Core i5-8400 CPU, 16 GB RAM).

4.1. Experiment 1: a simple network

In this experiment, a small-scale network consisting of three operating lines is considered, as shown in Figure 3. There are seven stations on each line. Station 3 of line 1 and station 4 of line 2 are the same physical station providing transfer services, while Station 5 of line 1 and station 4 of line 3 are the same physical transfer station. For this case network, a total of 15 trains are considered. The lower and upper limits to dwell time are 30 s and 60 s. The minimum headways are 200, 240, 200, for line 1, 2, 3, respectively. The running times between stations are considered given and fixed, which are set as [100 s, 120 s, 110 s, 120 s, 115 s, 110 s], [115 s, 105 s, 120 s, 125 s, 120 s, 115 s], [105 s, 115 s, 120 s, 100 s, 115 s, 120 s]. The average transfer walking time is 120 s. The passenger demand profiles for different stations of different lines are shown in Figure 4. The train capacity is taken as 100 pax. Regarding the parameters of station crowding, the accumulation risk values are set as 30 and 50, and the corresponding critical boundaries are 80 pax and 150 pax. The weights in the objective, ξ_1, ξ_2 and ξ_3 are set as 1, 10 and 1000, respectively. Concerning the specific values of parameters related to our algorithm, the discount factor $\gamma = 0.98$, $\eta = 0.8$; the proportion $\epsilon = \omega^{-0.5}$, changing with iteration ω; the maximum iteration for solving subproblems is 500, and the maximum iteration in distributed scheme is set as 100.

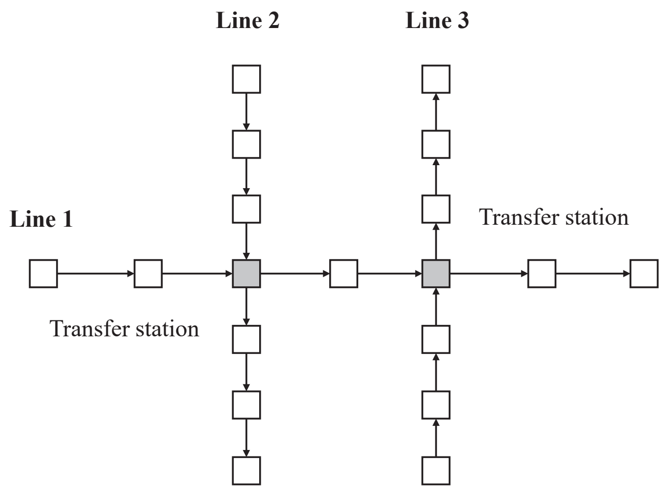


Figure 3: Illustration of the simple metro network.

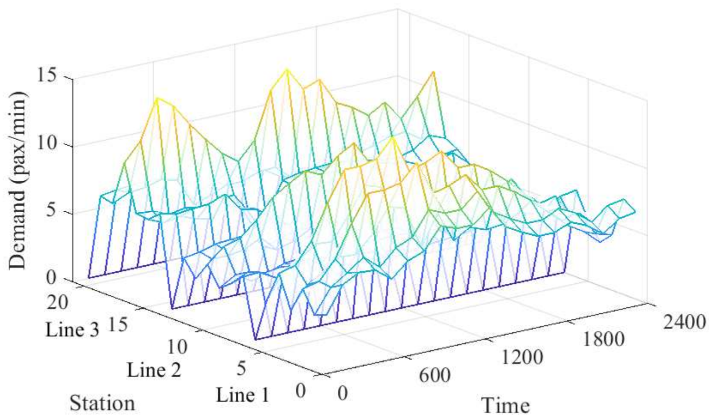
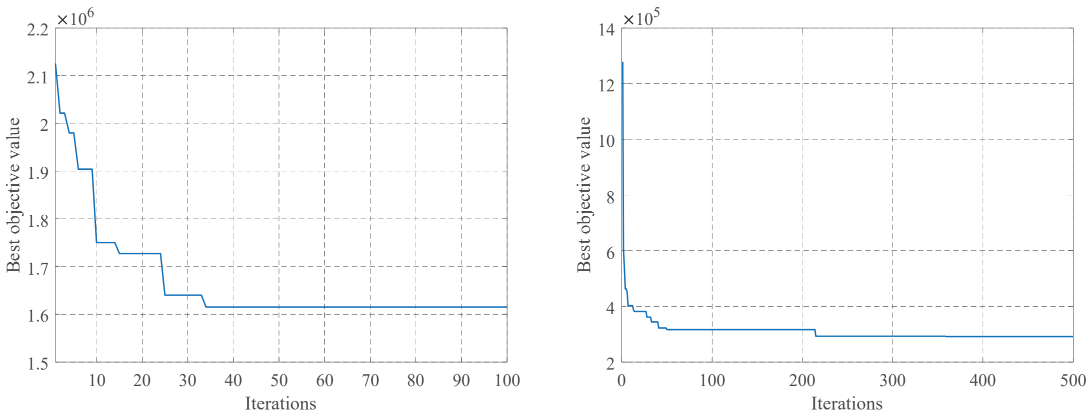


Figure 4: Passenger demand profiles for the simple network.

Given the above parameters, the model applied to the simple network can be solved. The iterative processes of our overall approach (Algorithm 3) and solving a subproblem (Algorithm 2) are given in Figure 5. It is clear that our approach converges after 34 iterations in 5 (a), and the best objective value of the subproblem keeps relatively stable after 358 iterations in the internal process in 5 (b).



(a) Iterative process of the whole decomposition and ADP approach

(b) Iterative process of solving a subproblem by ADP approach

Figure 5: The iterative processes of Algorithm 3 for the case.

4.1.1. Performance analysis.

In this section, we conduct experiments on the simple network with the following three timetable scenarios: Case 1: The timetable with fixed dwell times (the minimum dwell times). Case 2: The optimized timetable with skip-stop strategies (no coordination). In this case, skip stopping is allowed while the train timetable of each line is determined independently by ADP algorithm. Case 3: The optimized timetable with coordination between lines (no skip-stop strategy). In this case, skip stopping is not allowed while the network-level train timetables are determined by decomposition and ADP algorithm. Case 4: The timetable optimized for coordination and skip-stops, namely the proposed approach in this paper. We analyze the performances of the three timetables on the passenger waiting time, station crowding, number of skipped stops and the combined objectives. The computational results are presented in Table 4. We can conclude that:

Table 4: Comparison of results under different strategies for the simple network.

Performance index	Case 1	Case 2	Case 3	Case 4
Pssenger waiting time / 10^6	1.249	1.095	1.068	1.063
Station crowding / 10^4	13.494	7.964	7.190	5.464
Number of skipped stops	0	5	0	6
Objective function value / 10^6	2.598	1.896	1.787	1.615

The optimized timetables significantly reduce passenger waiting time and station crowding. Compared with the non-optimized timetable (Case 1), the optimized timetables with different strategies (Case 2, 3, 4) make marked improvements in varying degrees. Specifically, by adjusting the dwell times and skip-stop strategies, the passenger waiting time under three optimized timetables (Case 2, 3, 4) is reduced by 12.32%, 14.46%, and 14.91%, the station crowding is alleviated by 40.98, 46.72%, and 59.51%, while the total objective value is improved by 27.01%, 31.21%, and 37.84%, respectively.

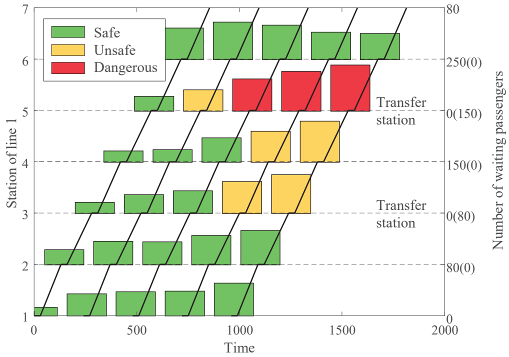
For the optimized timetable with skip-stop strategies but no coordination (Case 2), the performance of it is the worst among the three optimized timetables. The total waiting time is up to $1.095 \cdot 10^6$, the station crowding is $7.964 \cdot 10^4$, and the objective value is $1.896 \cdot 10^6$. This is mainly because such an independent optimization without coordination fails to consider transfer flows during the decision process, resulting in an inaccurate depiction of the interaction among passenger flows, trains and stations. Moreover, when the dwell times and skip-stop strategies decided based on such inaccurate interactions, are loaded into a network with a certain amount of transfer coupling, the performances lower than expected are obtained.

Under the optimized timetable with coordination between trains of different lines, but no skip-stop strategy (Case 3), the passenger waiting and station crowding are reduced by 2.44% and 9.72% compared with Case 2. The advantage of Case 3 over Case 2 is mainly since it can exactly capture the transfer demand, which can ensure that the obtained decisions for trains are effective in improving the total objective value of the whole metro network.

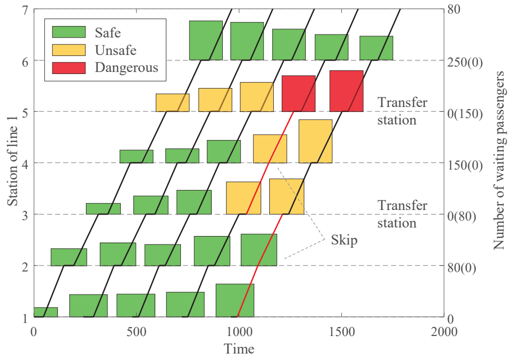
Our proposed approach (Case 4) coconsider coordination between lines and the skip-stop strategy, realize a further improvement over Case 3. The passenger waiting, station crowding, and the combined objective value are improved by 0.53%, 24.00% and 9.64% over Case 3, respectively. We note that with the skip-stop strategy, the improvement mainly reflects in station crowding. The reason is that the skip-stop strategy allows trains to selectively skip some stations to balance the capacity resources of trains plus stations, and the dynamic and uneven passenger demand, thus relieving the pressure of crowded stations. On the premise of accurately describing the transfer demand, the effective skip-stop strategy and dwell times of trains are formulated to improve the passenger service quality and operational safety for metro networks, thus achieving the best optimization effect among the three optimized timetables.

To further illustrate the effect of skip-stop strategies and coordination optimization. We now examine the optimized timetables in detail. Figure 6 presents the train timetables of line 1 with waiting passenger distribution under 4 cases. In this figure, bars represent the number of waiting passengers on platforms, and different colors (i.e., green, yellow, and red) indicate the different passenger accumulation levels of stations (i.e., safe, not safe, and dangerous) divided according to the values of critical boundaries for passenger accumulation. Specifically, the station is in a safe state, when the number of waiting passengers is less than 80 pax; an unsafe state, when it belonging to [80, 150] pax; a dangerous state, when it is greater than 150 pax. Lines describe the train trajectories, and red lines mean the train skipping operations.

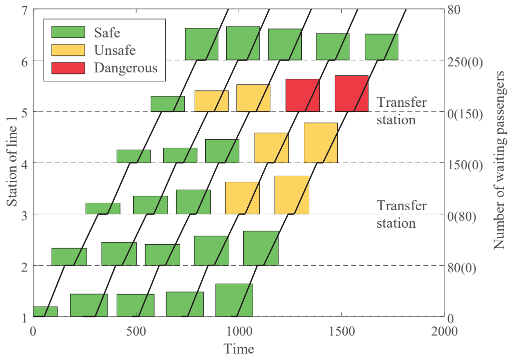
From Figure 6, the passenger accumulation at transfer stations with high demand under Case 1 is the most serious. Under Case 1, the dangerous state has occurred three times, and in the worst case (departure of train 5 from station 5), there are more than 200 passengers waiting. This shows that it is difficult to alleviate station crowding simply by minimizing dwell times to increase departure frequency, thus increasing the risk of safety accidents on platforms. Under Case 2, two skipping operations are performed, i.e. train 4 skips station 4 and train 5 skips station 2, which helps to relieve the crowding of the downstream busy stations, i.e. station 5. However, since Case 2 does not consider the transfer coordination between lines, the dwell times and skip-stop patterns of trains are determined based on inaccurate passenger flows. As a result, despite the asistance of skip operation, the station crowding is still serious, especially the transfer station 5 is always in an unsafe and dangerous state. For Case 3, the transfer coordination is taken into account, and thus the crowding of busy transfer station 5 is alleviated, compared with Case 2 without coordination. Both the number of unsafe states and the number of waiting passengers in dangerous states are reduced. However, for lack of flexible skip-stop patterns, the improvements are limited. Regarding our proposed approach, i.e., Case 4, thanks to the combination of the skip-stop strategy and transfer coordination, the station crowding is minimal among all cases. The number of safe states is the largest, and there is no dangerous state. A total of three skipping operations are executed, i.e. train 1 skips station 4, train 3 skips station 2, and train 5 skips station 2. We observe that there is a common regular pattern to such skip-stop operations, namely if the previous train skipped the station, the number of waiting passengers increases when the next train arrived since a certain number of passengers are stranded on platforms by the previous train. Note that in our strategies, trains are more likely to skip stations with low passenger flow at that time, which results in the waiting numbers on platforms still do not exceed the critical capacity in spite of obvious increases when the next train arrives (i.e., train 2 at station 4 and train 4 at station 2). Meanwhile, the effect of skipping operations for downstream stations is significant, especially for the transfer station with high demand. Trains skipped upstream station 2 with relatively lower passenger demand to be equipped with more space to accommodate the waiting passengers at downstream stations 3, 4, and 5, which brings significant alleviation of the pressure of the crowding transfer station 5 compared with other cases. The comparison of results in Table 4 also verifies the above observations, which is of significance to prevent or reduce crowding at stations and then improve safety and reliability for the metro network.



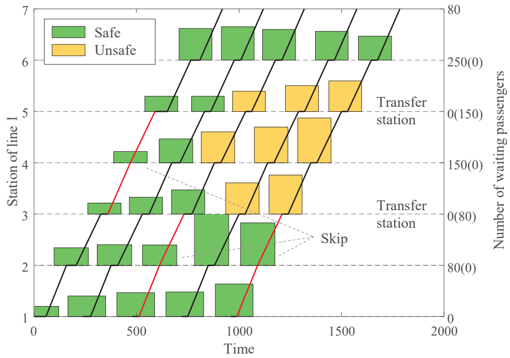
(a) Timetable of line 1 under Case 1



(b) Timetable of line 1 under Case 2



(c) Timetable of line 1 under Case 3



(d) Timetable of line 1 under Case 4

Figure 6: Illustration of the train timetables with the waiting passenger distribution.

4.1.2. Performance comparison with other algorithms.

For the purpose of verifying effectiveness of our decomposition and ADP algorithm (DADP), we compare the optimized results with simulated annealing algorithm (SA) and genetic algorithm (GA), widely adopted in solving train timetable optimization problems (Jamili and Aghaee, 2015b; Guo et al., 2017; Robenek et al., 2018).

For the parameters of SA, the initial temperature is 100. A geometric cooling schedule with the cooling rate 0.98 is adopted. The algorithm stops if the current temperature is smaller than 0.05. The number of iterations at each temperature is 20. Concerning the neighborhood structure, a neighbor solution is obtained by randomly selecting a few components of the current solution to be changed provided that the feasibility is guaranteed. The number of components selected decreases as the temperature decreases. The setting of parameters for GA is as follows. The population size is 40. The number of maximum generations is predetermined as 1000. The algorithm stops if the maximum number of generations are obtained. The crossover rate is 0.6, and the mutation rate is 0.4. Besides, in the process of selection, the roulette method and elite retention strategy are used to maintain stability of population. In the process of crossover and mutation, we adopt the multi-point crossover strategy. The number of points selected decreases as the iteration increases. The search processes concerning the two algorithms and our DADP algorithm are shown in Figure 7. Obviously, with increase of iterations, all of them show improvements in decreasing the best objective values, and the values maintain relatively stable during 1000 iterations.

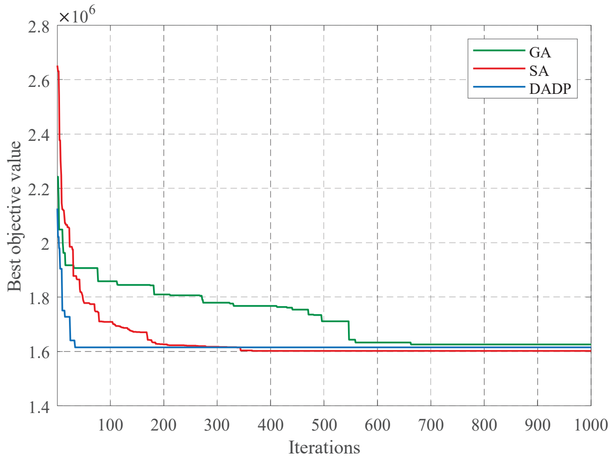


Figure 7: Trend of the objective value in GA, SA and ADP for the simple network.

Table 5 gives the comparison among different algorithms for our problem. We first check the quality of solutions. The final objective function values obtained by the three algorithms are close. The objective function value of SA is the minimum, i.e., $1.603 \cdot 10^6$. The value of DADP is $1.615 \cdot 10^6$, slightly larger than SA. GA obtains the worst solution and the objective function value is $1.625 \cdot 10^6$. Regarding the computation time, among the three algorithms, GA, as a classical swarm intelligence-based method, the computation time is the largest (181.107 s), since each iteration involves the operations of multiple solutions (i.e. a population). SA, as a neighborhood-based method, presents higher computational efficiency in this small-scale case. The computation time is 31.387 s. Our decomposition and ADP approach takes the minimum computation time, i.e., 27.802 s. For the small-scale case, SA obtains the solution with the minimum objective function value, our DADP algorithm generates a solution with the similar-quality solution with a shorter computation time. The performance of GA is the worst, it takes a the longest time (149.729 s and 153.304 s longer than SA and DADP) to get the worst solution (1.36% and 0.64% larger than SA and DADP). Totally, DADP and SA behave similarly in this small-scale case, with slight advantages in computation time and solution quality respectively. However the advantage of our DADP approach will be prominent in solving the problem involving large-scale network.

Table 5: Comparison of optimization results of different algorithms for the simple network.

Algorithm	Passenger waiting time / 10^5	Station crowding / 10^4	Number of skipped stops	Objective function value / 10^6	Computation time (s)
SA	9.535	6.440	6	1.603	31.378
GA	10.418	5.745	9	1.625	181.107
DADP	10.625	5.464	6	1.615	27.802

4.2. Experiment 2: a real-world network

In this part, we implement a set of examples in a large real-world metro network, so as to further test the effectiveness of our model and approach. Part of Beijing Subway is shown in Figure 8. It consists of 10 operating lines, 205 service stations and 46 transfer service stations, and the considered operation directions of lines are marked, where there are 32 stations on the longest operating line.

Figure 8: Illustration of part of Beijing Subway network.

Concerning specific values of relevant parameters, a total of 100 trains is considered. The minimum and maximum dwell time are 30 s and 60 s, the minimum headway and the number of stations of each line are given in Table 6. The running time between stations is respectively set according to the actual length of operating lines and the speed of trains. The train capacity is 1650 pax. Regarding the parameters of station crowding, the accumulation risk values are set as 30 and 50, and the corresponding critical boundaries are 1000 pax and 1500 pax. The demand information is obtained from historical operating data of the AFC systems. The average transfer walking time is 120 s. The weights of objectives are set as 1, 10 and 1000, respectively. For parameters related to the algorithm, the discount factor $\gamma = 0.98$, $\eta = 0.8$, and the proportion $\epsilon = \omega^{-0.5}$, changing with iteration ω. The maximum iterations for solving subproblems and the overall distributed scheme are 500 and 100.

Given the above parameters, the model applied to the real-world network can be solved. According to the result, train timetables of 10 lines with skip-stop patterns can be obtained, in which all operating trains conduct a total of 74 skipping operations. The passenger waiting time is $2.314 \cdot 10^8$, the penalty value of station crowding is $6.754 \cdot 10^7$, and the objective function is $9.068 \cdot 10^8$. To assess the performance of our model

Table 6: Parameters about operating lines of the real-world case.

Parameter	Line 1	Line 2	Line 4	Line 5	Line 6
Number of stations	23	18	24	23	32
Minimum headways (s)	105	120	120	120	120
Parameter	Line 7	Line 8 North	Line 8 South	Line 9	Line 15
Number of stations	21	19	12	13	20
Minimum headways (s)	180	120	300	120	190

and approach, the comparison of results between the common practical train timetables with fixed headways and those of our proposed method is given in Table 7. Compared with the practical train timetables, the objective function value of each line is improved with decreases between 6.828% and 48.473%. For the whole metro network, the total objective function value is decreased by 19.564%, where the total waiting time and station crowding are reduced by 21.418% and 18.917%, respectively. The above results present that, formulating flexible dwell times and skip-stop strategies for trains with our method considering transfer coordination, the capacity resources of trains plus stations, and the dynamic and uneven passenger flows can be balanced, thus reducing the passenger waiting time and relieving the pressure of crowded stations, which contributes to improving the passenger service quality and operational safety for the metro network.

Table 7: Comparison of results under different strategies for the real-world network.

Strategy	Performance indicator	Line 1	Line 2	Line 4	Line 5	Line 6
Common practical timetables	Passenger waiting time $/10^6$	15.662	12.660	16.234	57.209	70.808
	Station crowding $/10^6$	3.713	2.106	1.694	21.088	24.830
	Number of skipped stops	0	0	0	0	0
	Objective value $/10^6$	52.791	33.725	33.169	268.090	319.105
The proposed approach	Passenger waiting time $/10^6$	11.840	9.599	13.516	45.105	52.747
	Station crowding $/10^6$	2.198	1.397	1.073	18.214	20.393
	Number of skipped stops	3	4	7	15	12
	Objective value $/10^6$	33.826	23.570	24.251	227.257	256.693
Strategy	Performance indicator	Line 7	Line 8 North	Line 8 South	Line 9	Line 15
Common practical timetables	Passenger waiting time $/10^6$	46.999	36.339	4.043	10.528	23.931
	Station crowding $/10^6$	10.002	12.434	0.000	1.909	5.523
	Number of skipped stops	0	0	0	0	0
	Objective value $/10^6$	147.022	160.675	4.043	29.617	79.164
The proposed approach	Passenger waiting time $/10^6$	38.261	29.539	3.762	7.697	19.290
	Station crowding $/10^6$	8.615	10.406	0.000	0.756	4.490
	Number of skipped stops	6	8	5	7	7
	Objective value $/10^6$	124.413	133.603	3.767	15.261	64.197

Moreover, to further verify the effectiveness of our algorithm for large-scale network-level problems, we compare the optimized result with that of GA and SA algorithm. For the parameters of SA, the number of iterations at each temperature is 30. Other parameters and rules, i.e, the initial and stopping temperatures, the cooling schedule, the stopping condition, and the rule of generating neighborhood solutions, are similar to those in the small case. The number of components to change each time obtaining neighborhood solutions gets increased. For the parameters of GA, the population size is 30. The roulette method, elite retention strategy, and multi-point crossover strategies are adopted. Other settings are similar to those in the small case. The number of components in the processes if crossover and mutation is increased. The comparison of results is given in Table 8. Clearly, for train timetable optimization problems in large-scale networks, the advantage of decomposition and ADP algorithm is obvious.

Table 8: Comparison of results of different algorithms for the real-world network.

Algorithm	Passenger waiting time / 10^8	Station crowding / 10^7	Number of skipped stops	Objective function value / 10^9	Computation time (s)
SA	2.766	7.689	92	1.046	2588.733
GA	2.859	8.000	87	1.086	5156.506
DADP	2.314	6.754	74	0.907	530.712

As shown in Table 8, GA requires maintaining a certain size of the population in each iteration, which takes up a lot of computation time, especially when the scale of the problem expands to a relatively large level. Moreover, for the train timetable coordination optimization issue involving the large-scale metro network, GA fails to search for a good solution (the largest objective value $1.086 \cdot 10^9$) in spite of a long computation time (i.e., 5156.506 s). SA can generate a better result in a relatively short time (i.e., 2588.733 s), but actually, its advantage of computation efficiency and solution quality in the small-scale case is not shown in the large-scale case. Compared with the two centralized heuristic algorithms, the decomposition and ADP algorithm performs best for the large-scale case, both the solution quality (with the minimum objective value $0.907 \cdot 10^9$) and computation speed (with the minimum computation time 530.712 s) are the best among the three algorithms. It takes less than 89.708% and 79.499% computation time for our approach to get an objective value 16.494% and 13.269% better than GA and SA, respectively. In general, for the train timetable coordination optimization problems in large-scale networks, it takes much a shorter computation time to generate a solution with higher quality using decomposition and ADP approach.

5. Conclusions

As metro networks rapidly expand, the impact of transfer coordination and unbalanced temporal and spatial demand distribution on the service level of metro systems is becoming increasingly significant. A distributed optimization framework is presented in this paper for timetable coordination of metro networks, providing insights of significance to the efficient generation of train timetables and skip-stop strategies and bringing fundamental significance to the train operational management theory for metro networks. Specifically, a nonlinear programming model is formulated to jointly optimize transfer coordination and skip-stop strategies for metro networks. It aims to generate high-quality timetables for network-level trains, contributing to service quality for passengers and operational safety at stations. Besides, to apply to the large-scale nature of our problem, the computation efficiency is an important focus point. Actually, the large-scale metro network involves complex coupling interaction among non-transfer and transfer passengers and train capacity, which causes that the proposed model inevitably consists of many binary variables and noncovex and nonlinear constraints with high computation complexity. Hence, a decomposition and ADP approach is designed, converting the original large-scale problem into several line-level subproblems, which can ensure parallelism and DP characteristics of subproblms suitable for ADP method to quickly solve, so as to lighten the computation burden.

To demonstrate the validity and practicability of our model and approach, we implement a series of numerical experiments, consisting of a simple one and a complex realistic one on Beijing Subway. The results show that our method efficiently contributes to the reduction of total waiting time and the alleviation of station crowding, so that both service quality and operational safety can be enhanced for the overall metro network. Besides, compared with traditional heuristic algorithms, our approach can obtain better solutions with much less computation time for large-scale problems. Additionally, note that passenger behaviours are affected by many uncertain factors. For these uncertain factors such as transfer walking time, train running time, a robust optimal strategy will be meaningful in the future research.

References

Cadarso, L., Marín, Á., 2012. Integration of timetable planning and rolling stock in rapid transit networks. Annals of operations research 199, 113–135.

Chen, Z., Li, S., D'Ariano, A., Yang, L., 2022. Real-time optimization for train regulation and stop-skipping adjustment strategy of urban rail transit lines. Omega 110, 102631.

Corman, F., DAriano, A., Pacciarelli, D., Pranzo, M., 2012. Optimal inter-area coordination of train rescheduling decisions. Transportation Research Part E: Logistics and Transportation Review 48, 71–88.

Dessouky, M., Hall, R., Zhang, L., Singh, A., 2003. Real-time control of buses for schedule coordination at a terminal. Transportation Research Part A: Policy and Practice 37, 145–164.

Fonseca, J.P., van der Hurk, E., Roberti, R., Larsen, A., 2018. A matheuristic for transfer synchronization through integrated timetabling and vehicle scheduling. Transportation Research Part B: Methodological 109, 128–149.

Frey, M., Kolisch, R., Artigues, C., 2017. Column generation for outbound baggage handling at airports. Transportation Science 51, 1226–1241.

Freyss, M., Giesen, R., Muñoz, J.C., 2013. Continuous approximation for skip-stop operation in rail transit. Procedia-Social and Behavioral Sciences 80, 186–210.

Guo, X., Sun, H., Wu, J., Jin, J., Zhou, J., Gao, Z., 2017. Multiperiod-based timetable optimization for metro transit networks. Transportation Research Part B: Methodological 96, 46–67.

Han, Z., Han, B., Li, D., Ning, S., Yang, R., Yin, Y., 2021. Train timetabling in rail transit network under uncertain and dynamic demand using advanced and adaptive nsga-ii. Transportation Research Part B: Methodological 154, 65–99.

He, S.X., He, J.J., Liang, S.D., Dong, J.Q., Yuan, P.C., 2022. A dynamic holding approach to stabilizing a bus line based on the q-learning algorithm with multistage look-ahead. Transportation Science 56, 31–51.

Hu, Y., Li, S., Dessouky, M.M., Yang, L., Gao, Z., 2022. Computationally efficient train timetable generation of metro networks with uncertain transfer walking time to reduce passenger waiting time: A generalized benders decomposition-based method. Transportation Research Part B: Methodological 163, 210–231.

Jamili, A., Aghaee, M.P., 2015a. Robust stop-skipping patterns in urban railway operations under traffic alteration situation. Transportation Research Part C: Emerging Technologies 61, 63–74.

Jamili, A., Aghaee, M.P., 2015b. Robust stop-skipping patterns in urban railway operations under traffic alteration situation. Transportation Research Part C: Emerging Technologies 61, 63–74.

Jiang, F., Cacchiani, V., Toth, P., 2017. Train timetabling by skip-stop planning in highly congested lines. Transportation Research Part B: Methodological 104, 149–174.

Jiang, Z., Gu, J., Fan, W., Liu, W., Zhu, B., 2019. Q-learning approach to coordinated optimization of passenger inflow control with train skip-stopping on a urban rail transit line. Computers & Industrial Engineering 127, 1131–1142.

Kang, L., Wu, J., Sun, H., Zhu, X., Wang, B., 2015. A practical model for last train rescheduling with train delay in urban railway transit networks. Omega 50, 29–42.

Lamorgese, L., Mannino, C., 2015. An exact decomposition approach for the real-time train dispatching problem. Operations Research 63, 48–64.

Li, S., Zhou, X., Yang, L., Gao, Z., 2018a. Automatic train regulation of complex metro networks with transfer coordination constraints: A distributed optimal control framework. Transportation Research Part B: Methodological 117, 228–253.

Li, S., Zhou, X., Yang, L., Gao, Z., 2018b. Automatic train regulation of complex metro networks with transfer coordination constraints: A distributed optimal control framework. Transportation Research Part B: Methodological 117, 228–253.

Li, W., Xu, R., Luo, Q., Jones, S., 2016. Coordination of last train transfers using automated fare collection (afc) system data. Journal of Advanced Transportation 50, 2209–2225.

Li, Z., Hensher, D.A., 2011. Crowding and public transport: A review of willingness to pay evidence and its relevance in project appraisal. Transport Policy 18, 880–887.

Liu, R., Li, S., Yang, L., 2020. Collaborative optimization for metro train scheduling and train connections combined with passenger flow control strategy. Omega 90, 101990.

Liu, R., Li, S., Yang, L., Yin, J., 2018. Energy-efficient subway train scheduling design with time-dependent demand based on an approximate dynamic programming approach. IEEE Transactions on Systems, Man, and Cybernetics: Systems 50, 2475–2490.

Mannino, C., Mascis, A., 2009. Optimal real-time traffic control in metro stations. Operations Research 57, 1026–1039.

Mohri, M., Rostamizadeh, A., Talwalkar, A., 2018. Foundations of machine learning. MIT press.

Necoara, I., Clipici, D., 2013. Efficient parallel coordinate descent algorithm for convex optimization problems with separable constraints: application to distributed mpc. Journal of Process Control 23, 243–253.

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. Transportation Research Part B: Methodological 76, 117–135.

Papageorgiou, D.J., Cheon, M.S., Nemhauser, G., Sokol, J., 2015. Approximate dynamic programming for a class of long-horizon maritime inventory routing problems. Transportation Science 49, 870–885.

Powell, W.B., 2007. Approximate Dynamic Programming: Solving the curses of dimensionality. volume 703. John Wiley & Sons.

Richtárik, P., Takáč, M., 2016. Parallel coordinate descent methods for big data optimization. Mathematical Programming 156, 433–484.

Robenek, T., Azadeh, S.S., Maknoon, Y., de Lapparent, M., Bierlaire, M., 2018. Train timetable design under elastic passenger demand. Transportation research Part b: methodological 111, 19–38.

Shang, P., Li, R., Liu, Z., Yang, L., Wang, Y., 2018. Equity-oriented skip-stopping schedule optimization in an oversaturated urban rail transit network. Transportation Research Part C: Emerging Technologies 89, 321–343.

Taherkhani, G., Alumur, S.A., Hosseini, M., 2020. Benders decomposition for the profit maximizing capacitated hub location problem with multiple demand classes. Transportation Science 54, 1446–1470.

Wang, Y., De Schutter, B., van den Boom, T.J., Ning, B., Tang, T., 2014. Efficient bilevel approach for urban rail transit operation with stop-skipping. IEEE Transactions on Intelligent Transportation Systems 15, 2658–2670.

Wang, Y., Li, D., Cao, Z., 2020. Integrated timetable synchronization optimization with capacity constraint under time-dependent demand for a rail transit network. Computers & Industrial Engineering 142, 106374.

Wong, R.C., Yuen, T.W., Fung, K.W., Leung, J.M., 2008. Optimizing timetable synchronization for rail mass transit. Transportation Science 42, 57–69.

Wright, S.J., 2015. Coordinate descent algorithms. Mathematical Programming 151, 3–34.

Wu, J., Liu, M., Sun, H., Li, T., Gao, Z., Wang, D.Z., 2015. Equity-based timetable synchronization optimization in urban subway network. Transportation Research Part C: Emerging Technologies 51, 1–18.

Wu, Q., Zeng, Y., Zhang, R., 2018. Joint trajectory and communication design for multi-uav enabled wireless networks. IEEE Transactions on Wireless Communications 17, 2109–2121.

Xu, L., Ng, T.S.A., 2020. A robust mixed-integer linear programming model for mitigating rail transit disruptions under uncertainty. Transportation Science 54, 1388–1407.

Yin, J., DAriano, A., Wang, Y., Yang, L., Tang, T., 2021. Timetable coordination in a rail transit network with time-dependent passenger demand. European Journal of Operational Research 295, 183–202.

Yin, J., Wang, Y., Xun, J., Su, S., Tang, T., et al., 2019. Mixed-integer linear programming models for coordinated train timetabling with dynamic demand, in: 2019 IEEE Intelligent Transportation Systems Conference (ITSC), IEEE. pp. 863–868.

Zhu, Y., Goverde, R.M., 2020. Integrated timetable rescheduling and passenger reassignment during railway disruptions. Transportation Research Part B: Methodological 140, 282–314.