Delft University of Technology

Delft Center for Systems and Control

Technical report 24-003

Real-time train scheduling with uncertain passenger flows: A scenario-based distributed model predictive control approach*

X. Liu, A. Dabiri, Y. Wang, and B. De Schutter

If you want to cite this report, please use the following reference instead:

X. Liu, A. Dabiri, Y. Wang, and B. De Schutter, "Real-time train scheduling with uncertain passenger flows: A scenario-based distributed model predictive control approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 5, pp. 4219–4232, May 2024. doi:10.1109/TITS.2023.3329445

Delft Center for Systems and Control
Delft University of Technology
Mekelweg 2, 2628 CD Delft
The Netherlands
phone: +31-15-278.24.73 (secretary)
URL: https://www.dcsc.tudelft.nl

* This report can also be downloaded via https://pub.bartdeschutter.org/abs/24_003.html

Real-Time Train Scheduling with Uncertain Passenger Flows: A Scenario-Based Distributed Model Predictive Control Approach

Xiaoyu Liu, Azita Dabiri, Yihui Wang, and Bart De Schutter, *Fellow, IEEE*

***Abstract*—Real-time train scheduling is essential for passenger satisfaction in urban rail transit networks. This paper focuses on real-time train scheduling for urban rail transit networks considering uncertain time-dependent passenger origin-destination demands. First, a macroscopic passenger flow model we proposed before is extended to include rolling stock availability. Then, a distributed-knowledgeable-reduced-horizon (DKRH) algorithm is developed to deal with the computational burden and the communication restrictions of the train scheduling problem in urban rail transit networks. For the DKRH algorithm, a cost-to-go function is designed to reduce the prediction horizon of the original model predictive control approach while taking into account the control performance. By applying a scenario reduction approach, a scenario-based distributed-knowledgeable-reduced-horizon (S-DKRH) algorithm is proposed to handle the uncertain passenger flows with an acceptable increase in computation time. Numerical experiments are conducted to evaluate the effectiveness of the developed DKRH and S-DKRH algorithms based on real-life data from the Beijing urban rail transit network. The simulation results indicate that DKRH can be used to achieve real-time train scheduling for the urban rail transit network, while S-DKRH can handle the uncertainty in the passenger flows with an acceptable sacrifice in computation time.**

***Index Terms*—Urban rail transit networks, Time-dependent passenger origin-destination demands, Uncertain passenger flows, Distributed model predictive control, Scenario approach.**

I. Introduction

URBAN rail transit plays an increasingly prominent role in public transportation of big cities due to its stability, high transport capacity, and energy efficiency. Real-time train scheduling is recognized as an effective way to improve passenger satisfaction and to reduce the operational costs under the infrastructure limitations of urban rail transit networks. With the rapid expansion of network scale and the growing passenger demands in urban rail transit systems, it becomes increasingly challenging to achieve real-time train scheduling while considering uncertain time-dependent passenger origin-destination (OD) demands and operational costs.

This research has received funding from the National Natural Science Foundation of China (No. 72071016) and the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (Grant agreement No. 101018826 - CLariNet). The work of the first author is also supported by the China Scholarship Council under Grant 202007090003.

Xiaoyu Liu, Azita Dabiri, and Bart De Schutter are with the Delft Center for Systems and Control, Delft University of Technology, 2628CD Delft, The Netherlands (email: x.liu-20@tudelft.nl, a.dabiri@tudelft.nl, b.deschutter@tudelft.nl).

Yihui Wang is with the State Key Laboratory of Advanced Rail Autonomous Operation, Beijing Jiaotong University, Beijing 100044, China (email: yihui.wang@bjtu.edu.cn).

A. Passenger-Oriented Train Scheduling for a Single Line

Several methods are reported in the literature to optimize arrival and departure times of trains at each platform in a single line. One important trend is to develop more practically implementable train scheduling strategy by including more attributes of train operation and infrastructure restrictions, e.g., train speed profiles [1], [2], rolling stock circulation [3], [4], train stopping plan [5]. Wang *et al.* [1] explored the train scheduling problem of a metro line while taking train capacity and speed profiles into account, and then an iterative convex programming approach is proposed to solve the resulting nonlinear nonconvex optimization problem. Shi *et al.* [6] investigated a flexible train capacity allocation strategy for a metro line where carriages are reserved for different stations based on time-dependent passenger demands, and the resulting nonlinear integer programming problem is solved through a variable neighborhood search algorithm. Zhou *et al.* [7] incorporated rolling stock circulation into the train scheduling problem considering passenger demands on a tidal oversaturated metro line, so that passenger demands in different phases can be satisfied.

The above studies are limited to passenger-oriented train scheduling problems of a single line. For an urban rail transit network, different lines typically interact with each other through transfer passengers. Therefore, train scheduling considering detailed passenger origin-destination (OD) demands in urban rail transit networks is regarded as an important direction to further improve passenger satisfaction [8].

B. Passenger-Oriented Train Scheduling for Networks

Train scheduling in urban rail transit networks with time-dependent passenger OD demands is challenging due to the requirement for network coordination and the scale of the resulting problem. In order to minimize the energy consumption of trains and the total travel time of passengers, Wang *et al.* [9] formulated time-dependent passenger OD demands in an urban rail transit network by an event-driven model, where arrival events, departure events, and passenger arrival rates change events are proposed to describe the movement of trains and passengers. Yin *et al.* [10] proposed a mixed-integer linear programming (MILP) formulation to handle the overcrowdedness of stations in an urban rail transit network, and a decomposition-based adaptive large neighborhood search approach was developed to improve the computational efficiency. Luan and Corman [11] included the train scheduling

and passenger routing process in an integrated model, and the resulting mixed-integer nonlinear programming (MINLP) problem is reformulated as an MILP problem to minimize passenger disutility (i.e., passenger delay, travel time, and the number of stranded passengers) and total train delay.

Considering the computational complexity of explicitly integrating departure and arrival times in an urban rail network with time-dependent passenger OD demands, optimizing train departure frequencies of each line has become a promising direction in passenger-oriented train scheduling [12], [8]. Canca *et al.* [8] optimized line frequencies and capacities by solving an MINLP problem. Liu *et al.* [13] developed a novel passenger flow model to determine train departure frequencies, i.e., the number of trains per unit time in each line, where time-dependent passenger OD demands and train capacities are included. The resulting optimization problem can be exactly transformed into an MILP problem, which can be solved efficiently by state-of-the-art MILP solvers [13]. Nevertheless, most existing studies in passenger-oriented train scheduling of urban rail networks do not include rolling stock availability due to the computational complexity issue, leaving an open gap for generating a practically implementable timetable.

C. MPC for Real-Time Railway Traffic Management

The train scheduling problem is a typical constrained control problem [14]. Model predictive control (MPC) is a methodology for addressing real-time constrained control problems [15], [16]. Based on a switching max-plus-linear model, a real-time train scheduling method was developed in [17] to minimize train delays and operational costs. Caimi *et al.* [18] dealt with train rescheduling problems for complex railway station areas by using MPC. However, as it is an optimization-based control approach, centralized MPC can be difficult to implement in real-life railway networks because of its computational complexity and global information requirements. These issues become more challenging in the case of large-scale networks.

For general large-scale systems, many researchers have been developing non-centralized methods that coordinate subsystems in a decentralized, distributed, or hierarchical manner to achieve fast and effective solutions for the overall system [19], [20], [21]. Non-centralized control methods have also been used in railway train scheduling problems. Kersbergen *et al.* [22] developed several distributed MPC methods for the railway traffic management problem where the arrival and departure times, breaking connections, and train orders in the railway network were jointly optimized. Luan *et al.* [23] applied three distributed optimization approaches, i.e., an alternating direction method of multipliers approach, a priority-rule-based approach, and a cooperative distributed robust safe but knowledgeable (CDRSBK) algorithm for real-time traffic management of railway networks. Numerical experiments show that the CDRSBK approach with train-based decomposition performs best on the basis of feasibility, optimality, and computational efficiency.

The train scheduling of urban rail transit networks with time-dependent passenger OD demands is challenging because of the large computational burden. The advanced non-centralized control methods [19], [20], [21] and their successful applications in railway [22], [23] have open opportunities to develop a new efficient distributed MPC method for passenger-oriented train scheduling problems.

D. Train Scheduling under Uncertainties

There are many uncertain attributes in railway networks, e.g., uncertain passenger flows and uncertain delays, that could influence the performance of train schedules. Cacchiani *et al.* [24] developed three different MILP formulations based on light robustness (where uncertainty is handled by inserting different protection levels) to reduce passenger inconvenience caused by uncertain passenger demands in a high-speed railway line. The scenario approach [25], [26] is a general data-driven decision-making methodology that can deal with uncertainties of a system. The scenario approach typically captured uncertainties by a collection of representative scenarios, and the decision is then made by considering these representative scenarios. By using different scenarios to capture the uncertain train operation time in the network, Yang *et al.* [27] developed a two-stage stochastic integer programming model to minimize the expected passenger travel time and transfer activities, where the potential transfer stations are found at the first stage while the least time paths are provided at the second stage. Gong *et al.* [28] formulated an MINLP problem to optimize the operational costs on an urban rail transit line where passenger distribution is represented via several different scenarios. However, most research only considered uncertain passenger demands for a single line. Passenger demands in urban rail transit networks exhibit highly dynamic and random characteristics because trains typically operate with high density, and passengers can choose different routes and different trains to reach their destinations. Therefore, efficient approaches that can explicitly include uncertain passenger demands in urban rail networks still require further research.

E. Paper Contributions and Structure

The current paper deals with the real-time train scheduling problem considering uncertain time-dependent passenger origin-destination demands in urban rail transit networks. By extending the passenger absorption model developed in [13], the train scheduling problem with rolling stock availability can be addressed by using model predictive control where the optimization problem at each time step is formulated as a mixed-integer linear programming problem. Considering the computational issues, we develop a distributed model predictive control approach where each line is regarded as one subsystem. Furthermore, as passenger flows generally exhibit some degree of uncertainty, a scenario-based approach is incorporated into the distributed model predictive control approach to deal with these uncertainties.

The main contributions of the paper are as follows:

1) A novel distributed-knowledgeable-reduced-horizon (DKRH) algorithm is developed for the train scheduling problem, where a new cost-to-go function is proposed

considering computational complexity, prediction horizon, and future performance.
2) We incorporate a scenario-based distributed control scheme into the DKRH algorithm, and a scenario-based distributed-knowledgeable-reduced-horizon algorithm is developed to handle uncertain passenger flows in large-scale urban rail transit networks.
3) The passenger absorption model of [13] is extended to include rolling stock availability by taking into account the total number of available trains so as to generate practically implementable control strategies.

The remaining part of the paper is organized as follows: Section II introduces the mathematical model used in this paper. In Section III, a distributed knowledgeable-reduced-horizon algorithm is developed. In Section IV, we propose a scenario-based distributed knowledgeable-reduced-horizon algorithm. In Section V, the effectiveness of the developed approaches is evaluated based on real-life data from a part of the Beijing urban rail transit network. The paper is concluded with final remarks in Section VI.

II. Mathematical Model

This section starts with the description of the mathematical model proposed by the authors in [13], followed by an extension of the model to include rolling stock availability. Some general explanations for the research problem of this paper are as follows:

1) This paper aims to adjust train schedules for urban rail transit networks online based on real-time passenger demands. We assume the routes of passengers are given a priori. Disturbances and disruptions are not in the scope of this paper.
2) The current paper is based on the passenger absorption model developed in [13], which has been developed to determine train departure frequencies (i.e., the number of trains departing from each platform per unit time) for urban rail transit networks.
3) After obtaining the departure frequency of each platform, a dedicated lower-level controller [29] can determine the detailed departure and arrival times of trains, where the departure interval during each phase is determined according to the corresponding departure frequency.

We start with introducing the notations for the model formulation in Section II-A. Then, the passenger absorption model is summarized in Section II-B. Section II-C introduces the constraints for the model and further extends the model to include rolling stock availability.

A. Notations

Indices and Input Parameters

o, d	Index of stations, $o, d \in \mathcal{S}$, $\mathcal{S}$ is the set of stations
p	Index of platforms, $p \in \mathcal{P}$, $\mathcal{P}$ is the set of platforms
k	Index of phases
$\mathrm{s}^{\mathrm{pla}}(p)$	Succeeding platform of platform p
$\mathrm{p}^{\mathrm{pla}}(p)$	Preceding platform of platform p
T	Length of a phase
$h_p^{\min}$	Minimum departure-arrival headway at platform p
$\tau_p^{\min}$	Minimum dwell time of train at platform p
r_p	Average running time of trains from platform p to its succeeding platform
γ_p	Average time for a train from the first platform of a line to platform p
C_{train}	Maximum capacity of a train
$\alpha_{p,d}(k)$	Fraction of passengers absorbed by trains at platform p with destination d during phase k
$\chi_{p,q,d}$	Proportion of passengers transferring from platform p to q with station d as their destination
$t_{p,q}^{\mathrm{transf}}$	Average time for passengers walking from platform p to platform q
$\rho_{o,d}^{\mathrm{station}}(k)$	Passenger origin-destination demands with o as origin station and d destination station during phase k
$\lambda_{o,p,d}(k)$	Proportion of passengers at origin station o choosing platform p for their travel to destination d

Decision variables

$f_p(k)$	The number of trains departing from platform p during phase k

Output variables

$\rho_{p,d}(k)$	Passenger arrival rate at platform p with station d as destination during phase k
$n_{p,d}(k)$	Number of passengers waiting at platform p with station d as their destination at the beginning of phase k
$n_{p,d}^{\mathrm{absorb}}(k)$	Number of passengers at platform p with station d as their destination absorbed by trains during phase k
$C_p(k)$	Total remaining capacity of trains visiting platform p during phase k
$n_p^{\mathrm{want}}(k)$	Total number of passengers that want to board trains at platform p during phase k
$n_{p,d}^{\mathrm{on-board}}(k)$	Number of passengers on board of trains, when trains arrive at platform p, with destination d during phase k
$n_{p,d}^{\mathrm{alight}}(k)$	Number of passengers alighting from trains at platform p with station d as their destination during phase k
$n_{p,q,d}^{\mathrm{transf}}(k)$	Number of passengers transferring from platform p to q with station d as their destination during phase k
$n_{p,d}^{\mathrm{trans,arrive}}(k)$	Number of transfer passengers arriving at platform p with station d as their destination during phase k
$g_p(k)$	Total number of transfer passengers arriving at platform p during phase k
$n_{p,d}^{\mathrm{depart}}(k)$	Number of passengers departing from platform p with station d as destination during phase k
$m_p(k)$	Total number of passengers departing from platform p during phase k

B. Passenger Absorption Model

In the passenger absorption model, the number of passengers $n_{p,d}(k)$ waiting at platform p with station d as their destination at the start of each phase is updated by:

$$\begin{aligned} n_{p,d}(k+1) =& n_{p,d}(k) + \rho_{p,d}(k)\,T \\ &+ n_{p,d}^{\mathrm{trans,arrive}}(k) - n_{p,d}^{\mathrm{absorb}}(k), \end{aligned} \tag{1}$$

where $\rho_{p,d}(k)$ is the average passenger arrival rate at platform p with station d as their destination during phase k; T is the length of a phase; $n_{p,d}^{\mathrm{trans,arrive}}(k)$ is the number of transfer passengers arriving at platform p with destination d during

phase k, and $n_{p,d}^{\mathrm{absorb}}(k)$ is the number of passengers at platform p with destination d absorbed by trains during phase k. Then, $\rho_{p,d}(k)$, $n_{p,d}^{\mathrm{trans,arrive}}(k)$, and $n_{p,d}^{\mathrm{absorb}}(k)$ can be computed by

$$\rho_{p,d}(k) = \lambda_{o,p,d}(k)\,\rho_{o,d}^{\mathrm{station}}(k), \forall p \in \mathcal{P}_o^{\mathrm{sta}}, \tag{2}$$

$$n_{p,d}^{\mathrm{trans,arrive}}(k) = \sum_{q\in \mathrm{cop}(p)\backslash\{p\}} \left(\frac{T - t_{q,p}^{\mathrm{transf}}}{T} n_{q,p,d}^{\mathrm{transf}}(k) + \frac{t_{q,p}^{\mathrm{transf}}}{T} n_{q,p,d}^{\mathrm{transf}}(k-1) \right), \tag{3}$$

$$n_{p,d}^{\mathrm{absorb}}(k) = \alpha_{p,d}(k) n_p^{\mathrm{absorb}}(k), \tag{4}$$

where $\rho_{o,d}^{\mathrm{station}}(k)$ denotes passenger origin-destination demands at phase k with o and d as the origin station and the destination station, respectively; $\mathcal{P}_o^{\mathrm{sta}}$ defines a set of platforms at station o; and $\lambda_{o,p,d}(k)$ is the proportion[1] of passengers at station o who choose platform p for their travel to destination d; $\mathrm{cop}(p)$ defines a set of platforms located at the same station as platform p; $t_{q,p}^{\mathrm{transf}}$ denotes the average transfer time for passengers from platform q to platform p; $n_{p,q,d}^{\mathrm{transf}}(k)$ is the number of passengers transferring from platform p to platform q with destination d during phase k; $\alpha_{p,d}(k)$ is the fraction of passengers absorbed by trains at platform p with destination d during phase k; $n_p^{\mathrm{absorb}}(k)$ denotes the total number of passengers absorbed by trains at platform p during phase k.

For the variable $n_p^{\mathrm{absorb}}(k)$ in (4), we have

$$n_p^{\mathrm{absorb}}(k) = \min\left(n_p^{\mathrm{want}}(k),\ C_p(k)\right), \tag{5}$$

$$n_p^{\mathrm{want}}(k) = n_p(k) + \rho_p(k)\,T + g_p(k), \tag{6}$$

$$C_p(k) = f_p(k)\cdot C_{\mathrm{train}} - \sum_{d\in\mathcal{S}}\left(n_{p,d}^{\mathrm{on-board}}(k) - n_{p,d}^{\mathrm{alight}}(k)\right), \tag{7}$$

with

$$\begin{aligned} n_p(k) &= \sum_{d\in\mathcal{S}} n_{p,d}(k), \quad \rho_p(k) = \sum_{d\in\mathcal{S}} \rho_{p,d}(k), \\ g_p(k) &= \sum_{d\in\mathcal{S}} n_{p,d}^{\mathrm{arrive,transf}}(k), \end{aligned} \tag{8}$$

where $n_p^{\mathrm{want}}(k)$ is the total number of passengers that want to board trains at platform p during phase k; $C_p(k)$ is the total remaining capacity of trains that visit platform p during phase k; $f_p(k)$ is the number of trains that visit platform p during phase k; C_{train} is the maximum capacity of a train, $\mathcal{S}$ denotes the set of stations in the urban rail transit network, $n_{p,d}^{\mathrm{on-board}}(k)$ is the number of passengers on board of trains at platform p with destination d during phase k, and $n_{p,d}^{\mathrm{alight}}(k)$ is the number of passengers alighting from trains at platform p with destination d during phase k.

The number of passengers $n_{p,d}^{\mathrm{depart}}(k)$ departing from platform p with destination d during phase k is

$$n_{p,d}^{\mathrm{depart}}(k) = n_{p,d}^{\mathrm{on-board}}(k) - n_{p,d}^{\mathrm{alight}}(k) + n_{p,d}^{\mathrm{absorb}}(k), \tag{9}$$

[1]As passenger route choices observed from metro data collection systems typically exhibit consistent patterns, we assume that the proportions of passengers choosing each route are given a priori. Thus, $\lambda_{o,p,d}(k)$ can be estimated from historical data or obtained according to the shortest paths.

and we have

$$n_{p,d}^{\mathrm{on-board}}(k) = \frac{T - r_{\mathrm{p^{pla}}(p)}}{T} n_{\mathrm{p^{pla}}(p),d}^{\mathrm{depart}}(k) + \frac{r_{\mathrm{p^{pla}}(p)}}{T} n_{\mathrm{p^{pla}}(p),d}^{\mathrm{depart}}(k-1), \tag{10}$$

$$n_{p,d}^{\mathrm{alight}}(k) = \begin{cases} \sum\limits_{q\in\mathrm{cop}(p)/\{p\}} n_{p,q,d}^{\mathrm{transf}}(k), & \text{if } d \in \mathcal{S}\backslash\{\mathrm{sta}(p)\}, \\ n_{p,d}^{\mathrm{on-board}}(k), & \text{if } d = \mathrm{sta}(p), \end{cases} \tag{11}$$

$$n_{p,q,d}^{\mathrm{transf}}(k) = \chi_{p,q,d}\, n_{p,d}^{\mathrm{on-board}}(k), \quad \forall q \in \mathrm{cop}(p)\backslash\{p\}, \tag{12}$$

$$m_p(k) = \sum_{d\in\mathcal{S}} n_{p,d}^{\mathrm{depart}}(k), \tag{13}$$

where $r_{\mathrm{p^{pla}}(p)}$ refers to the average running time of trains from the preceding platform $\mathrm{p^{pla}}(p)$ to platform p, and $T \gg r_{\mathrm{p^{pla}}(p)}$; $\mathrm{sta}(p)$ defines the station of platform p; $\chi_{p,q,d}$ is the proportion for passengers transferring from platform p to $q \in \mathrm{cop}(p)$ with station d as their destination; $\mathrm{cop}(p)$ defines a set of platforms located at the same station as platform p; $m_p(k)$ denotes the total number of passengers departing from platform p during phase k.

C. Constraints for the Absorption Model

1) Departure Frequency Constraints: In this paper, we only consider the case that each line has one depot to accommodate trains. In general, each train at a line will visit every platform of the line before it returns to deport or starts as a new train service. In this context, the number of trains running on a line can be determined by the number of trains departing from the depot. Therefore, the number of trains $f_p(k)$ departing from platform p can be calculated by

$$f_p(k) = \frac{T-\phi_p}{T} f_{\mathrm{fst}(p)}(k-\beta_p) + \frac{\phi_p}{T} f_{\mathrm{fst}(p)}(k-\beta_p-1), \tag{14}$$

$$\beta_p = \lfloor \gamma_p / T \rfloor, \quad \phi_p = \gamma_p - \beta_p T, \tag{15}$$

where $\mathrm{fst}(p)$ defines the first platform of the line corresponding to platform p, i.e., the platform connected with the depot of the line, γ_p denotes the average time for a train from platform $\mathrm{fst}(p)$ to platform p.

To ensure the safe operation of urban rail transit systems, the number of trains departing from platform p during phase k should be constrained by

$$f_p(k)\left(h_p^{\min} + \tau_p^{\min}\right) \le T, \tag{16}$$

where $h_p^{\min}$ and $\tau_p^{\min}$ are the minimum headway and the minimum dwell time at platform p, respectively.

2) Rolling Stock Availability Constraints: In real-life operations, the number of trains used for each line is restricted by the total number of available trains, i.e., the total number of trains running on the line should be smaller than or equal to the total number of available trains. Therefore, the rolling stock availability should be included in order to generate a practically implementable timetable. Considering p as the platform connected with a depot, the train departing from platform p typically visits every platform of the line and requires an average time interval c_p to return to the depot, and

we define c_p as the circulation time. Then, for the passenger absorption model, the trains departing from a depot during the circulation time should satisfy

$$f_p(k)+\sum_{i=1}^{\sigma_p-1} f_p(k-i)+\frac{\omega_p}{T}f_p(k-\sigma_p) \le N_p^{\mathrm{rs}}, \quad \forall p \in \mathrm{dep}(p), \tag{17}$$

$$\sigma_p = \lfloor c_p/T \rfloor, \quad \omega_p = c_p - \sigma_p T, \tag{18}$$

where $\mathrm{dep}(p)$ is the set of platforms that use the same depot with platform p; N_p^{rs} is the total number of available trains for the line corresponding to platform p.

III. Distributed Knowledgeable-Reduced-Horizon Algorithm for Train Scheduling

Based on the model predictive control (MPC) framework, in this section, we first develop a knowledgeable-reduced-horizon (KRH) approach where a novel cost-to-go function is designed to shorten the prediction horizon. A distributed control framework is then proposed to further reduce the computational burden of solving the MPC optimization problem, thereby achieving real-time train scheduling in the urban rail transit network. In the distributed control framework, each local agent generates its control decisions based on its local information and information from its neighbor agents. Such a framework is in accordance with the real-life situation where global information is typically not available in large-scale urban rail transit networks.

A. Problem Formulation in MPC Set-Up

In an urban rail transit network, passenger satisfaction is strongly related to the total time spent in the network. Based on the absorption model, the total travel time of passengers in the urban rail transit network during phase k is represented by

$$J^{\mathrm{pass}}(k) = \sum_{p\in\mathcal{P}} \big(n_p(k)T + m_p(k)r_p + g_p(k)t_p^{\mathrm{transf}}\big), \tag{19}$$

where $n_p(k)T$ represents the total waiting time at platform p during phase k, $m_p(k)r_p$ denotes the total running time until the next platform for passengers departing from platform p during phase k, and $g_p(k)t_p^{\mathrm{transf}}$ represents the total transfer time of passengers at platform p during phase k.

The operational cost of an urban rail transit system is highly related to the energy consumption of trains. Based on the absorption model, the total energy consumption for trains departing from the platform during phase k is computed by

$$J^{\mathrm{roll}}(k) = \sum_{p\in\mathcal{P}} f_p(k)E_p, \tag{20}$$

where E_p represents the average energy consumption for a train to run from platform p to its succeeding platform.

Therefore, the MPC optimization problem $\mathbf{P}_{k_0}^{\mathrm{MPC}}$ for real-time train scheduling of urban rail transit networks is formulated as

$$\begin{aligned} &\min_{\mathbf{f}(k)} J(k_0) = \sum_{k=k_0}^{k_0+N_0-1} \big(J^{\mathrm{pass}}(k) + \xi J^{\mathrm{roll}}(k)\big), \\ &\text{subject to} \quad (1)-(14),\ (16)-(17), \end{aligned} \tag{21}$$

where N_0 is the prediction horizon, and ξ is a weight balancing the objectives.

As explained in [13], the nonlinear optimization problem $\mathbf{P}_{k_0}^{\mathrm{MPC}}$ can be transformed into a mixed-integer linear programming (MILP) problem $\mathbf{P}_{k_0}^{\mathrm{MILP}}$ with the following form, which is exactly equivalent to the original optimization problem:

$$\min_{\substack{\boldsymbol{x}(k),\boldsymbol{f}(k)\\ \boldsymbol{\delta}(k),\boldsymbol{z}(k)}} J(k_0) = \sum_{k=k_0}^{k_0+N_0-1} \big(J^{\mathrm{pass}}(k) + \xi J^{\mathrm{roll}}(k)\big) \tag{22}$$

subject to

$$\boldsymbol{x}(k+1) = A_k\boldsymbol{x}(k)+B_{1,k}\boldsymbol{f}(k)+B_{2,k}\boldsymbol{\delta}(k)+B_{3,k}\boldsymbol{z}(k), \tag{23}$$

$$E_{2,k}\boldsymbol{\delta}(k) + E_{3,k}\boldsymbol{z}(k) \le E_{1,k}\boldsymbol{f}(k) + E_{4,k}\boldsymbol{x}(k) + E_{5,k}, \tag{24}$$

$$\boldsymbol{f}(k) \le D_0 + \sum_{i=1}^{K} D_i\boldsymbol{f}(k-i), \tag{25}$$

$$k = k_0, \ldots, k_0+N_0-1,$$

where $\boldsymbol{x}(k)$ and $\boldsymbol{f}(k)$ respectively concatenate the state variables (i.e., the variables related to the passengers) and decision variables (i.e., the number of trains) of all platforms in the network in phase k; $\boldsymbol{\delta}(k)$ and $\boldsymbol{z}(k)$ respectively represent the vector of auxiliary binary variables and auxiliary continuous variables in phase k. The compact equation (23) represents the linear and mixed-integer linear formulations of the equations in (1)-(14). Constraint (24) collects all the linear and mixed-integer linear model constraints and operational constraints in a matrix form. Constraint (25) collects the constraints of decision variables, i.e., (14) and (17), in a matrix form, where $K = \max_{p\in\mathcal{P}} \sigma_p$.

For detailed information of transforming nonlinear terms of the model into mixed-integer linear inequalities, we refer the interested readers to [30], [31].

B. Knowledgeable-Reduced-Horizon Algorithm for Train Scheduling

The computational complexity of solving MILP problem $\mathbf{P}_{k_0}^{\mathrm{MILP}}$ increases rapidly with the prediction horizon N_0 due to the increasing number of variables. Solving $\mathbf{P}_{k_0}^{\mathrm{MILP}}$ at every MPC step is not tractable for large prediction horizons because of the real-time feasibility restriction. Shortening the prediction horizon to reduce the computational burden; however, a short prediction horizon may negatively affect the performance of the controller as less future information can be included in the decision-making process.

Inspired by the robust-safe-but-knowledgeable (RSBK) algorithm proposed in [20], [32], we develop a knowledgeable-reduced-horizon (KRH) algorithm to shorten the prediction horizon of the original MPC controller by a customized cost-to-go function. The optimization problem $\mathbf{P}_{k_0}^{\mathrm{KRH}}$ for the KRH algorithm is defined as

$$\min_{\substack{\boldsymbol{x}(k),\boldsymbol{f}(k)\\ \boldsymbol{\delta}(k),\boldsymbol{z}(k)}} J(k_0) = \sum_{k=k_0}^{k_0+N-1} \Big(J^{\mathrm{pass}}(k) + \xi J^{\mathrm{roll}}(k)\Big) + L_N(k_0) \tag{26}$$

subject to

$$\boldsymbol{x}(k+1) = A_k\boldsymbol{x}(k)+B_{1,k}\boldsymbol{f}(k)+B_{2,k}\boldsymbol{\delta}(k)+B_{3,k}\boldsymbol{z}(k), \quad (27)$$

$$E_{2,k}\boldsymbol{\delta}(k) + E_{3,k}\boldsymbol{z}(k) \leq E_{1,k}\boldsymbol{f}(k) + E_{4,k}\boldsymbol{x}(k) + E_{5,k}, \quad (28)$$

$$\boldsymbol{f}(k) \leq D_0 + \sum_{i=1}^{K} D_i\boldsymbol{f}(k-i), \quad (29)$$

$$k = k_0, \ldots, k_0 + N - 1,$$

where $L_N(k_0)$ denotes the cost-to-go function associated with the terminal states of passengers at the end of the shortened horizon.

As the target of the controller is to minimize the total travel time of the passengers, the cost-to-go function is designed to determine the cost associated with the passengers that remain at the platforms at the end of the reduced prediction window, i.e., a reasonable estimate of the remaining travel time for passengers waiting at the platforms at the end of the prediction time window.

The cost-to-go function for the remaining passengers at the platforms is defined as:

$$L_N(k_0) = \sum_{p\in\mathcal{P}}\sum_{d\in\mathcal{S}}\Big(n_{p,d}(k_0+N)\sum_{j\in\mathcal{R}_{p,d}}\eta_{p,d,j}t^{\text{total}}_{p,d,j}\Big), \quad (30)$$

where $\mathcal{R}_{p,d}$ represents the set of possible routes for passengers from platform p to their destination d (see *Remark 3.1* below for an example of $\mathcal{R}_{p,d}$), $\eta_{p,d,j}$ is defined as the percentage of passengers at platform p that will travel to station d through route j, and we have

$$\eta_{p,d,j} = \prod_{(q,q')\in\mathcal{P}^{\text{pair}}_j}\chi_{q,q',d}, \forall j \in \mathcal{R}_{p,d}, \quad (31)$$

where $\mathcal{P}^{\text{pair}}_j$ represents the set of platform pairs at a transfer station in route j, and $\chi_{q,q',d}$ is the proportion for passengers transferring from platform q to q'. As the route of passengers can be represented by several pairs of platforms, (31) calculates the percentage of passengers that intend to travel from p to d through route j. Since $\chi_{q,q',d}$ is estimated based on historical data, $\eta_{p,d,j}$ can be calculated offline.
Then, $t^{\text{total}}_{p,d,j}$ represents the average travel time for passengers from platform p to their destination d through route j, and $t^{\text{total}}_{p,d,j}$ can be calculated offline based on the average dwell times, the average running times, and the average transfer times related to the platforms in route j:

$$t^{\text{total}}_{p,d,j} = t^{\text{avrg}}_{p,j} + \sum_{(q,q')\in\mathcal{P}^{\text{pair}}_j}\Big(t^{\text{transf}}_{q,q'} + t^{\text{avrg}}_{q',j}\Big), \forall j \in \mathcal{R}_{p,d}, \quad (32)$$

where $t^{\text{avrg}}_{p,j}$ denotes the average time for passengers from platform p to reach either the next transfer station or the destination station in route j.

The construction of the sets $\mathcal{R}_{p,d}$ and $\mathcal{P}^{\text{pair}}_j$ is now illustrated in *Remark 3.1* through an example.

Remark 3.1 An example network is shown in Fig. 1. For passengers waiting at platform **a** with destination **h** at the end of the prediction window, there are two possible routes in the example network of Fig. 1. Thus, the set of possible routes for passengers from platform **a** with destination **h** is $\mathcal{R}_{\mathbf{a},\mathbf{h}} = \{\mathbf{a}-\mathbf{b}_1-\mathbf{b}_2-\mathbf{f}-\mathbf{g}_1-\mathbf{h},\ \mathbf{a}-\mathbf{b}_1-\mathbf{c}_1-\mathbf{c}_2-\mathbf{g}_2-\mathbf{g}_1-\mathbf{h}\}$. The set of platform pairs for route $\mathbf{a}-\mathbf{b}_1-\mathbf{b}_2-\mathbf{f}-\mathbf{g}_1-\mathbf{h}$ (named as route 1) is $\mathcal{P}^{\text{pair}}_1 = \{(\mathbf{b}_1,\mathbf{b}_2)\}$, and the set of platform pairs for route $\mathbf{a}-\mathbf{b}_1-\mathbf{c}_1-\mathbf{c}_2-\mathbf{g}_2-\mathbf{g}_1-\mathbf{h}$ (named as route 2) is $\mathcal{P}^{\text{pair}}_2 = \{(\mathbf{c}_1,\mathbf{c}_2),(\mathbf{g}_2,\mathbf{g}_1)\}$. Then, the corresponding cost-to-go function can be calculated according to (30)-(32).

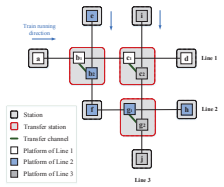

Fig. 1. Example network

Comparing $\mathbf{P}^{\text{KRH}}_{k_0}$ and $\mathbf{P}^{\text{MPC}}_{k_0}$, we can find that the number of variables and constraints in (27) and (28) are reduced as the prediction horizon is reduced from N_0 to N. Similar to $\mathbf{P}^{\text{MILP}}_{k_0}$, the optimization problem $\mathbf{P}^{\text{KRH}}_{k_0}$ for the KRH algorithm is an MILP problem.

C. Distributed KRH Algorithm for Train Scheduling in Urban Rail Transit Network

For large-scale urban rail transit networks, it may not be feasible to solve problem $\mathbf{P}^{\text{KRH}}_{k_0}$ in a centralized manner due to the computational burden and the communication restrictions for collecting global information. In the urban rail transit network, different lines typically interact with their neighbor lines through transfer passengers as described in (3). In this section, a distributed-knowledgeable-reduced-horizon (DKRH) algorithm is developed for passenger-oriented real-time train scheduling of urban rail transit networks.

In urban rail transit networks, we can regard each line as a subsystem, where different subsystems interact with each other through transfer passengers. The corresponding objective functions associated with the travel time and energy consumption of subsystem l during phase k are

$$J^{\text{pass}}_l(k) = \sum_{p\in\mathcal{P}^{\text{line}}_l}\big(n_p(k)T + m_p(k)r_p + g_p(k)t^{\text{transf}}_p\big), \quad (33)$$

$$J^{\text{roll}}_l(k) = \sum_{p\in\mathcal{P}^{\text{line}}_l} f_p(k)E_p, \quad (34)$$

where $\mathcal{P}^{\text{line}}_l$ is the set of platforms of line l. The cost-to-go functions corresponding to the terminal states of passengers of subsystem l is

$$L_{N,l}(k_0) = \sum_{p\in\mathcal{P}^{\text{line}}_l}\sum_{d\in\mathcal{S}}\Big(n_{p,d}(k_0+N)\sum_{j\in\mathcal{R}_{p,d}}\eta_{p,d,j}t^{\text{total}}_{p,d,j}\Big). \quad (35)$$

The proposed DKRH algorithm is an iterative algorithm. In every control step of the proposed DKRH algorithm, different subsystems exchange information with their neighbor several times over several iterations. In each iteration, different subsystems solve their local problems in parallel, and then

they exchange the new computed solution for the next iteration until the stopping criterion is met. At iteration step ϑ, the l-th subsystem calculates its control inputs through the following optimization problem, denoted as $\mathbf{P}^{\mathrm{D}}_{l,k_0}$, by setting the variables of other subsystems as the corresponding values of the last iteration $\vartheta-1$:

$$\min_{\substack{\boldsymbol{\delta}_l(k),\boldsymbol{f}_l(k)\\ \boldsymbol{x}_l(k),\boldsymbol{z}_l(k)}} J_l(k_0)=\sum_{k=k_0}^{k_0+N-1}\Big(J_l^{\mathrm{pass}}(k)+\xi J_l^{\mathrm{roll}}(k)\Big)+L_{N,l}(k_0) \tag{36}$$

subject to

$$\boldsymbol{x}_l(k+1)=A_{l,k}\boldsymbol{x}_l(k)+B_{1l,k}\boldsymbol{f}_l(k)+B_{2l,k}\boldsymbol{\delta}_l(k)+B_{3l,k}\boldsymbol{z}_l(k), \tag{37}$$

$$E_{2l,k}\boldsymbol{\delta}_l(k)+E_{3l,k}\boldsymbol{z}_l(k)\le E_{1l,k}\boldsymbol{f}_l(k)+E_{4l,k}\boldsymbol{x}_l(k)+E_{5l,k}, \tag{38}$$

$$\boldsymbol{f}_l(k)\le D_{l,0}+\sum_{i=1}^{K}D_{l,i}\boldsymbol{f}_l(k-i), \tag{39}$$

$$k=k_0,\dots,k_0+N-1.$$

Algorithm 1 describes the DKRH algorithm, where $l_{\max}$ is the total number of lines in the network; ε is a small positive value which can be the machine precision. An initial estimate for the decision variable can be that of the basic timetable, i.e., the timetable with regular departure frequencies, which is typically used in the daily operation. As each line is independent from the other lines, i.e., they do not share track and/or platforms with the other lines, trains in different lines will not conflict with each other. In this context, the regular departure frequencies are always feasible.

Algorithm 1 DKRH for real-time train scheduling

Input: k_{end}; $\vartheta_{\max}$; $l_{\max}$; ε; initial estimate for the decision variable: $\boldsymbol{f}_l^0(k)$, $l=1,\dots,l_{\max}$;
Output: optimal value $\boldsymbol{f}_l(k)$, J_l

repeat
  $k\leftarrow k_0$
  repeat
    $\vartheta\leftarrow 1$
    for $l=1,\dots,l^{\max}$ **do**
      solve problem $\mathbf{P}^{\mathrm{D}}_{l,k_0}$ and get $\boldsymbol{f}_l^{\vartheta}(k)$ and J_l^{ϑ}
      update (37), (38), and (39) for l by using $\boldsymbol{f}_l^{\vartheta}(k)$
    end for
    $\vartheta\leftarrow\vartheta+1$
  until $\vartheta=\vartheta_{\max}$ or $\left|J_l^{\vartheta}-J_l^{\vartheta-1}\right|\le\varepsilon$
  apply control decision $\boldsymbol{f}_l(k)$ to each subsystem l
  $k\leftarrow k+1$
until $k=k_{\mathrm{end}}$

Lemma 3.1: As we start with a feasible solution of the overall system and as the initial values of the decision variables are always feasible at every step, a feasible solution of problem $\mathbf{P}^{\mathrm{D}}_{k_0}$ can always be found.

IV. Scenario-based DKRH Algorithm

In this section, a scenario-based distributed-knowledgeable-reduced-horizon (S-DKRH) algorithm is developed to improve service quality in the presence of uncertain passenger flows.

For a large-scale urban rail transit network, the uncertainties generally consist of global uncertainties (e.g., the uncertainties caused by different weather conditions), and local uncertainties of each subsystems (i.e., the uncertainties due to different line conditions). Both global uncertainties and local uncertainties can be captured as several representative scenarios over the prediction window, which can be defined as global scenarios and local scenarios, respectively, based on historical data [33], [34]. If we include all combinations of global scenarios and local scenarios, the total number of combinations N_{com} is

$$N_{\mathrm{com}}=N_{\mathrm{glo}}\prod_{l=1}^{l_{\max}}N_{\mathrm{loc},l}, \tag{40}$$

where N_{glo} denotes the number of global scenarios; $N_{\mathrm{loc},l}$ is the number of scenarios for subsystem l; $l_{\max}$ is the total number of subsystems in the network; In this context, each subsystem should consider the complete set of scenarios, i.e., N_{com} scenarios, when generating its decision variables, which would rapidly increase the computational burden.

In order to address the computational complexity issue arising from the increasing number of scenarios for urban rail transit networks, we adopt a scenario reduction approach [35] into the DKRH algorithm. For subsystem l, the $N_{\mathrm{loc},l}$ scenarios will be directly used for subsystem l in the scenario-based approach. However, when considering the impact from subsystem l' ($l'\neq l$) on subsystem l, we use the scenario reduction approach to reduce the number of representative scenarios of subsystem l' from $N_{\mathrm{loc},l'}$ to $N_{l',l}$, i.e., $N_{l',l}\ll N_{\mathrm{loc},l'}$. In this context, subsystem l only needs to consider $N_{\mathrm{total},l}=N_{\mathrm{glo}}N_{\mathrm{loc},l}\prod_{l'\neq l}N_{l',l}$ representative scenarios, which can be much smaller than that of original scenario approach with $N_{\mathrm{glo}}\prod_{l=1}^{l_{\max}}N_{\mathrm{loc},l}$ representative scenarios. For example, Fig. 2 has three subsystems, and subsystem 2 only considers $N_{\mathrm{glo}}\cdot N_{\mathrm{loc},2}\cdot N_{1,2}\cdot N_{3,2}$ representative scenarios instead of $N_{\mathrm{glo}}\cdot N_{\mathrm{loc},1}\cdot N_{\mathrm{loc},2}\cdot N_{\mathrm{loc},3}$ representative scenarios, where $N_{1,2}\ll N_{\mathrm{loc},1}$, $N_{3,2}\ll N_{\mathrm{loc},3}$. Therefore, the computational burden of each subsystem is reduced significantly.

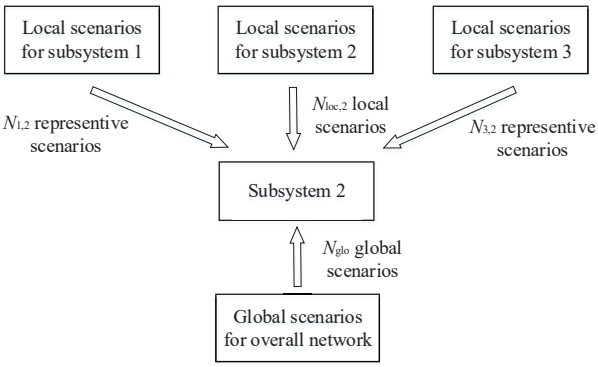

Fig. 2. Reduced scenarios for agent 2 in an example with 3 agents.

Based on the above scenario reduction approach, we develop the S-DKRH algorithm. For subsystem l with scenario s, the

corresponding objective functions are

$$J_{l,s}^{\text{pass}}(k)=\sum_{p\in\mathcal{P}_l^{\text{line}}}\left(n_{p,s}(k)T+m_{p,s}(k)r_p+g_{p,s}(k)t_p^{\text{transf}}\right), \tag{41}$$

$$J_{l,s}^{\text{roll}}(k)=\sum_{p\in\mathcal{P}_l^{\text{line}}}f_p(k)E_{p,s}, \tag{42}$$

where $n_{p,s}(k)$, $m_{p,s}(k)$, $g_{p,s}(k)$, and $E_{p,s}$ respectively represent the values of $n_p(k)$, $m_p(k)$, $g_p(k)$, and E_p under scenario s. The corresponding cost-to-go functions is

$$L_{N,l,s}(k_0)=\sum_{p\in\mathcal{P}_l^{\text{line}}}\sum_{d\in\mathcal{S}}\left(n_{p,d,s}(k_0+N)\sum_{r\in\mathcal{R}_{p,d}}\eta_{p,d,r}t_{p,d,r}^{\text{total}}\right), \tag{43}$$

with $n_{p,d,s}(k_0+N)$ denoting the variable $n_{p,d}(k_0+N)$ under scenario s.

In the S-DKRH algorithm, subsystem l considers only one representative scenario for each neighbor subsystem, and the variables of the neighbor subsystems are set as the corresponding values of the last iteration. At phase k_0, the l-th subsystem generates its control decisions by solving the following chance-constraint optimization problem $\mathbf{P}_{l,k_0}^{\text{S}}$:

$$\min_{\substack{\boldsymbol{x}_l(k),\boldsymbol{f}_l(k)\\ \boldsymbol{\delta}_l(k),\boldsymbol{z}_l(k)}}\sum_{s=1}^{N_{\text{total},l}}\mathbb{P}\{s\}\left(\sum_{k=k_0}^{k_0+N-1}\left(J_{l,s}^{\text{pass}}(k)+\xi J_{l,s}^{\text{roll}}(k)\right)+L_{N,l,s}(k_0)\right) \tag{44}$$

subject to

$$\begin{aligned}\boldsymbol{x}_{l,s}(k+1)=&A_{l,s,k}\boldsymbol{x}_{l,s}(k)+B_{1l,s,k}\boldsymbol{f}_l(k)\\&+B_{2l,s,k}\boldsymbol{\delta}_{l,s}(k)+B_{3l,s,k}^s\boldsymbol{z}_{l,s}(k),\end{aligned} \tag{45}$$

$$\begin{aligned}E_{2l,s,k}^{\text{hard}}\boldsymbol{\delta}_{l,s}(k)+E_{3l,s,k}^{\text{hard}}\boldsymbol{z}_{l,s}(k)\le\\ E_{1l,s,k}^{\text{hard}}\boldsymbol{f}_l(k)+E_{4l,s,k}^{\text{hard}}\boldsymbol{x}_{l,s}(k)+E_{5l,s,k}^{\text{hard}},\end{aligned} \tag{46}$$

$$\begin{aligned}\sum_{s=1}^{N_{\text{total}}}\mathbb{P}\{s\}\mathbf{1}\Big(E_{2l,s,k}^{\text{soft}}\boldsymbol{\delta}_{l,s}(k)+E_{3l,s,k}^{\text{soft}}\boldsymbol{z}_{l,s}(k)\le\\ E_{1l,s,k}^{\text{soft}}\boldsymbol{f}_l(k)+E_{4l,s,k}^{\text{soft}}\boldsymbol{x}_{l,s}(k)+E_{5l,s,k}^{\text{soft}}\Big)\ge\boldsymbol{\theta}_l,\end{aligned} \tag{47}$$

$$\begin{aligned}&\boldsymbol{f}_l(k)\le D_{l,0}^s+\sum_{i=1}^{K}D_{l,i}^s\boldsymbol{f}_l(k-i),\\&k=k_0,\dots,k_0+N-1,\end{aligned} \tag{48}$$

where $\mathbb{P}\{s\}$ denotes the probability of s, and $N_{\text{total},l}$ is the total number of scenarios for agent l after scenario reduction. Eq. (45) represents the linear and mixed-integer linear formulations of the model explained in (1)-(14) for subsystem l under scenario s; (46) collects the corresponding hard constraints; (47) denotes the chance constraints, i.e., the constraints related to operational performance, $\mathbf{1}(\cdot)$ defines the indicator function[2], and $\boldsymbol{\theta}_l\in(0,1)$ indicates the minimally required probability that there is no constraint violation; (48) collects the hard constraints of decision variables, i.e., (14) and (17), for subsystem l with scenario s. By solving $\mathbf{P}_{l,k_0}^{\text{S}}$, we minimize the expected value of objective function (44) while including the corresponding constraint satisfaction in (46). Problem $\mathbf{P}_{l,k_0}^{\text{S}}$ for the S-DKRH algorithm is also an MILP problem and can be solved efficiently by using existing MILP solvers.

[2] $\mathbf{1}(\cdot)=1$ if the corresponding constraint is satisfied, otherwise $\mathbf{1}(\cdot)=0$.

Algorithm 2 provides the process of the S-DKRH algorithm, where $\mathcal{L}_l$ represents the neighboring subsystems of l, i.e., lines connected with line l via transfer stations.

Algorithm 2 S-DKRH algorithm for real-time train scheduling

Input: k_{end}; $\vartheta_{\max}$; N_{glo}; $N_{\text{loc},l}$; $N_{l',l}$; ε; initial estimate for the decision variable: $\boldsymbol{f}_l^0(k)$, $l=1,\dots,l_{\max}$;
Output: optimal value $\boldsymbol{f}_l(k)$, J_l

```
for l = 1, ..., l_max do
    construct N_{local,l} scenarios for local controller
    construct 1 combined scenario for its neighbors
end for
repeat
    k ← k_0
    repeat
        ϑ ← 1
        for l = 1, ..., l^max do
            solve problem (44) and get f_l^ϑ(k) and J_l^ϑ
            update constraints in problem P^S_{l,k_0} for l ∈ L_l
        end for
        ϑ ← ϑ + 1
    until ϑ = ϑ_max or |J_l^ϑ − J_l^{ϑ−1}| ≤ ε
    apply control decision f_l(k) to each subsystem l
    k ← k + 1
until k = k_end
```

V. Case Study

To evaluate the performance of the developed approaches, numerical experiments are conducted based on real-life data of the Beijing urban rail transit network. First, we present the urban rail transit network and some basic settings we use in the case study. Then, simulations are conducted to illustrate the effectiveness of the developed KRH and DKRH algorithms. Finally, we include uncertainty in the passenger flows in the simulation to show the performance of the S-DKRH algorithm.

A. Setup

The network we consider includes four bidirectional lines of the Beijing urban rail transit network, i.e., Changping Line, Line 8, Line 13, and Line 15 (see Fig. 3). Therefore, we have four subsystems for the distributed control approaches. The main parameters for the case study are shown in TABLE I, where the circulation time c_p mentioned in (17) and (18) is estimated based on the average running time and the regular dwell time. The basic timetable is generated by implementing the regular headway and the regular dwell time in TABLE I.

The passenger OD demands are obtained based on the real-life passenger data of the Beijing urban rail transit network. In particular, we use the real-life data on passengers entering and exiting flows of each station in the network of Fig. 3. The data is updated every 30 minutes. In the case study, we consider passenger flows from 7:00-12:00, which includes situations of both peak hours and off-peak hours. We directly use the

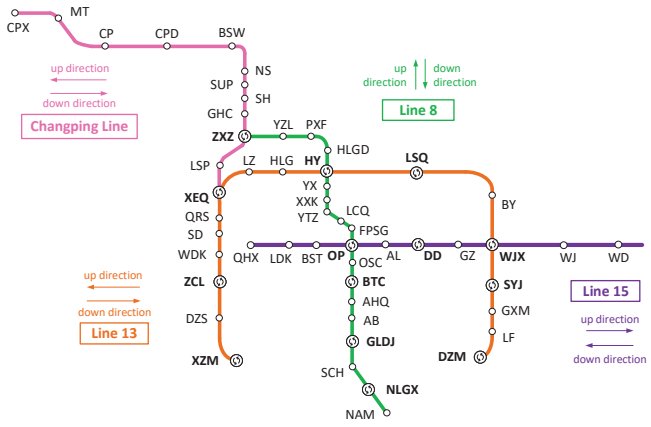

Fig. 3. Layout of the considered urban rail transit network (with 4 lines).

TABLE I
PARAMETERS OF THE NETWORK FOR THE CASE STUDY

Parameters	Changping Line	Line 8	Line 13	Line 15
Minimum headway	120 s	120 s	120 s	120 s
Regular headway	180 s	180 s	180 s	180 s
Maximum dwell time	360 s	360 s	360 s	360 s
Minimum dwell time	30 s	30 s	30 s	30 s
Regular dwell time	60 s	60 s	60 s	60 s
Train capacity	2400	2400	2400	2400
Average transfer time	60 s	60 s	60 s	60 s
Phase time T	1800 s	1800 s	1800 s	1800 s
Total available trains	24 trains	28 trains	32 trains	24 trains
Circulation time	5407.5 s	6227.6 s	7101.9 s	3315.9 s

passenger OD demands for the simulation of the deterministic case. For the uncertain case, we generate uncertain passenger OD demands using Poisson distribution [36] based on the passenger flow data with an additional variation within 30% for simulations and the scenario-based approach to simulate the over-crowdedness cases. The number of representative scenarios for each local subsystem is 5, while both the number of global scenarios and scenarios for neighbor subsystems are 1. In this context, each subsystem only needs to consider 5 scenarios in total.

After generating the number of trains departing from each platform during each phase, we generate the detailed departure and arrival times of each train by a lower-level controller (e.g., the controller developed in [29]). We use the passenger absorption model as the prediction model and an elaborate model from the literature (i.e., the model in [9], [37]) as the simulation model to evaluate the effectiveness of the developed approaches. In each MPC step, the resulting mixed-integer linear programming problem is solved by the `gurobi` solver called from MATLAB (R2019b). The simulations are performed on a computer with an Intel Xeon W-2223 CPU and 8GB RAM.

B. Real-Time Train Scheduling for the Deterministic Case

We conduct simulations for the deterministic case to show the effectiveness of the developed knowledgeable-reduced-horizon (KRH) algorithm and the distributed knowledgeable-reduced-horizon (DKRH) algorithm. For comparison, we also perform simulations for the basic timetable as well as the original MPC approach.

According to the circulation time of each line, the prediction horizon of all MPC approaches should be $N \geq 4$ (i.e., the length of the prediction time window satisfies $t \geq 7200$ s) to ensure that the MPC optimization problem can cover every station in the network. The prediction horizon of the original MPC approach is set as $N = 6$, while the prediction horizon for KRH and DKRH is reduced to $N = 4$. For the DKRH approach, we use three subsystems, where each line in Fig. 3 is regarded as one subsystem. Considering the real-time implementation, we set the maximum solution time for each MPC step to 3600 s to meet the real-time feasibility requirement.

TABLE II
SIMULATION RESULTS FOR DIFFERENT APPROACHES UNDER THE DETERMINISTIC CASE

Approach	Objective	Improvement	CPU time (s)	
			t_{avrg}	$t_{\max}$
Basic timetable	$8.4607 \cdot 10^4$	-	-	-
MPC (N=6)	$7.0633 \cdot 10^4$	16.52%	3600.0	3600.0
KRH (N=4)	$7.1614 \cdot 10^4$	15.36%	250.7	636.0
DKRH (N=4)	$7.1174 \cdot 10^4$	15.88%	35.5	37.8

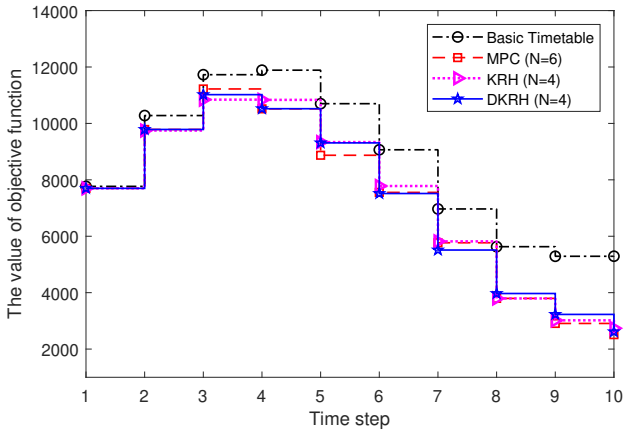

Fig. 4. Value of the objective function at each time step.

The simulation results are displayed in TABLE II, where the value of objective function and computation time of each approach are collected. The value of objective function in each MPC step is shown in Fig. 4. These results show that MPC, KRH, and DKRH can improve the performance of the basic timetable, with an improvement of 16.52%, 15.36%, and 15.88%, respectively. As a real-time control approach, the online computational burden is an essential issue for MPC, which is significantly influenced by the prediction horizon. The original MPC approach with prediction horizon $N = 6$ cannot calculate its optimal solution within 3600 s. By using the cost-to-go function in the developed KRH algorithm, the prediction horizon and the solution space are reduced. Thus, the CPU time of the KRH algorithm is reduced significantly while ensuring an acceptable level of solution quality.

As we divide the original problem into three smaller subproblems in the DKRH algorithm, the computational burden of each subproblem is further reduced. Compared with the KRH algorithm, the average CPU time for the DKRH algorithm is reduced from 250.7 s to 35.5 s, and the maximum CPU time is reduced from 636.0 s to 37.8 s. The solution time of the DKRH algorithm is further reduced while maintaining the same level of control performance.

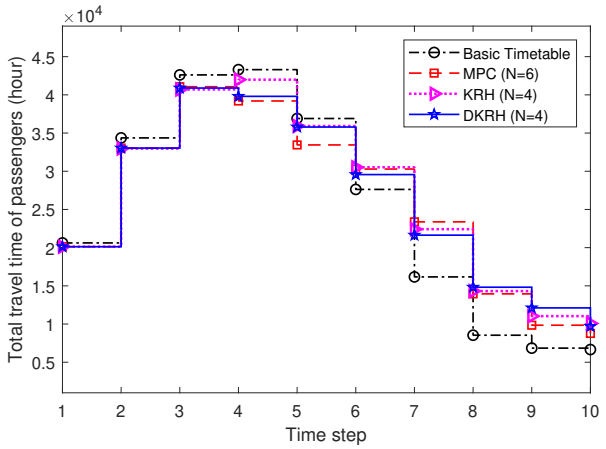

Fig. 5. Total travel time of passengers at each time step.

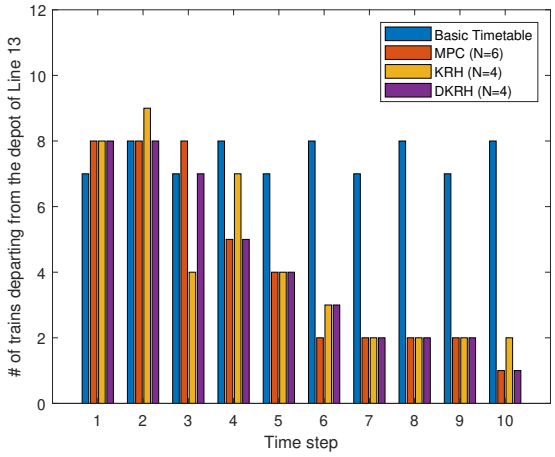

Fig. 6. Number of trains departing from the depot of Line 13.

The total travel time of passengers is shown in Fig. 5. To further illustrate the results, the number of trains departing from the depot of Line 13 is given in Fig. 6. In Fig. 5 and Fig. 6, time steps 1-3 represent the morning peak hours at 7:00-8:30. Compared with the basic timetable, more trains are scheduled to attend the large passenger demand in the morning peak hours. Since the maximum number of available trains for Line 13 is 32, and the circulation time for Line 13 is 7101.9 s, which is approximately equal to the length of 4 phases, the maximum number of trains scheduled for each phase is restricted. Compared with peak hours, fewer trains are scheduled in off-peak hours to reduce operational costs with an acceptable increase in the total passenger travel time. Based on the developed approaches, we can obtain the number of trains departing from each platform during each phase, and the corresponding timetable can be further generated. Examples of the timetable are shown in Appendix A.

The simulation results indicate that both KRH and DKRH can be used for real-time train scheduling for urban rail transit networks. In particular, when there are no communication restrictions between different lines, KRH can be used to get a high-quality solution; otherwise, especially for large-scale networks when centralized control for the whole network is not possible due to the communication restrictions, DKRH can be used to achieve real-time train scheduling for the urban rail transit network.

C. Real-Time Train Scheduling Considering Uncertain Passenger Flows

In general, passenger demands in urban rail transit networks satisfy a Poisson distribution [36]. In this section, we perform simulations when there exists uncertainty in passenger flows to evaluate the effectiveness of the developed scenario-based distributed knowledgeable-reduced-horizon (S-DKRH) algorithm.

We first start simulations for one uncertain scenario. To have a baseline, we also conduct a simulation with perfect knowledge of the uncertainties, which is indicated as P-DKRH below. It is worth noting that P-DKRH is not realistic as it is not possible to have perfect knowledge of the uncertainties in real life. For the DKRH algorithm, we use the expected value of the passenger demands to calculate the timetable. Using the simulation results in Section V-B, the prediction horizon for P-DKRH, DKRH, and S-DKRH is set as $N = 4$.

TABLE III
SIMULATION RESULTS FOR DIFFERENT APPROACHES UNDER THE UNCERTAIN CASE

Approach	Objective	Improvement	CPU time (s)	
			t_{avrg}	$t_{\max}$
Basic timetable	$9.7262 \cdot 10^4$	-	-	-
P-DKRH	$8.4282 \cdot 10^4$	13.35%	35.4	40.9
DKRH	$8.7171 \cdot 10^4$	10.38%	34.4	37.7
S-DKRH	$8.4718 \cdot 10^4$	12.90%	347.6	385.9

TABLE III and Fig. 7 show the simulation results of different approaches under uncertain passenger flows. Compared with the basic timetable, an improved performance can be observed for both DKRH and S-DKRH, with an improvement of 10.38% and 12.90%, respectively. Compared with DKRH, the objective function value of S-DKRH is closer to that of P-DKRH, which implies the effectiveness of the scenario-based approach. Both DKRH and S-DKRH satisfy the real-time feasibility requirement for the given case study. The computational burden of S-DKRH is larger than that of DKRH, and the average CPU time increases from 34.4 s to 347.6 s for S-DKRH. The simulation results demonstrate that a suitable choice is required in real-life applications, i.e., when the CPU power is sufficient, S-DKRH is a better choice to obtain a higher-quality solution; otherwise, when the CPU power is not

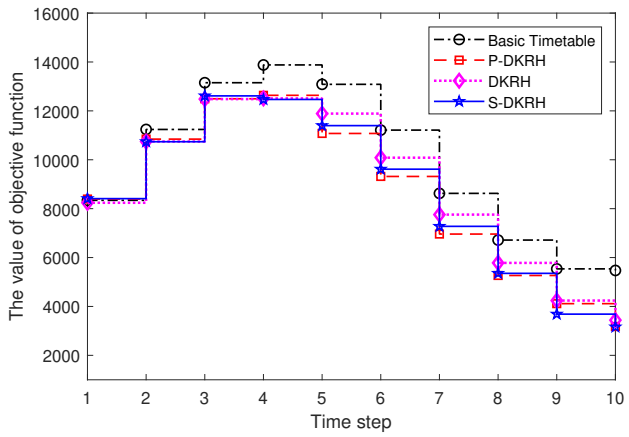

Fig. 7. Value of the objective function at each time step.

sufficient, DKRH can be used to calculate a timetable within a shorter period of time with acceptable performance.

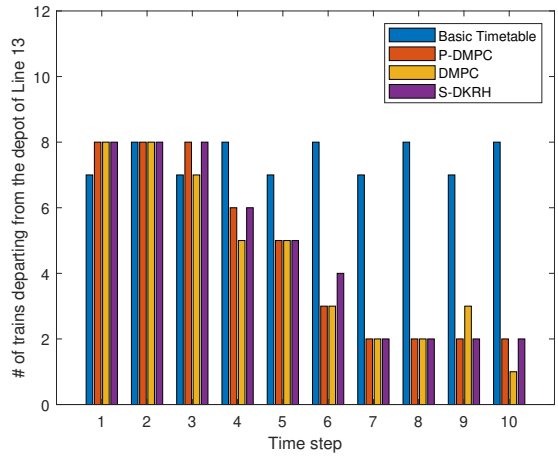

Fig. 8. Number of trains departing from the depot of Line 13 in the uncertain case.

The number of trains departing from the depot of Line 13 in the uncertain case is shown in Fig. 8. Time steps 1-3 are associated with the morning peak hours at 7:00-8:30, and it can be observed that more trains are scheduled at time steps 1-3 to attend the large passenger demands. We can also obtain a timetable based on the above results; the timetables of DKRH and S-DKRH for Line 13 at 7:00-8:00 are given in Appendix B.

TABLE IV
COMPARISON OF THE OBJECTIVE FUNCTION VALUES FOR DIFFERENT APPROACHES

	Average	Standard deviation
Basic timetable	$9.3997 \cdot 10^4$	$6.6490 \cdot 10^3$
P-DKRH	$8.0825 \cdot 10^4$	$6.5604 \cdot 10^3$
DKRH	$8.2794 \cdot 10^4$	$7.7432 \cdot 10^3$
S-DKRH	$8.1147 \cdot 10^4$	$7.2378 \cdot 10^3$

To further demonstrate the effectiveness of the developed S-DKRH algorithm, simulations are carried out in 10 different scenarios. The 10 scenarios are generated based on Poisson distribution with the real-life passenger entering and exiting flow data as the expected value. The average value and the standard deviation of the objective function values for the basic timetable, and the timetable obtained by P-DKRH, DKRH, and S-DKRH are calculated. Compared with the average objective function value of the basic timetable, P-DKRH, DKRH, and S-DKRH yield an improvement of 14.01%, 11.92%, and 13.67%, respectively. Although P-DKRH outperforms DKRH and S-DKRH with respect to both the average value and the standard deviation, as stated before, P-DKRH is not realizable in real life[3]. It can be observed in Table IV that the average objective function value and the standard deviation of S-DKRH are smaller than that of DKRH. The simulation results imply that S-DKRH can be a suitable choice to handle uncertain passenger flows.

VI. CONCLUSIONS

In this paper, we have investigated the real-time train scheduling problem for urban rail transit networks considering uncertain time-dependent passenger OD demands. The passenger absorption model of [13] has been extended to include the rolling stock availability to generate more practically implementable timetables by considering the total number of available trains. To reduce the prediction horizon of the real-time train scheduling problem, a novel cost-to-go function has been developed. By considering different lines as different subsystems, a distributed-knowledgeable-reduced-horizon (DKRH) algorithm has been proposed considering the computational complexity and communication restrictions in practical urban rail transit networks. Furthermore, a scenario-based distributed-knowledgeable-reduced-horizon algorithm (S-DKRH) has been developed to deal with uncertain passenger flows. Numerical experiments have been conducted to illustrate that 1) DKRH can be used for real-time train scheduling of urban rail transit networks and 2) the S-DKRH algorithm yields better performance than DKRH with an acceptable increase in computation time for uncertain cases.

The results in this paper can help the operator to optimize train schedules to handle uncertain time-dependent passenger demands. Future research includes developing efficient solution approaches for the resulting optimization problems to further improve the real-time feasibility of the approach. In particular, integrating learning-based strategies to learn integer variables can be a possible choice to speed up the optimization process. Furthermore, next to optimizing the train departure frequencies, adjusting train composition can also be a choice to handle time-dependent passenger demands.

[3]As we use the absorption model as the prediction model and the model in [9] as the simulation model, there exists a model mismatch issue, which may yield the objective function value of P-DKRH larger than that of DKRH and S-DKRH in some scenarios.

APPENDIX A
TIMETABLES OBTAINED BY DIFFERENT APPROACHES UNDER THE DETERMINISTIC CASE

The basic timetable and the timetables generated by MPC, KRH, and DKRH for Line 13 in 7:00-8:00 are given in Figures 9-12, respectively. The time slot 7:00-8:00 corresponds to time step 1-2 in Fig. 6. It can be observed from Figures 9-12 that all timetables are feasible, i.e., the departure and arrival times of trains in each station satisfy the minimum dwell time and headway constraints. The simulation results show that the developed KRH and DKRH approaches can be used to generate a feasible timetable in real time.

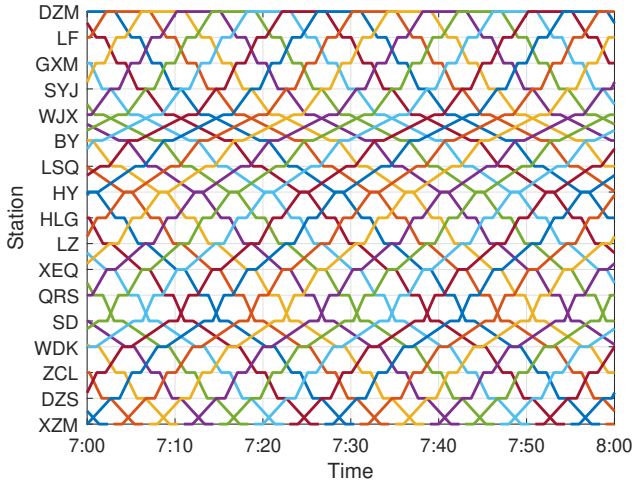

Fig. 9. Basic timetable for Line 13.

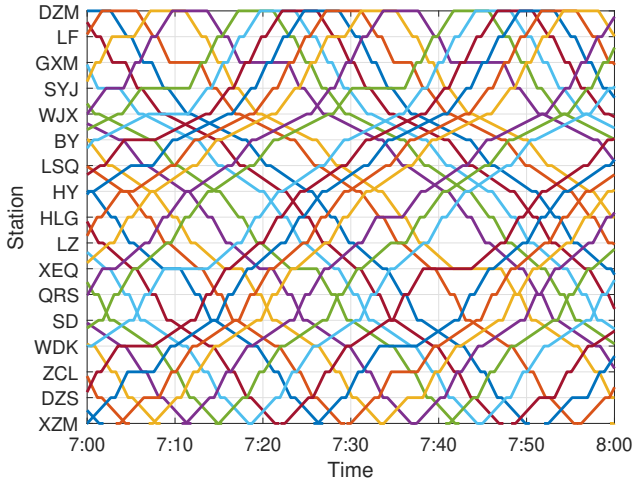

Fig. 10. Timetable for Line 13 generated by MPC (N=6) under the deterministic case.

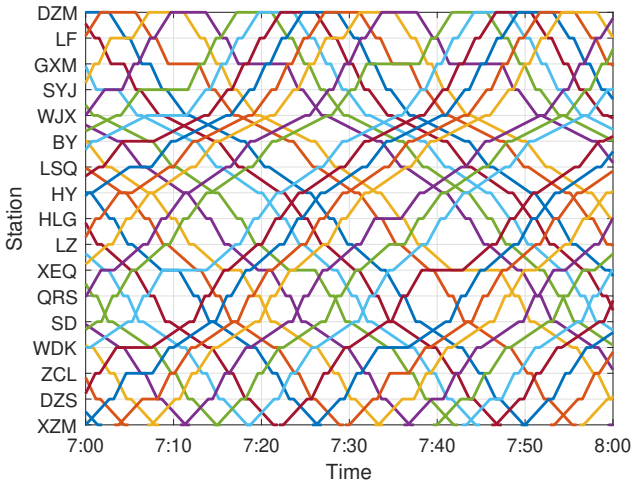

Fig. 11. Timetable for Line 13 generated by KRH (N=4) under the deterministic case.

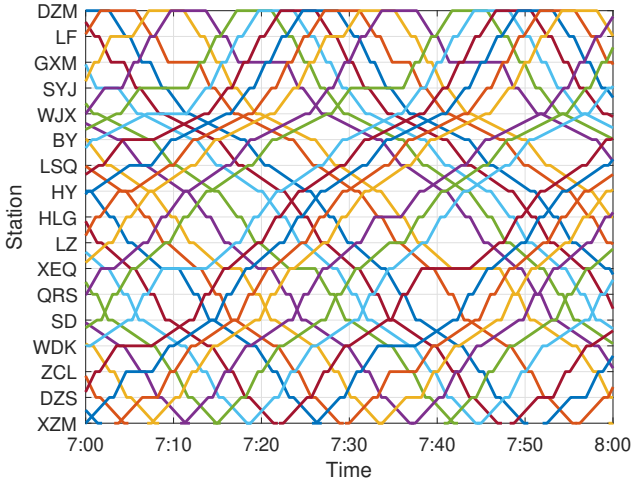

Fig. 12. Timetable for Line 13 generated by DKRH (N=4) under the deterministic case.

APPENDIX B
TIMETABLES OBTAINED BY DIFFERENT APPROACHES UNDER THE UNCERTAIN CASE

Fig. 13 and Fig. 14 presents the timetable of Line 13 generated by DKRH and S-DKRH under the uncertain case in Section V-C. The timetables correspond to time steps 1-2 in Fig. 8.

REFERENCES

[1] Y. Wang, B. Ning, T. Tang, T. J. van den Boom, and B. De Schutter, "Efficient real-time train scheduling for urban rail transit systems using iterative convex programming," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 6, pp. 3337–3352, 2015.

[2] Z. Hou, H. Dong, S. Gao, G. Nicholson, L. Chen, and C. Roberts, "Energy-saving metro train timetable rescheduling model considering ato profiles and dynamic passenger flow," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 7, pp. 2774–2785, 2019.

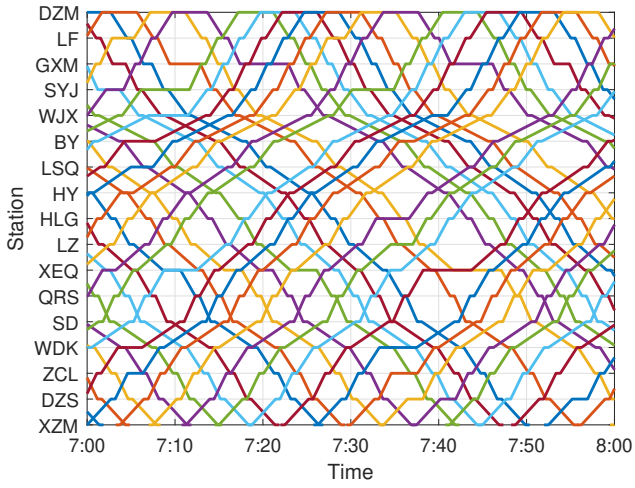

Fig. 13. Timetable for Line 13 generated by DKRH under the uncertain case.

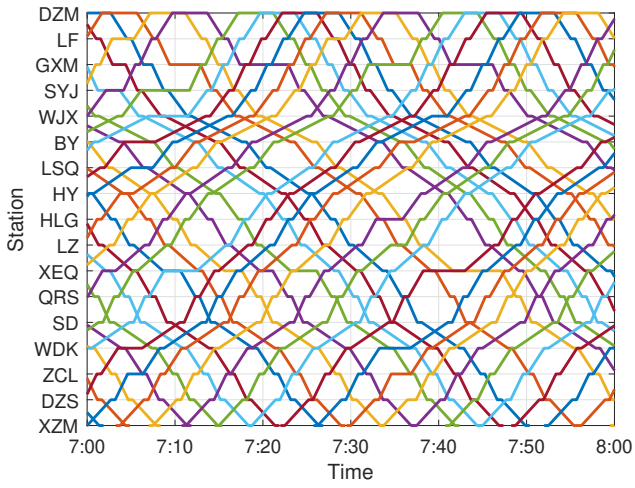

Fig. 14. Timetable for Line 13 generated by S-DKRH the uncertain case.

[3] Y. Wang, A. D'Ariano, J. Yin, L. Meng, T. Tang, and B. Ning, "Passenger demand oriented train scheduling and rolling stock circulation planning for an urban rail transit line," *Transportation Research Part B: Methodological*, vol. 118, pp. 193–227, 2018.

[4] P. Mo, L. Yang, A. D'Ariano, J. Yin, Y. Yao, and Z. Gao, "Energy-efficient train scheduling and rolling stock circulation planning in a metro line: a linear programming approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 9, pp. 3621–3633, 2020.

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

[6] J. Shi, T. Qin, L. Yang, X. Xiao, J. Guo, Y. Shen, and H. Zhou, "Flexible train capacity allocation for an overcrowded metro line: A new passenger flow control approach," *Transportation Research Part C: Emerging Technologies*, vol. 140, p. 103676, 2022.

[7] H. Zhou, J. Qi, L. Yang, J. Shi, and P. Mo, "Joint optimization of train scheduling and rolling stock circulation planning with passenger flow control on tidal overcrowded metro lines," *Transportation Research Part C: Emerging Technologies*, vol. 140, p. 103708, 2022.

[8] D. Canca, E. Barrena, A. De-Los-Santos, and J. L. Andrade-Pineda, "Setting lines frequency and capacity in dense railway rapid transit networks with simultaneous passenger assignment," *Transportation Research Part B: Methodological*, vol. 93, pp. 251–267, 2016.

[9] Y. Wang, T. Tang, B. Ning, T. J. van den Boom, and B. De Schutter, "Passenger-demands-oriented train scheduling for an urban rail transit network," *Transportation Research Part C: Emerging Technologies*, vol. 60, pp. 1–23, 2015.

[10] J. Yin, A. D'Ariano, Y. Wang, L. Yang, and T. Tang, "Timetable coordination in a rail transit network with time-dependent passenger demand," *European Journal of Operational Research*, vol. 295, no. 1, pp. 183–202, 2021.

[11] X. Luan and F. Corman, "Passenger-oriented traffic control for rail networks: An optimization model considering crowding effects on passenger choices and train operations," *Transportation Research Part B: Methodological*, vol. 158, pp. 239–272, 2022.

[12] A. De-Los-Santos, G. Laporte, J. A. Mesa, and F. Perea, "Simultaneous frequency and capacity setting in uncapacitated metro lines in presence of a competing mode," *Transportation Research Procedia*, vol. 3, pp. 289–298, 2014.

[13] X. Liu, A. Dabiri, and B. De Schutter, "Timetable scheduling for passenger-centric urban rail networks: Model predictive control based on a novel absorption model," in *2022 IEEE Conference on Control Technology and Applications (CCTA)*. IEEE, 2022, pp. 1147–1152.

[14] X. Liu, A. Dabiri, Y. Wang, and B. De Schutter, "Modeling and efficient passenger-oriented control for urban rail transit networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 3, pp. 3325–3338, 2023.

[15] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

[16] J. Jeschke, D. Sun, A. Jamshidnejad, and B. De Schutter, "Grammatical-evolution-based parameterized model predictive control for urban traffic networks," *Control Engineering Practice*, vol. 132, p. 105431, 2023.

[17] T. J. van den Boom and B. De Schutter, "Modelling and control of discrete event systems using switching max-plus-linear systems," *Control Engineering Practice*, vol. 14, no. 10, pp. 1199–1211, 2006.

[18] G. Caimi, M. Fuchsberger, M. Laumanns, and M. Lüthi, "A model predictive control approach for discrete-time rescheduling in complex central railway station areas," *Computers & Operations Research*, vol. 39, no. 11, pp. 2578–2593, 2012.

[19] T. Keviczky, F. Borrelli, and G. J. Balas, "Decentralized receding horizon control for large scale dynamically decoupled systems," *Automatica*, vol. 42, no. 12, pp. 2105–2115, 2006.

[20] Y. Kuwata, A. Richards, T. Schouwenaars, and J. P. How, "Distributed robust receding horizon control for multivehicle guidance," *IEEE Transactions on Control Systems Technology*, vol. 15, no. 4, pp. 627–641, 2007.

[21] J. M. Maestre, R. R. Negenborn *et al.*, *Distributed Model Predictive Control Made Easy*. Springer, 2014, vol. 69.

[22] B. Kersbergen, T. van den Boom, and B. De Schutter, "Distributed model predictive control for railway traffic management," *Transportation Research Part C: Emerging Technologies*, vol. 68, pp. 462–489, 2016.

[23] X. Luan, B. De Schutter, L. Meng, and F. Corman, "Decomposition and distributed optimization of real-time traffic management for large-scale railway networks," *Transportation Research Part B: Methodological*, vol. 141, pp. 72–97, 2020.

[24] V. Cacchiani, J. Qi, and L. Yang, "Robust optimization models for integrated train stop planning and timetabling with passenger demand uncertainty," *Transportation Research Part B: Methodological*, vol. 136, pp. 1–29, 2020.

[25] G. C. Calafiore and M. C. Campi, "The scenario approach to robust control design," *IEEE Transactions on Automatic Control*, vol. 51, no. 5, pp. 742–753, 2006.

[26] M. C. Campi, S. Garatti, and F. A. Ramponi, "A general scenario theory for nonconvex optimization and decision making," *IEEE Transactions on Automatic Control*, vol. 63, no. 12, pp. 4067–4078, 2018.

[27] L. Yang, Y. Zhang, S. Li, and Y. Gao, "A two-stage stochastic optimization model for the transfer activity choice in metro networks," *Transportation Research Part B: Methodological*, vol. 83, pp. 271–297, 2016.

[28] C. Gong, J. Shi, Y. Wang, H. Zhou, L. Yang, D. Chen, and H. Pan, "Train timetabling with dynamic and random passenger demand: A stochastic optimization method," *Transportation Research Part C: Emerging Technologies*, vol. 123, p. 102963, 2021.

[29] X. Liu, A. Dabiri, J. Xun, and B. De Schutter, "Bi-level model predictive control for metro networks: Integration of timetables, passenger flows, and train speed profiles," *Transportation Research Part E: Logistics and Transportation Review*, 2023, manuscript accepted for publication.

[30] A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints," *Automatica*, vol. 35, no. 3, pp. 407–427, 1999.
[31] H. P. Williams, *Model Building in Mathematical Programming*. John Wiley & Sons, 2013.
[32] Y. Kuwata and J. P. How, "Cooperative distributed robust trajectory optimization using receding horizon MILP," *IEEE Transactions on Control Systems Technology*, vol. 19, no. 2, pp. 423–431, 2010.
[33] N. Growe-Kuska, H. Heitsch, and W. Romisch, "Scenario reduction and scenario tree construction for power management problems," in *2003 IEEE Bologna Power Tech Conference Proceedings,*, vol. 3. IEEE, 2003, pp. 1 – 7.
[34] J. M. Latorre, S. Cerisola, and A. Ramos, "Clustering algorithms for scenario tree generation: Application to natural hydro inflows," *European Journal of Operational Research*, vol. 181, no. 3, pp. 1339–1353, 2007.
[35] S. Liu, A. Sadowska, and B. De Schutter, "A scenario-based distributed model predictive control approach for freeway networks," *Transportation Research Part C: Emerging Technologies*, vol. 136, p. 103261, 2022.
[36] J. Yin, T. Tang, L. Yang, Z. Gao, and B. Ran, "Energy-efficient metro train rescheduling with uncertain time-variant passenger demands: An approximate dynamic programming approach," *Transportation Research Part B: Methodological*, vol. 91, pp. 178–210, 2016.
[37] N. Bešinović, Y. Wang, S. Zhu, E. Quaglietta, T. Tang, and R. M. Goverde, "A matheuristic for the integrated disruption management of traffic, passengers and stations in urban railway lines," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 8, pp. 10 380–10 394, 2022.