

Joint rolling stock and crew scheduling with multi-train composition in urban rail networks

E. Wang, L. Yang, Y. Adulyasak, J.-F. Cordeau, Z. Gao

G-2025-47

July 2025

La collection *Les Cahiers du GERAD* est constituée des travaux de recherche menés par nos membres. La plupart de ces documents de travail a été soumis à des revues avec comité de révision. Lorsqu'un document est accepté et publié, le pdf original est retiré si c'est nécessaire et un lien vers l'article publié est ajouté.

The series *Les Cahiers du GERAD* consists of working papers carried out by our members. Most of these pre-prints have been submitted to peer-reviewed journals. When accepted and published, if necessary, the original pdf is removed and a link to the published article is added.

Citation suggérée : E. Wang, L. Yang, Y. Adulyasak, J.-F. Cordeau, Z. Gao (juillet 2025). Joint rolling stock and crew scheduling with multi-train composition in urban rail networks, Rapport technique, Les Cahiers du GERAD G- 2025-47, GERAD, HEC Montréal, Canada.

Suggested citation: E. Wang, L. Yang, Y. Adulyasak, J.-F. Cordeau, Z. Gao (July 2025). Joint rolling stock and crew scheduling with multi-train composition in urban rail networks, Technical report, Les Cahiers du GERAD G-2025-47, GERAD, HEC Montréal, Canada.

Avant de citer ce rapport technique, veuillez visiter notre site Web (<https://www.gerad.ca/fr/papers/G-2025-47>) afin de mettre à jour vos données de référence, s'il a été publié dans une revue scientifique.

Before citing this technical report, please visit our website (<https://www.gerad.ca/en/papers/G-2025-47>) to update your reference data, if it has been published in a scientific journal.

La publication de ces rapports de recherche est rendue possible grâce au soutien de HEC Montréal, Polytechnique Montréal, Université McGill, Université du Québec à Montréal, ainsi que du Fonds de recherche du Québec – Nature et technologies.

The publication of these research reports is made possible thanks to the support of HEC Montréal, Polytechnique Montréal, McGill University, Université du Québec à Montréal, as well as the Fonds de recherche du Québec – Nature et technologies.

Dépôt légal – Bibliothèque et Archives nationales du Québec, 2025
– Bibliothèque et Archives Canada, 2025

Legal deposit – Bibliothèque et Archives nationales du Québec, 2025
– Library and Archives Canada, 2025

GERAD HEC Montréal
3000, chemin de la Côte-Sainte-Catherine
Montréal (Québec) Canada H3T 2A7

Tél. : 514 340-6053
Télec. : 514 340-5665
info@gerad.ca
www.gerad.ca

Joint rolling stock and crew scheduling with multi-train composition in urban rail networks

Entai Wang ^{a, b, c}

Lixing Yang ^a

Yossiri Adulyasak ^{b, c}

Jean-François Cordeau ^{b, c}

Ziyou Gao ^a

^a School of Systems Science, Beijing Jiaotong University, Beijing, China, 100044

^b GERAD, Montréal (Qc), Canada, H3T 1J4

^c HEC Montréal, Montréal (Qc), Canada, H3T 2A7

entai.wang@hec.ca

lxyang@bjtu.edu.cn

yossiri.adulyasak@hec.ca

jean-francois.cordeau@hec.ca

zygao@bjtu.edu.cn

July 2025

Les Cahiers du GERAD

G–2025–47

Copyright © 2025 Wang, Yang, Adulyasak, Cordeau, Gao

Les textes publiés dans la série des rapports de recherche *Les Cahiers du GERAD* n'engagent que la responsabilité de leurs auteurs. Les auteurs conservent leur droit d'auteur et leurs droits moraux sur leurs publications et les utilisateurs s'engagent à reconnaître et respecter les exigences légales associées à ces droits. Ainsi, les utilisateurs:

- Peuvent télécharger et imprimer une copie de toute publication du portail public aux fins d'étude ou de recherche privée;
- Ne peuvent pas distribuer le matériel ou l'utiliser pour une activité à but lucratif ou pour un gain commercial;
- Peuvent distribuer gratuitement l'URL identifiant la publication.

Si vous pensez que ce document enfreint le droit d'auteur, contactez-nous en fournissant des détails. Nous supprimerons immédiatement l'accès au travail et enquêterons sur votre demande.

The authors are exclusively responsible for the content of their research papers published in the series *Les Cahiers du GERAD*. Copyright and moral rights for the publications are retained by the authors and the users must commit themselves to recognize and abide the legal requirements associated with these rights. Thus, users:

- May download and print one copy of any publication from the public portal for the purpose of private study or research;
- May not further distribute the material or use it for any profit-making activity or commercial gain;
- May freely distribute the URL identifying the publication.

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Abstract : Rolling stock scheduling and crew scheduling are two fundamental problems that arise in the planning of urban rail operations and that are especially important in the case of flexible operations in real-world networks. These problems are often solved separately and sequentially in different planning stages, resulting in limited options to adjust crew schedules after rolling stock decisions have been made. To better adjust these two decision-making processes and achieve better solutions, this paper studies a joint rolling stock and crew scheduling problem in urban rail networks. A novel optimization model is formulated with the aim of reducing the operational cost of rolling stock units and crew members. In addition, the multi-train composition mode is considered to adequately match different frequency requirements and rolling stock transport capacities. To solve the model, a customized branch-and-price-and-cut solution algorithm is proposed to find the optimal schedule schemes, in which Benders decomposition is used to solve the linear programming relaxation of the path-based reformulation. Two customized column generation methods with label correcting are embedded to solve the master problem and pricing sub-problem for generating paths (columns) corresponding to rolling stock units and crew groups, respectively. Finally, a branch-and-bound procedure with several acceleration techniques is proposed to find integer solutions. To demonstrate the computational performance and the robustness of the proposed approaches, a series of numerical experiments are performed in real-world instances of the Beijing urban rail network under different settings. The computational results confirm the high efficiency of the solution methodology and the benefits of the flexible operation schemes based on the solutions found by the proposed methods.

Keywords: Urban rail network, rolling stock scheduling, crew scheduling, branch-and-price-and-cut, Benders decomposition, column generation

1 Introduction

Urban rail systems have several benefits over other urban transportation modes (e.g., bus, private car) in terms of capacity, punctuality, and comfort. They have been developing rapidly in several megacities (e.g., New York, Montréal, Hong Kong). In 2024, the total length of the Beijing urban rail network has reached 880 kilometers, and has become the backbone of city commuting. In general, the management of an urban rail network system involves a complex operational planning process, illustrated in Fig. 1, which includes line planning, train scheduling, rolling stock circulation, and crew scheduling (Heil et al., 2020).

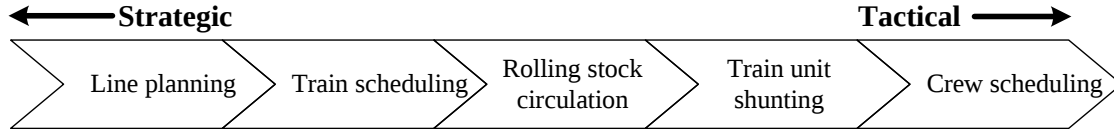


Figure 1: Operational planning process in urban rail network systems

Among the different components of the operational process, the *train scheduling* problem, which specifies the train departure and arrival times at stations based on the *line planning* scheme, plays a core role in urban rail networks. Then, with the train schedule as an input, the *rolling stock circulation* and *train unit shunting* plans assign rolling stock units to trips by minimizing the total operational costs. Generally, the *rolling stock scheduling* problem aims to simultaneously specify the train scheduling process with the rolling stock circulation and train unit shunting processes. To adequately match the different frequency requirements at different physical segments and time windows, the multiple-train composition mode is often used in the process of train unit shunting. Under this mode, a long composition with larger passenger capacity can be constructed by coupling multiple rolling stock units (Zhou et al., 2022; Pan et al., 2023; Wang et al., 2024), which is beneficial to satisfy different frequencies by covering trips based on passenger demand at different physical segments and peak/off-peak hours. We note that the rolling stock scheduling process with the multi-train composition mode in urban rail networks increases the operational complexity since different operational components need to be considered simultaneously (e.g., train scheduling, rolling stock circulation, and train coupling/decoupling). After designing rolling stock circulation plans with appropriate train unit shunting schemes, the next step is to schedule crew members. That is, each crew member is assigned a sequence of appropriate trips within the defined time window.

According to Heil et al. (2020) and Pärper et al. (2025), the above decision processes are generally performed successively owing to their complex interdependencies. For example, the dispatchers need to properly assign trips to individual crew members under the information of the given rolling stock scheduling plan so that all trips can be covered by specific crew members with the satisfaction of labor requirements. Thus, dispatchers have to give feedback repeatedly to re-plan the train schedule and rolling stock circulation to eventually make these operation plans consistent.

Focusing on providing a system-optimal solution, this paper aims to investigate a joint rolling stock and crew scheduling problem (JRCSP) with the multi-train composition in urban rail networks to generate a consistent and optimal operation plan with respect to different operational components (e.g., trips, trip connections, coupling/decoupling activities, pull-in/out operations, etc.). Then, the goal of the JRCSP is to find a series of paths for rolling stock units and crew members, satisfying all operational and labor requirements while minimizing the total cost of train routes, trips, trip connections, and crew members.

The remainder of this paper is organized as follows. Section 2 introduces a review of the related literature and summarizes the main contributions of this study. Section 3 provides a detailed problem statement with some concepts and notation. In Section 4, a path-based model is introduced for the

JRCSP. In Section 5, a customized BPC solution algorithm is designed to solve the proposed model, and computational results are represented in Section 6. We finally conclude the paper in Section 7.

2 Literature review and paper contribution

Rolling stock and crew scheduling are two critical components in developing an urban rail operation scheme, which have attracted tremendous attention from both academia and industry in recent decades (see Lusby et al. (2011); Heil et al. (2020); Pärper et al. (2025); Correia Duarte et al. (2025) for detailed reviews). In the following discussion, we focus on reviewing the state of the art in the field from three angles: rolling stock scheduling, crew scheduling, and joint optimization of these two problems.

2.1 Rolling stock scheduling

The rolling stock scheduling problem aims to specify the train departure and arrival times at stations together with the circulation of rolling stock units in an urban rail system. In some early literature, to reduce the computational complexity, the train schedule is predetermined and given as an input to assign trips and their connections to complete rolling stock circulations (Cacchiani et al., 2010b, 2013; Lin and Kwan, 2016), with the aim of minimizing operational costs of the rolling stock units. With the rapid development of urban rail systems, the topic of rolling stock scheduling has evolved in three directions to further improve efficiency: passenger-oriented operations, multi-train composition, and network-based operations.

First, compared to the traditional rolling stock scheduling problem, the passenger-demand-oriented operation focuses on the operation schemes based on the effect of passenger demand distribution (Löbel, 1998; Cordeau et al., 2001b; Abbink et al., 2004). In this context, the train frequency is dependent on the time-varying passenger volume at segments and stations (Kroon et al., 2014, 2015; Canca et al., 2018; Amberg et al., 2019), and both the train operational cost and the passenger service level are formulated as objective functions (Niu et al., 2015; Kidd et al., 2019). Second, in recent years, some flexible operation schemes have been implemented for the scheduling of rolling stock units. For example, rolling stock units can be allocated to each trip efficiently by using the multi-train composition mode (Pan et al., 2023; Wang et al., 2024) and the short-turning strategy (Schettini et al., 2022; Yuan et al., 2022), to better match passenger demand with non-equilibrium spatial and temporal distributions. Regarding the solution algorithm, the column generation framework is usually used to solve the rolling stock scheduling problems by specifying space-time paths for rolling stock units (Cacchiani et al., 2008, 2010a; Correia Duarte et al., 2025).

Third, by implementing network-based operations in urban rail networks, rolling stock units can travel flexibly by using connection tracks among different physical urban rail lines. In the literature, the network-based operation is mainly studied from two aspects: train timetable coordination to improve passenger service levels, and rolling stock *cross-line* utilization to improve efficiency. In the first case, the passenger service level can be largely improved with fewer alighting and waiting activities at transfer stations after coordinating train schedules among different physical lines (Nguyen et al., 2001; Wong et al., 2008; Yin et al., 2021). In the second case, there are a few publications considering rolling stock operations in urban rail networks with multiple connected lines and depots, such as the integrated timetabling and vehicle circulation scheduling with periodic and cyclic scenarios (Van Lieshout et al., 2021), and macroscopic rolling stock assignment with appropriate depots (Van Lieshout, 2021). In contrast, our study considers a more holistic view where the crew scheduling and the multi-train composition are jointly planned with rolling stock decisions in urban rail networks.

2.2 Crew scheduling

The existing studies on the crew scheduling problem mainly focuses on assigning trips and meal/rest tasks to crew members based on the rolling stock scheduling plan. Heil et al. (2020) give a detailed

overview in the area of railway crew scheduling with models, methods, and applications. The crew scheduling process often comprises two planning stages: short-term (e.g., one day or shorter) and long-term (e.g., one week or longer). In the short-term problem, each crew member is assigned to several appropriate trips for a relatively short period (e.g., one day). From the perspective of urban rail operators, some practical operation rules are usually incorporated into the crew scheduling problem, such as trip connections (Kroon and Fischetti, 2001), deadheading trips (Abbink et al., 2005), trip numbers (Park and Ryu, 2006), disturbances/disruptions (Breugem et al., 2022b), uncertain passenger demand (Rählmann et al., 2021), planner preferences (Gattermann-Itschert et al., 2023), etc. Then, some legal regulations are also restricted to ensure sustainable working hours for crew members, such as the average working times (Kroon and Fischetti, 2001), time windows/durations of meal/rest breaks (Han and Li, 2014), paid time per duty (Hoffmann et al., 2017), etc. Next, in the long-term crew (roster) scheduling process, short-term crew schedules are combined weekly or monthly with the premise of the long-term planned cycle time, work time, and rest time requirements for crew members (Cordeau et al., 2001b; Caprara et al., 2007). Compared to the short-term crew scheduling problem, more individual crew requirements are considered inevitably to achieve a sustainable mix of work and vacation days, such as work/vacation time accounts (Huisman et al., 2005), fair work distributions among crew members (Breugem et al., 2022a), etc.

Lastly, three major model formulations, including set covering (Kroon et al., 2009), set packing (Borndörfer et al., 2017), and network flow models (Vaidyanathan, 2015), are usually developed for the crew scheduling problems in the literature (i.e., >90% according to Heil et al. (2020)) with the objective functions of the schedule efficiency (Heil et al., 2020), robustness (Lusby et al., 2018) and employee satisfaction (Jütte et al., 2017; Breugem et al., 2022a). In addition, column generation methods can be incorporated into the branch-and-price (B&P) modeling frameworks, then pricing sub-problems are tackled by dynamic programming (Abbink et al., 2011), label setting (Desaulniers and Hickman, 2007), constraint programming (Han and Li, 2014), or genetic algorithms (Hoffmann et al., 2017), etc.

2.3 Joint rolling stock and crew scheduling

In general, the rolling stock schedule is mainly composed of space-time paths of rolling stock units, while the crew schedule has more flexible and complex paths with trip connections and meal/rest requirements for crew members (Heil et al., 2020). In the operation, due to the limitation of rolling stock unit and crew member resources, some studies begin to explore the benefits of joint scheduling optimization in conventional rail systems (Tatsuhiro et al., 2009; Dauzère-Pérès et al., 2015; Bach et al., 2016; Pan et al., 2021). At the same time, some joint optimization problems with similar modeling framework structures have also been studied in other transportation modes, such as the integrated simultaneous aircraft routing and crew scheduling problem in airlines (Cordeau et al., 2001b; Sandhu and Klabjan, 2007; Ruther et al., 2017), berth allocation and pilotage planning in seaport vessel services (Wu et al., 2022), vehicle routing and driver scheduling in road public transportation (Goel and Irnich, 2017; Andrade-Michel et al., 2021), etc.

To highlight the contributions and differences of the present paper, Table 1 presents a detailed comparison of notable studies related to joint rolling stock and crew scheduling or network-based rolling stock operations, from the perspectives of problem features, model, solution method, and largest instance solved. Our main contributions are the following:

- A unified path-based modeling framework is proposed to solve the JRCSP, which is amenable to solution by Benders decomposition (BD) and column generation (CG). Compared with the joint optimization of the rolling stock circulation and crew scheduling problem in urban rail lines (Tatsuhiro et al., 2009; Dauzère-Pérès et al., 2015; Bach et al., 2016; Pan et al., 2021), this study extends more practical problem considerations of rolling stock schedule (i.e., joint train schedule and rolling stock circulation), multi-train composition, and cross-line operations in urban rail networks.

Table 1: Characteristics comparison of some closely related studies

Publication	Problem consideration				Model	Solution method	Largest instance scale	
	Train schedule	Rolling stock circulation	Crew schedule	Network based operation			# of units	# of trips
Tatsuhiro et al. (2009)	No	Re-	Re-	No	NFM+SPM/SCM	LS	185	786
Potthoff et al. (2010)	No	No	Re-	CL	SCM+SPM	CG+LH	59	N/A
Veelenturf et al. (2012)	Re-	No	Re-	No	GILP	INE	46	835
Dauzère-Pérès et al. (2015)	No	Yes	Yes	No	GILP	LR	29	416
Bach et al. (2016)	Yes	Yes	Yes	No	SPM/SCM	BP	30	228
Pan et al. (2021)	No	Yes	Yes	No	NFM+SPM	CG	25	400
Van Lieshout (2021)	Yes	Yes	No	CL	GILP	CT+VI	140	N/A
Yin et al. (2021)	Yes	No	No	TC	GILP	ALNS	N/A	58
This paper	Yes	Yes	Yes	CL	NFM+SPM/SCM	BPC	43	685

(1) Problem consideration: Train re-scheduling (Re-); Re-planning rolling stock circulation (Re-); Crew re-scheduling (Re-); Train schedule coordination (TC); Cross-line rolling stock/crew operation (CL).

(2) Model: Network-flow model (NFM); Set packing model (SPM); Set covering model (SCM); General integer linear programming model (GILP).

(3) Solution method: Local search (LS); Lagrangian heuristic (LH); Iterative neighborhood exploration (INE); Lagrangian relaxation (LR); Branch-and-price (BP); Column generation (CG); Contraction technique (CT); Valid inequality (VI); Adaptive Large Neighborhood Search (ALNS); Branch-and-price-and-cut (BPC).

(4) Largest instance scale: Number of rolling stock units (# of units); Number of trips (# of trips); Not available (N/A), i.e., the specific numbers are not directly given in the study.

- An exact branch-and-price-and-cut (BPC) solution algorithm is developed to solve the JRCSP. By applying Benders decomposition, the relaxed path-based model is first decomposed into a master problem and pricing sub-problems, which are solved by a column generation procedure. To generate crew variables through this process, the customized multi-stage label correcting (LC) algorithm is designed to incorporate crew labor requirements of meal/rest tasks.
- The overall basic BPC approach is enhanced by several acceleration techniques, which are able to obtain (near) optimal solutions with valid lower bounds in a shorter computing time compared with the general-purpose solvers and the standard Benders decomposition (Veelenturf et al., 2012; Dauzère-Pérès et al., 2015). Specifically, we propose lower-bound-lifting (LBL) valid inequalities and priority heuristic rules to better adapt rolling stock and crew operations, and adapt five other techniques from vehicle routing literature.
- A series of computational experiments demonstrate that the proposed approach can obtain high quality solutions to real-world instances with multiple physical lines throughout the day. Our exact algorithm could efficiently handle comparable or even larger instance sizes than those that were considered in other relevant JRCSP studies that employ exact algorithms (i.e., more than 20–30 rolling stock units and 200–500 trips in Dauzère-Pérès et al. (2015); Bach et al. (2016); Pan et al. (2021)). We analyze and present the value of the JRCSP with multi-train composition and cross-line operations are further demonstrated as opposed to the traditional sequential scheduling, single train composition, and single rail-line operations.

3 Problem statement

This paper considers an urban rail network with multiple bidirectional physical lines and multiple depots, which pre-store a number of rolling stock units, as shown in Fig. 2(a). Specifically, we denote by $\mathcal{K}$, indexed by k , the set of rolling stock units. With the multi-train composition mode in the JRCSP, we define two types of stations in the urban rail network: *common stations*, which only process trip dwelling operations but cannot couple or decouple rolling stock units, and *operating stations* (including terminal stations) which permit coupling or decoupling activities in addition to general trip dwelling operations. Then, a *physical segment* is defined as the connection between two adjacent operating stations. In practice, the train compositions and traveling times are usually kept constant on segments

for rolling stock units, especially in urban rail network systems. Thus, we denote the sets of operating stations and physical segments as $\mathcal{I}$ and $\mathcal{A}$, respectively, by omitting common stations along segments. Lastly, the indices i and a are used to denote operating stations and physical segments in the rail network (i.e., $i \in \mathcal{I}, a \in \mathcal{A}$), respectively.

To properly characterize the movements of rolling stock units among different physical segments in the rail network and avoid some obviously inappropriate movements in the solution, a *train route* is defined to represent the movement from origin depot to destination depot for specific rolling stock units, as shown in Fig. 2(b). Specifically, for each rolling stock unit $k \in \mathcal{K}$, a set of train routes $\mathcal{L}_k \subseteq \mathcal{L}$, indexed by l , is pre-determined, and only one train route l can be chosen from set $\mathcal{L}_k$ for the actual operation. In addition, the maintenance cost of assigning a rolling stock unit k to train route l is denoted by $c_{k,l}^1$. Then, one train route $l \in \mathcal{L}$ consists of a sequence of trips, indexed by r , which can be defined as the movement of a rolling stock unit from one operating station to another on a physical segment. We denote by $\mathcal{R}_l^T$ the set of trips associated with train route l , and $\mathcal{R}^T = \bigcup_{l \in \mathcal{L}} \mathcal{R}_l^T$ the set of trips associated with all rolling stock units. Next, we note traveling time T_r^R and the departure time window $\mathcal{T}_r = \mathbb{Z}_{[\bar{T}_r^R, \underline{T}_r^R]}$ of trip r , where $\bar{T}_r^R$ and $\underline{T}_r^R$ denote the earliest and the latest feasible departure times of trip $r \in \mathcal{R}^T$. Lastly, the cost of a trip r starting at timestamp t (denoted by $c_{r,t}^2$) refers to the fixed energy consumption cost on the corresponding segment.

We note that trip set $\mathcal{R}_l^T$ is related to the train route l from set $\mathcal{L}$, and the train route set $\mathcal{L}_k$ is related to rolling stock unit k from set $\mathcal{K}$. As a consequence, the modification in one set will also change the other related sets. Thus, compared with the well-known definition of “trip” in the rolling stock scheduling field, one trip index cannot be shared by different rolling stock units. However, multiple trips can overlap in the visualization (but with different trip indices) to represent the multi-train composition.

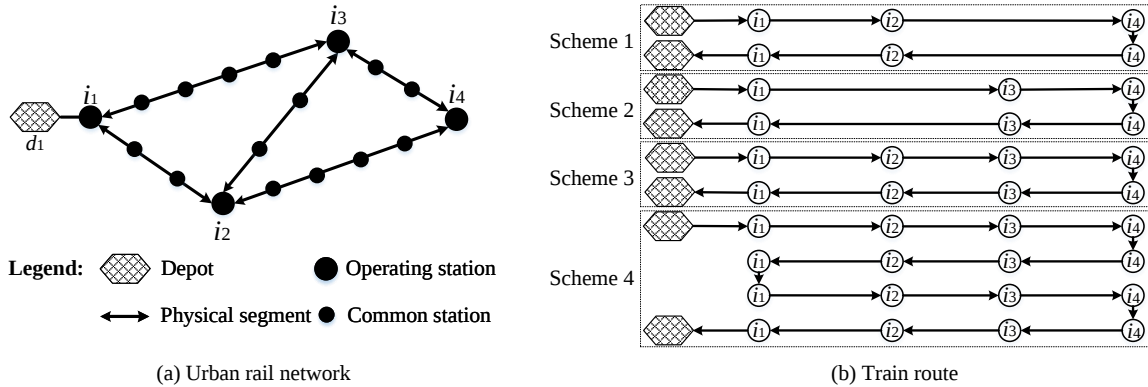


Figure 2: Urban rail network with associated train routes

For clarity, an illustration of the urban rail network is given in Fig. 2(a), which includes one depot d_1 , four operating stations i_1 – i_4 , and five physical segments. Then, based on the urban rail network in Fig. 2(a), four train routes are involved to represent the candidate passing operating station sequences for rolling stock units, as shown in Fig. 2(b). Particularly, we identify train routes 3 and 4 as different schemes with different movements. That is, train route 3 refers to one rolling stock unit starting from depot d_1 , then passing operating stations $i_1 \rightarrow i_2 \rightarrow i_3 \rightarrow i_4 \rightarrow i_3 \rightarrow i_2 \rightarrow i_1$, and returning to depot d_1 with one complete loop. Train route 4 refers to one rolling stock unit starting from depot d_1 , then passing operating stations $i_1 \rightarrow i_2 \rightarrow i_3 \rightarrow i_4 \rightarrow i_3 \rightarrow i_2 \rightarrow i_1 \rightarrow i_2 \rightarrow i_3 \rightarrow i_4 \rightarrow i_3 \rightarrow i_2 \rightarrow i_1$, and returning to depot d_1 with two complete loops.

Furthermore, a certain number of individual crew members (denoted by index c and set $\mathcal{C}$) are available to start work at any operating station. Note that we consider the drivers as crew members in the urban rail system. These crew members are divided into a set $\mathcal{G}$ of crew groups, indexed by g .

Let $\mathcal{C}_g \subseteq \mathcal{C}$ be the set of crew members in crew group g . All crew members who belong to one crew group g should perform their work within the same predetermined time window. In addition, each crew member should be assigned with a series of tasks based on a certain set of requirements, such as meal/rest activities, appropriate connection times, etc.

This study focuses on investigating a JRCSP with multi-train compositions in an urban rail network. The input data consists of the physical structure of the involved rail network, the number of available rolling stock units, and the number of crew groups, etc. The aim is to determine train route selections for rolling stock units, departure times at the first stations for trips, and task sequences for in-service crew members while minimizing both the operational cost for rolling stock units and in-service crew members. Then, the departure/arrival times at common stations along the trips can be calculated based on the departure time at the first stations of trips. Next, we introduce key concepts and notation to formulate the problem in the following.

3.1 Rolling stock unit and crew member operational activities

To capture the spatial and temporal operational activities of rolling stock units and crew members in a rail system, the considered time horizon from time t_0 to time $|\mathcal{T}|$ is discretized into a set of timestamps $\mathcal{T} = \mathbb{Z}_{[t_0, |\mathcal{T}|]}$ with a predefined time granularity. Then, we use origin depot vertex $(d_{ori}(k), t_0)$ and destination depot vertex $(d_{des}(k), t_{|\mathcal{T}|})$ to indicate the source and sink for rolling stock unit k . For modeling convenience, we next introduce the following three types of rolling stock operational activities.

Pull-in/out operation: The pull-in/pull-out operation represents the activity of entering/leaving depots for rolling stock units, respectively. Specifically, for rolling stock unit k , one pull-out operation can be defined with the activity of leaving depot $d_{ori}(k)$ and arriving at its connected operating station i . The pull-in operation can be defined similarly.

Connection operation: Considering that the adjacent previous trip (denoted by $\sigma(r)$) is determined uniquely for trip r based on the associated train route, the connection operations occur between a pair of consecutive trips (i.e., $\sigma(r), r$) at the same operating station i with respect to the same train routes and rolling stock units. Then, the unit-cost of a connection operation between trips $\sigma(r)$ and r (denoted by $c_{k, \sigma(r), r}^3$) refers to the fixed operational cost per min by rolling stock unit k at the operating station.

Rolling stock path: A rolling stock path comprises a sequence of pull-out operations, trips, connection operations, and pull-in operations from the origin depot vertex $(d_{ori}(k), t_0)$ to the destination depot vertex $(d_{des}(k), t_{|\mathcal{T}|})$ for rolling stock unit k during the considered planning horizon. For clarity, we let $\mathcal{P}_{k, l}^R$, indexed by p , denote the set of feasible paths for rolling stock k and train route l . Then, let $\mathcal{P}_k^R = \bigcup_{l \in \mathcal{L}_k} \mathcal{P}_{k, l}^R$ be the set of feasible paths for rolling stock k , and let $\mathcal{P}^R = \bigcup_{k \in \mathcal{K}} \mathcal{P}_k^R$ be the set of feasible paths for all rolling stock units. Then, based on the rolling stock path representation, all departure and arrival times at stations (including operating and common stations) can be specified accurately for each rolling stock unit. For each path p , let $k(p)$ and $l(p)$ denote the associated rolling stock unit and the train route, respectively. Lastly, we denote the cost of path p by C_p^R with the summation of the train route cost, trip traveling cost and connection cost, i.e., $C_p^R = c_{k(p), l(p)}^1 + \sum_{r \in \mathcal{R}_{l(p)}^T} \sum_{t \in \mathcal{T}_r} c_{r, t}^2 \cdot \alpha_{p, r, t} + \sum_{r \in \mathcal{R}_{l(p)}^T} c_{\sigma(r), r}^3 \left(\sum_{t \in \mathcal{T}_r} t \cdot \alpha_{p, r, t} - \sum_{t \in \mathcal{T}_{\sigma(r)}} t \cdot \alpha_{p, \sigma(r), t} \right)$. In this expression, we define two binary parameters $\alpha_{p, r, t}, \forall p \in \mathcal{P}^R, r \in \mathcal{R}^T, t \in \mathcal{T}$ and $\beta_{p, a, t}, \forall p \in \mathcal{P}^R, a \in \mathcal{A}, t \in \mathcal{T}$, where $\alpha_{p, r, t}$ is equal to 1 if and only if trip r is included in path p and starts at timestamp t , and $\beta_{p, a, t}$ is equal to 1 if and only if path p passes through physical segment a from timestamp t .

Furthermore, the crew task should also be included in the operation to characterize the operational activities of crew members. Let $\mathcal{T}_g$ be the set of available timestamps for crew members in crew group g ($\bigcup_{g \in \mathcal{G}} \mathcal{T}_g = \mathcal{T}$), and let $\mathcal{G}_t$ be the set of available crew groups at timestamp t . The operational cost c_g^4 for assigning one crew member in crew group g is constant during the planning horizon with no

relation to the specific number of assigned trips. Note that this is a common payroll regulation with real-world urban rail company applications (e.g., daily wage for crew members based on the labour law). Next, we formally introduce two activities with respect to the meal/rest task and crew task sequence for crew members.

Meal/rest task: The meal tasks and rest tasks, indexed by r , which start within predetermined time windows and last for a given time duration, should be assigned to any crew member c who is scheduled to work in service. Let $\mathcal{R}_c^M$ and $\mathcal{R}_c^R$ be the sets of the required meal/rest tasks for crew member c , and further define $\mathcal{R}^M = \bigcup_{c \in \mathcal{C}} \mathcal{R}_c^M$ and $\mathcal{R}^R = \bigcup_{c \in \mathcal{C}} \mathcal{R}_c^R$. For each meal task $r \in \mathcal{R}^M$ or rest task $r \in \mathcal{R}^R$, we define $\mathcal{T}_r = \mathbb{Z}_{[\underline{T}_r^C, \overline{T}_r^C]}$ as the available meal/rest task time window, and $\underline{T}_r^C / \overline{T}_r^C$ as the feasible earliest/latest starting time of meal/rest task r .

Crew task sequence: A crew task sequence performed by a crew member can also be represented by a path that records assigned trips, meal tasks, and rest tasks in the total planning horizon. Then, considering that the available crew task sequences are exactly the same for crew members in the same group with the same working time windows, we let $\mathcal{P}_g^C$, indexed by p , denote the set of feasible task sequences for crew group g . Then, let $\mathcal{P}^C = \bigcup_{g \in \mathcal{G}} \mathcal{P}_g^C$ be the set of feasible task sequences for all crew groups. Next, let $g(p)$ denote the associated crew group for task sequence p . Lastly, the cost of a crew task sequence $p \in \mathcal{P}^C$ is denoted by C_p^C , which equals C_g^A .

For clarity, an illustration of the rolling stock schedule and Gantt diagram with respect to the instance in Fig. 2 is given in Fig. 3, which includes five physical lines, four operating stations, and one depot. Specifically, station i_1 is connected with depot d_1 , and operating stations i_2, i_3, i_4 have turn-around and connection tracks for coupling/decoupling activities, respectively. In this case, we discretize a 40 min planning horizon into 21 timestamps based on a 2 min time granularity. Three types of train routes (i.e., schemes 1–3 in Fig. 2(b)) are considered in this case with different train physical routes and stop patterns. We use space-time paths to represent the involved operations for four rolling stock units, where all rolling stock units k_1 – k_4 are dispatched from depot d_1 . As shown in Fig. 3(a), rolling stock units k_1 and k_2 are assigned to train routes 1 and 2 of Fig. 2(b), respectively, with different physical stations and segments to cover. In addition, rolling stock units k_3 and k_4 are assigned to the same train route 3 with a coupled large composition from depot d_1 . Then, this long train composition is decoupled into two separate rolling stock units k_3 and k_4 , which execute trips r_7 and r_8 , and then go back to depot d_1 respectively. Lastly, four crew members are scheduled to cover these eight trips while time requirements for the meal/rest tasks are respected, as shown in Fig. 3(b). Compared with the rolling stock units, when it is time to take the meal/rest, the crew members (i.e., drivers) would get off the current rolling stock unit, then take the meal/rest, and get on another rolling stock unit after the meal/rest at the same operating station. For example, crew member 1 first takes a meal task at timestamp 2, then is assigned to trip r_3 associated with rolling stock unit k_3 . After taking the rest from time 19 to 24, crew member 1 is assigned to trip r_6 associated with rolling stock unit k_2 .

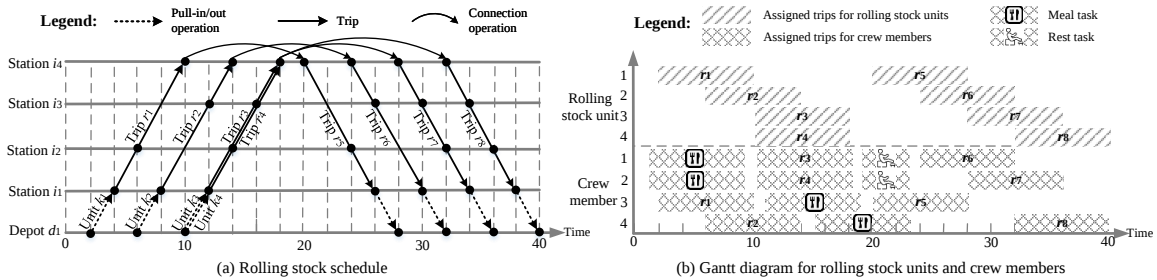


Figure 3: Rolling stock schedule and Gantt diagram for the JRCSP instance

3.2 Assumptions

Without loss of generality, we make five assumptions associated with the characteristics of the rolling stock and crew scheduling operations.

Assumption 1. To make the problem tractable for the proposed BPC algorithm, in the data preparation process, the dynamic passenger demand volume is treated and transformed into the number of required trips (i.e., different frequency requirements) at the corresponding physical segments and time windows. This allows a significant reduction in the number of decision variables related to passenger activities.

Assumption 2. The number of rolling stock units at depots is given as input data and rolling stock units are required to start from and return to the same depot. This maintains the consistency of rolling stock operation schemes over successive days.

Assumption 3. The coupling and decoupling operations are only permitted at operating stations, i.e., transfer stations that connect with multiple physical lines and terminal stations that connect with depots. In addition, the all-stop mode is taken into consideration in the involved problem (i.e., all trips should dwell at all stations). In reality, this is a common operation strategy for real-world applications, e.g., Hong Kong, Montréal, and Beijing urban rail systems.

Assumption 4. As a medium-term plan at the tactical level, the crew scheduling process focuses on determining the number of required crew members to finish the trip tasks with the minimum cost before employing and allocating the specific crew members. Thus, the number of crew members in each group is not limited. However, the objective is to cover all selected trips by assigning the minimum number of crew members as indicated in the objective function.

Assumption 5. The specific locations for starting the first and ending the last trips are flexible at any operating station for all crew members. Then, in the modeling process, all crew members are allowed to depart/end at a virtual source/sink that connects to all operating stations without deadheading trips and pull-in/out operations.

4 Mathematical formulation

This section presents a path-based mathematical optimization model M1 for the JRSCP. We first introduce the notation in Section 4.1. This is followed by the objective function and system constraints in Section 4.2. Considering that the path-based model cannot be implemented with the general-purpose solvers (e.g., CPLEX) directly due to the impractical enumeration of the full set of rolling stock paths and crew task sequences, the equivalent primal model of the JRCSP is also formulated in the electronic companion EC. A for reference.

4.1 Notations

For convenience, the notation used in this study (including the problem statement and the mathematical formulation) is summarized in Table 2. The following path-based binary decision variables are used to model the problem: rolling stock path selection variable λ_p taking value 1 if rolling stock path p is selected (otherwise $\lambda_p = 0$), and crew task sequence selection variable μ_p taking value 1 if crew task sequence p is selected (otherwise $\mu_p = 0$).

Table 2: Notations and parameters in the study

Notation	Definition
Indices	
k	Index of rolling stock units
i	Index of operating stations
a	Index of physical segments
l	Index of train routes
r, r', r''	Index of trips
$\sigma(r), \sigma^{-1}(r)$	Index of the adjacent previous/next trip of trip r

Table 2: Notations and parameters in the study

Notation	Definition
t	Index of timestamps
g	Index of crew groups
p	Index of rolling stock paths/crew task sequences
s	Index of trip frequency requirement scenarios
Sets	
$\mathcal{K}$	Set of rolling stock units
$\mathcal{I}$	Set of operating stations
$\mathcal{A}$	Set of physical segments
$\mathcal{L}_k$	Set of train routes for rolling stock unit k
$\mathcal{L}$	Set of train routes
$\mathcal{R}_l^T$	Set of trips for train route l
$\mathcal{R}^T$	Set of trips
$\mathcal{T}_r$	Set of departure times at the first stations for trip r
$\mathcal{G}$	Set of crew groups
$\mathcal{C}$	Set of crew members
$\mathcal{C}_g$	Set of crew members belonging to crew group g
$\mathcal{T}$	Set of timestamps
$\mathcal{P}_{k,l}^R, \mathcal{P}_k^R, \mathcal{P}^R$	Set of feasible paths for rolling stock unit k and train route l / rolling stock unit k / all rolling stock units
$\mathcal{T}^g$	Set of available timestamps for crew members in group g
$\mathcal{G}^t$	Set of available crew groups at timestamp t
$\mathcal{R}_c^M, \mathcal{R}_c^R$	Set of meal/rest tasks for crew member c
$\mathcal{R}_g^M, \mathcal{R}_g^R$	Set of meal/rest tasks for crew members in crew group g
$\mathcal{P}_g^C, \mathcal{P}^C$	Set of feasible task sequences for crew group g / all crew groups
$\mathcal{V}$	Set of possible conflicting trip pairs
$\mathcal{S}_a$	Set of trip frequency requirement scenarios for physical segment a
$\mathcal{T}_s$	Set of involved timestamps for trip frequency requirement scenario s
Parameters	
$c_{k,l}^1$	Fixed cost for train route l of rolling stock unit k
$c_{r,t}^2$	Traveling cost for trip r starting at timestamp t
$c_{\sigma(r),r}^3$	Unit cost of a connection operation between trips $\sigma(r)$ and r
c_g^4	Fixed cost for arranging one crew member in crew group g
$\bar{T}_r^R, \bar{T}_r^R$	Earliest/latest feasible departure timestamps of trip r
$\bar{T}_r^C, \bar{T}_r^C$	Earliest/latest feasible start timestamps of meal/rest task r
$\bar{C}_p^R, \bar{C}_p^C$	Cost for rolling stock path / crew task sequence p
$h_{r,r'}^{\min}$	Minimum headway between the departure timestamps of trips r and r'
$N_s^{\min}$	Minimum number of trips with the associated time window of passenger frequency requirement scenario s
$N_a^{\max}$	Maximum number of covering rolling stock units on segment a at one timestamp
T_r^R	Traveling time of trip r
$\alpha_{p,r,t}$	Binary indicator, =1 if trip r is included in path p and starts at timestamp t ; =0 otherwise
$\beta_{p,r,t}$	Binary indicator, =1 if rolling stock path p passes through physical segment a from timestamp t ; =0 otherwise
$\gamma_{p,r}$	Binary indicator, =1 if task r is covered by crew task sequence p ; =0 otherwise
Decision variables	
λ_p	Binary variables, =1 if rolling stock path p is selected; =0 otherwise
μ_p	Binary variables, =1 if crew task sequence p is selected; =0 otherwise

4.2 Objective function and system constraints

We formulate the general path-based objective functions and constraints based on the overall path set $\mathcal{P}^R = \bigcup_{k \in \mathcal{K}} \mathcal{P}_k^R = \bigcup_{l \in \mathcal{L}_k, k \in \mathcal{K}} \mathcal{P}_{k,l}^R$ for rolling stock units and train routes, and the overall task sequence set $\mathcal{P}^C = \bigcup_{g \in \mathcal{G}} \mathcal{P}_g^C$ for all crew groups. Specifically, we first formulate the objective function in Eq. (1), which aims to minimize the total cost of rolling stock paths and crew task sequences, where parameters $\bar{C}_p^R$ and $\bar{C}_p^C$ denote the costs for rolling stock path $p \in \mathcal{P}^R$ and crew task sequence $p \in \mathcal{P}^C$, respectively. In particular, the second term of Eq. (1) aims to minimize the number of assigned crew members in each group to cover all the in-service trips. For the case when the maximum crew capacity

must be imposed as a hard constraint, we can set a large penalty value for the maximum crew capacity violation.

$$\min \sum_{p \in \mathcal{P}^R} C_p^R \cdot \lambda_p + \sum_{p \in \mathcal{P}^C} C_p^C \cdot \mu_p \quad (1)$$

Note that since the operational requirements, i.e., train route selection, trip connection time, crew task flow balance, meal/rest tasks, and crew groups, are only associated with individual rolling stock paths and crew task sequences (i.e., columns), these operational requirements are part of the pricing sub-problems that will be presented in Section 5.4 to generate feasible columns to this model. The feasibility and consistency of the model are guaranteed by three groups of system constraints formulated below, where the rolling stock scheduling related constraints, crew scheduling coupling constraints, and the domain of the decision variables are expressed in Groups I, II, and III, which are described as follows:

Group I: Rolling stock scheduling constraints

$$\sum_p \mathbb{1}_{p \in \mathcal{P}_k^R} \lambda_p = 1 \quad \forall k \in \mathcal{K} \quad (2)$$

$$\sum_{p \in \mathcal{P}^R} \alpha_{p,r,t} \lambda_p + \sum_{p \in \mathcal{P}^R} \lambda_p \sum_{t'=t+1}^{\min\{t+h_{r,r'}^{\min}-1, |\mathcal{T}|\}} \alpha_{p,r',t'} \leq 1 \quad \forall t \in \mathcal{T}, (r, r') \in \mathcal{V} \quad (3)$$

Constraints (2) ensure that each rolling stock unit $k \in \mathcal{K}$ is assigned to one rolling stock path $p \in \mathcal{P}_k^R$. Then, constraints (3) are the safety headway constraints among trips, where $\alpha_{p,r,t}$ is equal to 1 if and only if trip r is included in path p and starts at timestamp t . For operational safety, the minimum headway $h_{r,r'}^{\min}$ should be guaranteed between the departure timestamps of trips r and r' . Compared with the headway formulation in existing studies (Caprara et al., 2002; Cacchiani et al., 2008), considering that the trip traveling times are fixed from one to the next operating stations, we simplified the headway constraints by guaranteeing no conflicts between two trips only at the departure time to avoid enumerating a large number of incompatible pairs. Specifically, we pre-generate a set $\mathcal{V}$ of potential conflicting trip pairs, which run on the same physical segment within overlapping trip departure time windows, such that $\mathcal{V} = \{(r, r') \in \mathcal{R}^T \times \mathcal{R}^T \mid a(r) = a(r')\}$, where $a(r)$ and $a(r')$ denote the associated physical segments of trips r and r' respectively. Then, given a timestamp t and a pair of trips $(r, r') \in \mathcal{V}$, if trip r starts from timestamp t , then the minimum headway requirement prevents trip r' from starting within incompatible time window $\mathbb{Z}_{[t+1, \min\{t+h_{r,r'}^{\min}-1, |\mathcal{T}|\}]}$. Next, since departure time windows for trips have been predetermined, these constraints remain valid only within time windows $\mathcal{T}_r$ and $\mathcal{T}_{r'}$ for trips r and r' respectively. Lastly, we note that the multi-train composition operation requirement can be satisfied with such safety headway constraints (Wang et al., 2024). In other words, if a trip r starts from timestamp t on one physical segment, this segment can also be serviced by other trips which start from the same timestamp.

$$\sum_{p \in \mathcal{P}^R} \sum_{t \in \mathcal{T}_s} \beta_{p,a,t} \cdot \lambda_p \geq N_s^{\min} \quad \forall s \in \mathcal{S}_a, a \in \mathcal{A} \quad (4)$$

Constraints (4) impose the minimum trip frequency constraints, where $\beta_{p,a,t}$ is equal to 1 if and only if path p passes through physical segment a from timestamp t . Considering that different trip frequency requirements should be adopted in different physical segments and periods, we pre-generate a set $\mathcal{S}_a$ of trip frequency requirement scenarios (indexed by s) within different time windows for physical segment a . Then, based on the input data of the passenger demand over multiple scenarios (e.g., 5 working days), we first calculate the average passenger demand on physical segments a within the associated time windows $\mathcal{T}_s$ of scenarios $s \in \mathcal{S}_a$. Next, by dividing the average passenger demand by the rolling stock capacity, we obtain the minimum required numbers of trips $N_s^{\min}$ within the involved time windows $\mathcal{T}_s$ of scenarios s .

$$\sum_{p \in \mathcal{P}^R} \beta_{p,a,t} \cdot \lambda_p \leq N_a^{\max} \quad \forall a \in \mathcal{A}, t \in \mathcal{T} \quad (5)$$

Constraints (5) are the covering constraints of rolling stock units for physical segment a at timestamp t . With the multi-train composition mode, we assume the maximum number of rolling stock units (i.e., $N_a^{\max}$), which are permitted to cover one physical segment a at one timestamp.

Group II: Crew scheduling coupling constraints

$$\sum_{p \in \mathcal{P}^R} \alpha_{p,r,t} \cdot \lambda_p - \sum_{p \in \mathcal{P}^C} \gamma_{p,r} \cdot \mu_p = 0 \quad \forall t \in \mathcal{T}_r, r \in \mathcal{R}^T \quad (6)$$

Constraints (6) represent the coupling relationships between rolling stock paths and crew task sequences. Specifically, if trip r starts from timestamp t with one rolling stock path (i.e., $\sum_{p \in \mathcal{P}^R} \alpha_{p,r,t} \cdot \lambda_p = 1$), it must be covered by one crew task sequence (i.e., $\sum_{p \in \mathcal{P}^C} \gamma_{p,r} \cdot \mu_p = 1$), where $\gamma_{p,r}, \forall p \in \mathcal{P}^C, r \in \mathcal{R}^T \cup \mathcal{R}^M \cup \mathcal{R}^R$ is a binary parameter with setting to 1 if and only if task r is covered by crew task sequence p .

Group III: Range of the decision variables

All the rolling stock path and crew task sequence selection decisions are binary variables, as formulated in the following constraints.

$$\lambda_p \in \{0, 1\} \quad \forall p \in \mathcal{P}^R \quad (7)$$

$$\mu_p \in \{0, 1\} \quad \forall p \in \mathcal{P}^C \quad (8)$$

With Eqs. (1)–(8), the path-based mathematical formulation M1 is modeled for the JRCSP. Compared with existing path-based models at the rolling stock scheduling level in the literature (e.g., Cacchiani et al. (2008, 2010a)), we extend the set partitioning constraints to incorporate rolling stock path selection constraints (2), extend set covering constraints to incorporate minimum trip frequency requirement constraints (4), and extend the set packing constraints to incorporate headway constraints (3) and the maximum rolling stock covering constraints (5). Lastly, we observe that the complexity of the proposed model is related to the number of trips and rolling stock units. In addition, there is possibly a large number of rolling stock paths and crew task sequences, as well as their coupling constraints (6). Thus, it is challenging and not practical to solve the model M1 directly by enumerating all the rolling stock paths and crew task sequences. Based on this observation, to find high-quality solutions within an acceptable computing time, an efficient and exact branch-and-price-and-cut algorithm is next designed to solve the path-based model M1, which is discussed in the following section.

5 Solution algorithm

We develop a branch-and-price-and-cut (BPC) algorithm that combines a branch-and-price (B&P) and Benders decomposition to efficiently solve the JRCSP. In addition, we generalize this solution framework and design a novel scheme tailored to the JRCSP with several acceleration techniques to improve the performance of the approach. First, we pre-process the time windows for departure times of trips to strengthen the path-based model into a more compact formulation (Section 5.1). Then, at each node of the search tree, Benders decomposition is embedded to solve the LP relaxation of model M1 (Section 5.2). To avoid a large number of Benders cuts in the Benders decomposition process, the lower bound from the Benders master problem is lifted by using lower-bound lifting (LBL) valid inequalities (Section 5.3). Subsequently, the decomposed Benders master problem (BMP) and Benders sub-problem (BSP) can be viewed as the LP relaxations of the rolling stock scheduling and crew scheduling problems, respectively. Both problems can be tackled by column generation algorithms (Section 5.4), in which pricing sub-problems are solved by finding the shortest paths for rolling stock units and crew groups, respectively. If an integer optimal solution is obtained for model M1, it means that a feasible integer solution and a new upper bound are obtained at the corresponding branch-and-bound (B&B) node. Otherwise, two new nodes will be created by branching on a fractional variable

(Section 5.5). At each node, Benders decomposition with column generation is used again to solve the LP relaxation of model M1 with the branching conditions. The aforementioned approach iterates with the bounding and pruning processes until the termination criteria are met, and the best encountered feasible solution is returned as an (exact) optimal solution to the model M1. Lastly, Section 5.6 refers to a series of acceleration techniques to improve the performance of the solution approach.

5.1 Pre-processing

We observe from the path-based model M1 that the complexity of the rolling stock and crew column generation process is largely dependent on the trip departure time windows (i.e., $\mathcal{T}_r$). For this reason, this section introduces a pre-processing method that shrinks time windows for departure times of trips, which can strengthen the path-based model M1 into a more compact formulation. Specifically, we denote by $r_{q(l)}$ the q -th trip with respect to train route $l \in \mathcal{L}_k$ for rolling stock unit k . Then, the earliest timestamp to start the q -th trip (denoted by $\underline{T}_{r_q(l)}$) can be calculated by Eq. (9), where $\underline{T}_{r_0(l)}$ is the earliest timestamp to start the first trip, and $T_{r_{q'}(l), r_{q'+1}(l)}^{R \min}$ is the minimum connection time between the q' -th and $(q' + 1)$ -th trips based on train route l . Similarly, the latest timestamp to start the q -th trip (denoted by $\bar{T}_{r_q(l)}$) can be calculated by Eq. (10), where $\bar{T}_{r_0(l)}$ is the latest timestamp to start the first trip and $T_{r_{q'}(l), r_{q'+1}(l)}^{R \max}$ is the maximum connection time between the q' -th and $(q' + 1)$ -th trips based on train route l . Therefore, the time window $\mathcal{T}_r$ to start trip r is updated by $\mathcal{T}_r \leftarrow \mathcal{T}_r \cap \mathbb{Z}_{[\underline{T}_r, \bar{T}_r]}$ in the column generation process to orient rolling stock units to schedule trips within effective time windows.

$$\underline{T}_{r_q(l)} = \underline{T}_{r_0(l)} + \sum_{q'=0}^{q-1} \left(T_{r_{q'}(l)}^R + T_{r_{q'}(l), r_{q'+1}(l)}^{R \min} \right) \quad \forall q \in \{1, 2, \dots, |\mathcal{R}_l^T| - 1\}, l \in \mathcal{L}_k, k \in \mathcal{K} \quad (9)$$

$$\bar{T}_{r_q(l)} = \bar{T}_{r_0(l)} + \sum_{q'=0}^{q-1} \left(T_{r_{q'}(l)}^R + T_{r_{q'}(l), r_{q'+1}(l)}^{R \max} \right) \quad \forall q \in \{1, 2, \dots, |\mathcal{R}_l^T| - 1\}, l \in \mathcal{L}_k, k \in \mathcal{K} \quad (10)$$

5.2 Benders decomposition

In the path-based model M1, there possibility exists a large number of coupling constraints (6) between rolling stock paths and crew task sequences, and it is still challenging and time-consuming to solve the LP relaxation of model M1 by the column generation method directly. Furthermore, we observe that once variables $\lambda_p, p \in \mathcal{P}^R$ are fixed, the reduced problem becomes a crew scheduling problem with fixed trips, which is much easier to solve. Thus, based on this observation, we first decompose the LP relaxation of model M1 by using the Benders decomposition method in this section. Specifically, let $\mathbf{\Lambda}$ be the set of vectors for the rolling stock path selection variables that satisfy $0 \leq \lambda_p \leq 1$ and constraints (2)–(5), i.e., $\mathbf{\Lambda} = \left\{ \boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_p, \dots, \lambda_{|\mathcal{P}^R|})^T \mid 0 \leq \lambda_p \leq 1, \text{ constraints (2)–(5), } p \in \mathcal{P}^R \right\}$. For any given rolling stock path selection solution vector $\bar{\boldsymbol{\lambda}} \in \mathbf{\Lambda}$, the LP relaxation of model M1 is reduced to the BSP M2 in Eq. (11) involving only crew task sequence selection variables (i.e., μ_p) and related constraints. In this LP model, we denote the dual variables associated with constraints (11b)–(11c) by $\boldsymbol{\theta}^1 = \{\theta_{r,t}^1 \mid t \in \mathcal{T}_r, r \in \mathcal{R}^T\}$, $\boldsymbol{\theta}^2 = \{\theta_p^2 \mid p \in \mathcal{P}^R\}$.

$$\text{M2:} \begin{cases} \min & \sum_{p \in \mathcal{P}^C} C_p^C \cdot \mu_p & (11a) \\ \text{s.t.} & \sum_{p \in \mathcal{P}^R} \alpha_{p,r,t} \cdot \bar{\lambda}_p - \sum_{p \in \mathcal{P}^C} \gamma_{p,r} \cdot \mu_p = 0 \quad \forall t \in \mathcal{T}_r, r \in \mathcal{R}^T & (11b) \\ & \mu_p \leq 1 \quad \forall p \in \mathcal{P}^C & (11c) \\ & \mu_p \geq 0 \quad \forall p \in \mathcal{P}^C & (11d) \end{cases}$$

Based on Proposition 1 (see electronic companion EC. B for details), BSP M2 is always feasible and bounded for any given feasible vector $\bar{\boldsymbol{\lambda}} \in \mathbf{\Lambda}$, which implies that it is sufficient to add only

Benders optimality cuts into the BMP. Thus, let Δ denote the polyhedron defined by the dual solutions of the BSP M2 and let $\overline{\mathcal{O}}_\Delta$ be the set of current enumerated extreme points with respect to dual solutions of the BSP M2. Then, by introducing the auxiliary variable η , the LP relaxation of model M1 can be reformulated as the following relaxed BMP M3 in Eq. (12) with Benders optimality cuts in constraints (12c):

$$\text{M3:} \begin{cases} \min & \sum_{p \in \mathcal{P}^R} C_p^R \cdot \lambda_p + \eta & (12a) \\ \text{s.t.} & \text{Constraints (2)–(5)} & (12b) \\ & \eta \geq \sum_{r \in \mathcal{R}^T} \sum_{t \in \mathcal{T}_r} \sum_{p \in \mathcal{P}^C} \alpha_{p,r,t} \theta_{r,t}^1 \cdot \lambda_p + \sum_{p \in \mathcal{P}^C} \theta_p^2 \quad \forall (\theta^1, \theta^2) \in \overline{\mathcal{O}}_\Delta & (12c) \\ & \lambda_p \leq 1 \quad \forall p \in \mathcal{P}^R & (12d) \\ & \lambda_p \geq 0 \quad \forall p \in \mathcal{P}^R & (12e) \\ & \eta \geq 0 & (12f) \end{cases}$$

5.3 Lower-bound-lifting (LBL) valid inequalities

In our preliminary experiments, we observed that the optimality gap of the relaxed BMP M3 is large with low-quality lower bounds in the early iterations of the Benders decomposition process since the BSP M2 is projected out from the relaxed BMP M3. Thus, it is inevitable that a large number of Benders cuts are generated to improve lower bounds and close the relative gap (Adulyasak et al., 2015). To address this issue, we lift the lower bound of the relaxed BMP M3 by using the lower-bound-lifting (LBL) valid inequalities, which contain some information with respect to the crew scheduling cost that is removed from the BMP. Particularly, it is possible to formulate the LBL cut to represent a lower bound by estimating crew scheduling costs without solving the BSP.

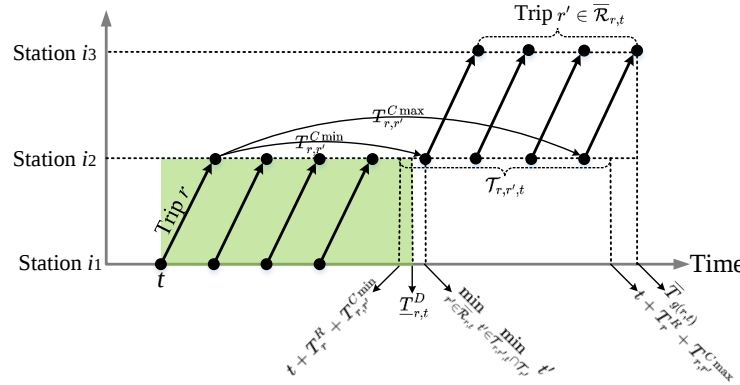


Figure 4: Illustration of the LBL cut

Specifically, for each trip r starting at timestamp t , we first denote the set $\mathcal{T}_{r,r',t}$ in Eq. (13) to represent possible departure timestamps of adjacent next trip r' , if trip r is possible to connect with trip r' , as shown in Fig. 4. In this equation, $g(r,t)$ denotes the associated crew group for trip r starting at timestamp t , and $\overline{T}_{g(r,t)}$ denotes the upper bound of the time window to schedule crew members in crew group $g(r,t)$.

$$\mathcal{T}_{r,r',t} = \{t' \mid t + T_r^R + T_{r,r'}^{C \min} \leq t' \leq \min \{\overline{T}_{g(r,t)}, t + T_r^R + T_{r,r'}^{C \max}\} \} \quad (13)$$

Then, we denote all possible next adjacent trips by the set $\overline{\mathcal{R}}_{r,t} = \{r' \mid r' \in \mathcal{R}^T, \mathcal{T}_{r,r',t} \cap \mathcal{T}_{r'} \neq \emptyset, r' \neq r\}$ for trip r starting at timestamp t . Next, we calculate one timestamp before the minimum departure time within the possible next trips in set $\overline{\mathcal{R}}_{r,t}$ (i.e., $\underline{T}_{r,t}^D$) for trip r starting at timestamp t in Eq. (14).

Particularly, when no possible next adjacent trip is involved for trip r starting from timestamp t (i.e., empty set $\overline{\mathcal{R}}_{r,t} = \emptyset$), we let $\underline{T}_{r,t}^D = t$ in Eq. (14a) to ensure the feasibility of the LBL cut in Eq. (15).

$$\underline{T}_{r,t}^D = \begin{cases} t, & \text{if } \overline{\mathcal{R}}_{r,t} = \emptyset \\ \min_{r' \in \overline{\mathcal{R}}_{r,t}} \min_{t' \in \mathcal{T}_{r,r',t} \cap \mathcal{T}_{r'}} (t' - 1), & \text{if } \overline{\mathcal{R}}_{r,t} \neq \emptyset \end{cases} \quad (14a)$$

$$(14b)$$

For one trip starting from one timestamp t , the minimum number of required crew members can be calculated from the current timestamp t to one timestamp before the minimum departure time of the next adjacent trip of trip r (i.e., $\min\{\underline{T}_{r,t}^D, \overline{T}_{g(r,t)}\}$), i.e., the shadow area filled with the green color in Fig. 4. Thus, for any timestamp t , we let variable η be no smaller than the summation of estimated operational cost of required crew members within the time window $[t, \min\{\underline{T}_{r,t}^D, \overline{T}_{g(r,t)}\}]$ on all trips $r \in \mathcal{R}^T$ and rolling stock paths $p \in \mathcal{P}^R$ (see Proposition 2 in electronic companion EC. B for proof details).

$$\eta \geq \sum_{p \in \mathcal{P}^R} \sum_{r \in \mathcal{R}^T} \sum_{t'=t}^{\min\{\underline{T}_{r,t}^D, \overline{T}_{g(r,t)}\}} c_{g(r,t)}^4 \alpha_{p,r,t} \lambda_p \quad \forall t \in \mathcal{T} \quad (15)$$

Lastly, according to the relaxed BMP M3 and LBL valid inequalities (15), the enhanced and relaxed BMP M4 can be formulated in Eq. (16).

$$\text{M4: } \begin{cases} \min & \sum_{p \in \mathcal{P}^R} C_p^R \cdot \lambda_p + \eta \\ \text{s.t.} & \text{Constraints (12b) -- (12f)} \\ & \text{Constraints (15)} \end{cases} \quad (16a)$$

$$(16b)$$

$$(16c)$$

5.4 Column generation

From the BSP M2 and the relaxed BMP M4, we can see that there is potentially a very large number of possible paths for rolling stock units and crew groups, and it is not practical to enumerate all of them. Column generation has been used in rolling stock scheduling problems by characterizing and combining space-time paths for rolling stock units (Cacchiani et al., 2008, 2010a). Therefore, in this section, we aim to solve the BSP M2 and relaxed BMP M4 through a customized column generation framework by embedding Benders/LBL cuts and a procedure to handle meal/rest requirements. Specifically, at the beginning of the column generation process, two sets of artificial columns with large costs are first included in the restricted master problems (i.e., BSP M2 and relaxed BMP M4), which can ensure the feasibility of BSP and BMP models. Specifically, let $\overline{\mathcal{P}}^R$ and $\overline{\mathcal{P}}^C$ be the initial column pools composed of paths for all rolling stock units and task sequences for all crew groups, respectively. After solving the BSP M2 and relaxed BMP M4, the pricing sub-problem is constructed to find new columns based on the dual variables (i.e., π, θ) of constraints, which could reduce the objective values of the restricted master problems. If all the reduced costs for the pricing sub-problems obtained by the label correcting (LC) algorithm are greater than 0, an optimal solution is obtained for the restricted master problems (i.e., BSP M2 and relaxed BMP M4). Otherwise, the newly generated columns are added to the column pool in the next iterations. The column generation procedure is terminated when no more columns are found with negative reduced costs. Detailed column generation methods for solving the relaxed BMP M4 and BSP M2 are discussed in Sections 5.4.1 and 5.4.2, respectively.

5.4.1 Pricing sub-problems and label correcting for the relaxed BMP M4

In order to formulate the pricing sub-problems of the relaxed BMP M4, we first denote the dual variables corresponding to constraints (2)–(5), (12c) and (15) by π^1 – π^6 respectively, i.e., $\pi^1 = \{\pi_k^1\}_{k \in \mathcal{K}}$, $\pi^2 = \{\pi_{t,r,r'}^2\}_{t \in \mathcal{T}, (r,r') \in \mathcal{V}}$, $\pi^3 = \{\pi_{s,a}^3\}_{s \in \mathcal{S}_a, a \in \mathcal{A}}$, $\pi^4 = \{\pi_{a,t}^4\}_{a \in \mathcal{A}, t \in \mathcal{T}}$, $\pi^5 = \{\pi_{(\theta^1, \theta^2)}^5\}_{(\theta^1, \theta^2) \in \overline{\mathcal{O}}_\Delta}$,

$\pi^6 = \{\pi_t^6\}_{t \in \mathcal{T}}$. Thus, the reduced cost for each path p of train route l and rolling stock unit k can be calculated in Eqs. (17)–(20). Similar to Eqs. (13)–(14), we let $\overline{T}_{r,t}^D$ denote one timestamp after the maximum departure time of the previous trip for trip r which starts at timestamp t , let $\underline{T}_{g(r,t)}$ denote the lower bound of the time window to schedule crew members in crew group $g(r,t)$, and let $\mathcal{T}_{r',r,t}$ denote the possible departure timestamps of trip r' , if trip r' is possible to connect with trip r (which starts at timestamp t). Particularly, we observe that the third and fourth terms in Eq. (17) correspond to the headway constraints (12b), and then the seventh and eighth terms correspond to the Benders cut constraints (12f) and LBL cut constraints (15), respectively, which could orient the LC algorithm to find the best space-time paths for rolling stock units with the consideration of newly generated (cut) constraints in the Benders decomposition process.

$$\begin{aligned}
\overline{C}_{k,l,p}^R = & C_p^R - \pi_k^1 - \sum_{t \in \mathcal{T}} \sum_{(r',r) \in \mathcal{V}} \alpha_{p,r,t} \pi_{t,r',r}^2 - \sum_{t \in \mathcal{T}} \sum_{(r,r') \in \mathcal{V}} \sum_{t' = \max\{t - h_{r,r'}^{\min} + 1, 0\}}^{t-1} \alpha_{p,r',t'} \pi_{t',r,r'}^2 \\
& + \sum_{a \in \mathcal{A}} \sum_{s \in \mathcal{S}_a} \sum_{t \in \mathcal{T}_s} N_s^{\min} \beta_{p,a,t} \pi_{s,a}^3 - \sum_{t \in \mathcal{T}} \sum_{a \in \mathcal{A}} N_a^{\max} \beta_{p,a,t} \pi_{a,t}^4 \\
& - \sum_{(\theta^1, \theta^2) \in \overline{\mathcal{O}}_\Delta} \sum_{r \in \mathcal{R}_l^T} \sum_{t \in \mathcal{T}_r} \theta_{r,t}^1 \alpha_{p,r,t} \pi_{(\theta^1, \theta^2)}^5 + \sum_{t \in \mathcal{T}} \sum_{r \in \mathcal{R}_l^T} \sum_{t' = \max\{\overline{T}_{r,t}^D, \underline{T}_{g(r,t)}\}}^t \alpha_{p,r,t'} \pi_{t'}^6 \\
& \forall p \in \mathcal{P}_{k,l}^R, l \in \mathcal{L}_k, k \in \mathcal{K}
\end{aligned} \tag{17}$$

$$\overline{T}_{r,t}^D = \begin{cases} t, & \text{if } \overline{\mathcal{R}}'_{r,t} = \emptyset \\ \max_{r' \in \overline{\mathcal{R}}'_{r,t}} \max_{t \in \mathcal{T}_{r',r,t} \cap \mathcal{T}_r} (t+1), & \text{if } \overline{\mathcal{R}}'_{r,t} \neq \emptyset \end{cases} \tag{18}$$

$$\overline{\mathcal{R}}'_{r,t} = \{r' \mid r' \in \mathcal{R}^T, \mathcal{T}_{r',r,t} \cap \mathcal{T}_r \neq \emptyset, r' \neq r\} \tag{19}$$

$$\mathcal{T}_{r',r,t'} = \left\{ t' \mid t - T_{r'}^R - T_{r',r}^C \max \leq t' \leq \max \left\{ \underline{T}_{g(r,t)}, t - T_{r'}^R - T_{r',r}^C \min \right\} \right\} \tag{20}$$

The pricing sub-problem of the relaxed BMP M4 aims to find new paths for rolling stock units, which could reduce the objective value of the relaxed BMP M4. Therefore, the pricing sub-problem associated with train route $l \in \mathcal{L}_k$ and rolling stock unit $k \in \mathcal{K}$ (i.e., $SPR_{k,l}$) becomes a problem to find a path $p \in \mathcal{P}_{k,l}^R$ with the minimum reduced cost (i.e., $\min_{p \in \mathcal{P}_{k,l}^R} \overline{C}_{k,l,p}^R$). Thus, only the flow balance constraints should be imposed to keep the feasibility of the global path for rolling stock units. In addition, the complexity of the pricing sub-problem is closely related to the number of trips, trip connections, and departure timestamps. Furthermore, the pricing problem of the relaxed BMP M4 can be formulated as a shortest path problem after applying the topological sort method. This problem can be solved by the LC algorithm in parallel without keeping track of a (potentially very large) number of paths in the searching process, which makes the procedure highly scalable (see the electronic companion EC. C.1 for the LC algorithm details). In addition, the branching rule verification step is also required to exclude infeasible trips, trip connections, and departure timestamps from the feasible solution space. After applying the LC algorithm, we obtain the shortest path p^* as the optimal solution of model $SPR_{k,l}$. If $\overline{C}_{k,l,p^*}^R \geq 0$ (i.e., all columns with positive reduced costs), the column generation procedure terminates. Otherwise, the optimal solution of $SPR_{k,l}$ becomes a new column which will be inserted to column pool $\overline{\mathcal{P}}^R$ in subsequent iterations.

5.4.2 Pricing sub-problems and multi-stage label correcting for the BSP M2

To characterize the movements of crew members in crew groups, the fixed solution of rolling stock path columns obtained from the relaxed BMP M4 will be embedded into the BSP M2 for scheduling crew tasks. Specifically, given a vector $\overline{\lambda}$ to represent the solution of obtained decision variables of

the associated relaxed BMP M4, we use set $\bar{\mathcal{R}}_{\lambda}^C = \{(r, t) \mid \sum_{p \in \bar{\mathcal{P}}^R} \bar{\lambda}_p \alpha_{p,r,t} > 0, \forall t \in \mathcal{T}_r, r \in \mathcal{R}^T\}$ to represent fixed trips and set $\bar{\mathcal{R}}_g^C = \{(r, t) \mid g \in \mathcal{G}, (r, t) \in \bar{\mathcal{R}}^C\}$ to represent fixed trips available for crew group $g \in \mathcal{G}$. At the same time, we denote the dual variables associated with constraints (11b) by θ^1 , i.e., $\theta^1 = \{\theta_{r,t}^1\}_{t \in \mathcal{T}_r, r \in \mathcal{R}^T}$. Thus, the reduced cost for each task sequence p belonging to crew group g can be calculated by Eq. (21):

$$\bar{C}_{g,p}^C = C_{g,p}^C - \sum_{r \in \bar{\mathcal{R}}_g^C} \sum_{t \in \mathcal{T}_r} \gamma_{p,r} \cdot \theta_{r,t}^1 \quad \forall p \in \mathcal{P}_g^C, g \in \mathcal{G} \quad (21)$$

The pricing sub-problem of the BSP M2 aims to find task sequences for crew members, which might improve the quality of the solution for the BSP M2. The pricing sub-problem for crew group g (i.e., SPC_g) is solved to find a task sequence $p \in \mathcal{P}_g^C$ with the minimum reduced cost (i.e., $\min_{p \in \mathcal{P}_g^C} \bar{C}_{g,p}^C$). Unlike the pricing sub-problems for rolling stock units in Section 5.4.1, the meal/rest requirements must also be imposed to generate feasible task sequences for crew groups. Thus, the pricing sub-problems for crew groups are the shortest path problems with additional constraints, and they cannot be solved directly by using the standard LC algorithm. At the same time, each meal/rest task (e.g., lunch meal task) is assigned once to a crew member within its admissible time windows (e.g., 11:00–13:00). Thus, it is possible to partition the whole task sequence finding process into multiple stages using the division of meal/rest tasks. Then, the sub-task-sequence finding process at a single stage can be modeled as the shortest path problem with flow balance constraints, which can be solved by the LC algorithm directly. Lastly, the sub-task-sequences at individual stages can be merged and joined to generate globally feasible task sequences with satisfying crew meal/rest requirements.

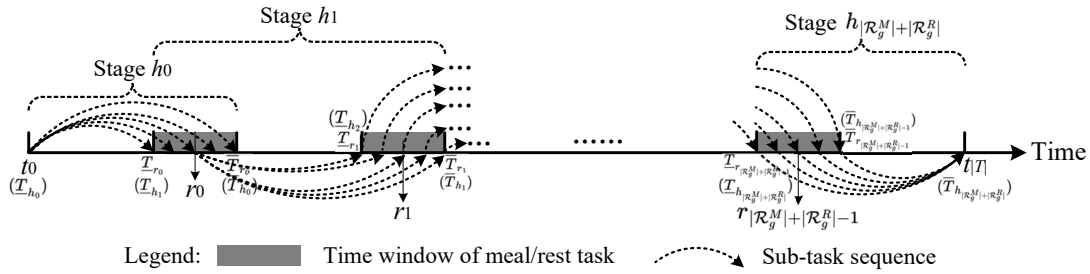


Figure 5: Illustration of the multi-stage LC algorithm

To this end, we develop a multi-stage LC algorithm to tackle additional meal/rest requirements. Specifically, as illustrated in Fig. 5, the entire planning horizon $[t_0, t_{|T|}]$ is first partitioned into a finite number of stages, denoted by $\mathcal{H} = \{h_0, h_1, h_2, \dots, h_{|\mathcal{R}_g^M|+|\mathcal{R}_g^R|}\}$, where $|\mathcal{R}_g^M| + |\mathcal{R}_g^R|$ is the total number of meal/rest tasks for crew members in crew group g . In addition, $[\underline{T}_r, \bar{T}_r]$ is the time window of meal/rest task $r \in \mathcal{R}_g^M \cup \mathcal{R}_g^R$. Then, in the first stage h_0 , it only calls the LC algorithm once to determine sub-task-sequences from virtual source at time t_0 to trip tasks within time window $[\underline{T}_{r_0}, \bar{T}_{r_0}]$ of first meal/rest task r_0 . Next, in the middle stages $h_1, \dots, h_{|\mathcal{H}|-2}$, it needs to call the LC algorithm with $(\bar{T}_{\sigma(r)} - \underline{T}_{\sigma(r)})$ times to determine sub-task-sequences which start from trip tasks within the time window $[\underline{T}_{\sigma(r)}, \bar{T}_{\sigma(r)}]$ of meal/rest task $\sigma(r)$, and end with trip tasks within time window $[\underline{T}_r, \bar{T}_r]$ of meal/rest task r , where $\sigma(r)$ is the adjacent previous meal/rest task of meal/rest task r . Lastly, it also applies one time the LC algorithm to determine sub-task-sequences from the virtual sink to trip task within $[\underline{T}_{|\mathcal{R}_g^M|+|\mathcal{R}_g^R|-1}, \bar{T}_{|\mathcal{R}_g^M|+|\mathcal{R}_g^R|-1}]$ associated with the last meal/rest task $r_{|\mathcal{R}_g^M|+|\mathcal{R}_g^R|-1}$.

After applying the multi-stage LC algorithm at individual stages (see electronic companion EC. C.2 for the multi-stage LC algorithm details), the global feasible task sequences are generated by merging sub-task-sequences at stages while satisfying the required time for each meal/rest task. Then, the global task sequence p^* from virtual source to virtual sink with the minimum reduced cost is obtained

as the optimal solution of model SPC_g . If $\overline{C_{g,p}^C} \geq 0$ (i.e., all columns with the positive reduced costs), the column generation procedure terminates. Otherwise, the optimal solution of SPC_g becomes a new column, which will be inserted into the column pool $\overline{\mathcal{P}}^C$ in subsequent iterations.

5.5 Branch-and-bound (B&B)

After solving the linear programming relaxation of path-based model M1 by using Benders decomposition and column generation at each B&B node, if an integer solution is obtained, this solution can be regarded as a feasible solution corresponding to the model M1; otherwise, the branching strategy is required to transform the obtained fractional solutions into integer feasible solutions. This section presents the details of the branching, bounding, and pruning rules, which are used to determine an optimal integer solution.

(1) Branching rules: It is not tractable to branch on column variables (i.e., λ_p, μ_p) directly due to a large number of columns for rolling stock units and crew groups in this model. Instead, we first make use of the following four groups of auxiliary branching variables to represent (a) the assignments between rolling stock units and train routes; (b) the departure timestamps of trips; (c) the assignments between crew groups and trips; and (d) the connections of trips for crew members, where $\overline{\lambda} = \{\overline{\lambda}_p \mid p \in \overline{\mathcal{P}}^R\}$ and $\overline{\mu} = \{\overline{\mu}_p \mid p \in \overline{\mathcal{P}}^C\}$ are the optimal solutions to the associated relaxed BMP M4 and BSP M2 at the current node, respectively. We can easily verify that a solution to the path-based model M1 is integral in the JRCSP if and only if all auxiliary variables (i.e., $\overline{\mathbf{u}}, \overline{\mathbf{x}}, \overline{\mathbf{n}}, \overline{\mathbf{z}}$) are integers.

$$\begin{aligned}
 (a) \ \overline{\mathbf{u}} &= \left\{ \overline{u}_{k,l} = \sum_{p \in \overline{\mathcal{P}}^R} \overline{\lambda}_p \sum_{r \in \mathcal{R}_l^T} \sum_{t \in \mathcal{T}_r} \alpha_{p,r,t} \mid \forall l \in \mathcal{L}_k, k \in \mathcal{K} \right\} \\
 (b) \ \overline{\mathbf{x}} &= \left\{ \overline{x}_{r,t} = \sum_{p \in \overline{\mathcal{P}}^R} \overline{\lambda}_p \alpha_{p,r,t} \mid \forall t \in \mathcal{T}_r, r \in \mathcal{R}^T \right\} \\
 (c) \ \overline{\mathbf{n}} &= \left\{ \overline{n}_{g,r} = \sum_{p \in \overline{\mathcal{P}}^C, g(p)=g} \overline{\mu}_p \gamma_{p,r} \mid \forall r \in \mathcal{R}^T \cup \mathcal{R}^M \cup \mathcal{R}^R, g \in \mathcal{G} \right\} \\
 (d) \ \overline{\mathbf{z}} &= \left\{ \overline{z}_{g,r,r'} = \sum_{p \in \overline{\mathcal{P}}^C, g(p)=g} \overline{\mu}_p \gamma_{p,r} \gamma_{p,r'} \mid \forall r, r' \in \mathcal{R}^T \cup \mathcal{R}_g^M \cup \mathcal{R}_g^R \cup \{0\}, r \neq r', g \in \mathcal{G} \right\}
 \end{aligned}$$

We then select a branching variable whose value is the nearest to 0.5 among these four groups of auxiliary branching variables and create two child nodes n_1 with =0 and n_2 with =1 for the selected branching variable. In order to preserve the structure of the BSP M2 and relaxed BMP M4 without adding more branching constraints, all branching rules are imposed into the pricing sub-problems (i.e., $SPR_{k,l}$ and SPC_g) by directly skipping certain label updating processes of the involved trips and tasks in the LC algorithm (see electronic companion EC. C for details). Lastly, the two new branching nodes will be added into the active node list (ANL).

(2) Bounding and pruning rules: We adopt the commonly used lower bound updating rules based on the minimum lower bound among branching nodes on the same layer, and upper bound updating rules based on the newly obtained feasible solutions. Then, we remove redundant branching nodes when infeasible solutions, integer solutions, or larger lower bounds than the upper bound are encountered at the current node (Lin and Kwan, 2016; Lusby et al., 2017; Wang et al., 2024).

5.6 Acceleration techniques

5.6.1 Dynamic constraint generation (DCG)

The solution efficiency of the proposed algorithm largely depends on the efficiency and difficulty of solving the relaxed BMP M4 model. We observe that the number of constraints (3)–(5) is too large to be handled explicitly to real-world instances and increases exponentially with the number of rolling stock units, trips, and considered timestamps. However, most of these constraints are inactive for an arbitrary feasible solution. For this reason, we use a dynamic constraint generation (DCG) technique to deal with them. In the beginning, we omit all the associated constraints (3)–(5) and construct an empty *constraint pool*. After performing Benders decomposition for one iteration, we check the solution of the relaxed BMP M4 and identify the violated constraints (3)–(5). If some constraints are violated but not yet in the constraint pool, we add them into the constraint pool and then update the relaxed BMP M4 associated with the constraint pool. Then, the relaxed BMP M4 is solved again. This DCG procedure repeats until no violated constraints can be found.

5.6.2 Pareto-optimal cuts (POC)

We observe that the dual variables of the BMP M4 has multiple optimal solutions (i.e., extreme points), and some solutions may be used to generate stronger Benders optimality cuts than other solutions. Along this line, the Pareto-optimal cut is introduced to construct stronger and non-dominated cuts to improve the convergence of the Benders decomposition process (Magnanti and Wong, 1981; Cordeau et al., 2001a). Specifically, we identify and generate a Pareto-optimal cut in Eq. (22) from the extreme point $(\tilde{\theta}^1, \tilde{\theta}^2)$, which could dominate the cuts generated from other extreme points $(\theta^1, \theta^2) \in \overline{\mathcal{O}}_\Delta$ if and only if satisfying the following condition for all $\lambda = \{\lambda_p\}_{p \in \overline{\mathcal{P}}^C} \in \Lambda$ with the strict inequality for at least one point.

$$\sum_{r \in \mathcal{R}^T} \sum_{t \in \mathcal{T}_r} \sum_{p \in \overline{\mathcal{P}}^R} \alpha_{p,r,t} \tilde{\theta}_{r,t}^1 \cdot \lambda_p + \sum_{p \in \overline{\mathcal{P}}^C} \tilde{\theta}_p^2 \geq \sum_{r \in \mathcal{R}^T} \sum_{t \in \mathcal{T}_r} \sum_{p \in \overline{\mathcal{P}}^R} \alpha_{p,r,t} \theta_{r,t}^1 \cdot \lambda_p + \sum_{p \in \overline{\mathcal{P}}^C} \theta_p^2 \quad \forall (\theta^1, \theta^2) \in \overline{\mathcal{O}}_\Delta \quad (22)$$

Lastly, based on the preliminary experiments, we observed that, although the Pareto-optimal cuts can certainly reduce the number of Benders decomposition iterations, the total computing time may increase due to the additional computing time required for solving the auxiliary models. To alleviate this effect, the maximum computing time is restricted (e.g., 1s) for solving the auxiliary models in the implementations.

5.6.3 Dynamic management of column pool (DMC)

We observe that a total of $\sum_{k \in \mathcal{K}} |\mathcal{L}_k|$ pricing sub-problems needs to be solved to generate new columns for rolling stock units at each iteration of the column generation process. In fact, it is possible to update the column pool for rolling stock units dynamically to simplify the column generation procedure by checking whether rolling stock columns may appear in the optimal solution. Specifically, after solving the relaxed BMP M4 in the column generation process at a node in the B&B tree, we use the criterion provided by Proposition 2 to dynamically remove some rolling stock columns associated with the specific train routes and rolling stock units. After applying this proposition, the column pool $\overline{\mathcal{P}}^R$ for rolling stock units is updated safely by removing and forbidding the generation of columns associated with the selected rolling stock units and train routes.

Proposition 2. *Let z_{M4} and UB be the current lower bound and upper bound on the optimal value of the relaxed BMP M4, respectively. If λ_p is a non-basic variable in the optimal solution to the relaxed BMP M4 and $z_{M4} + RC_{k,l} > UB$, where $RC_{k,l}$ is the reduced cost associated with rolling stock unit $k = k(p)$ and train route $l = l(p)$, then $\lambda_p = 0$ in any optimal solution of the relaxed BMP M4.*

In addition, to further reduce the size of the column pool, all columns that are infeasible with respect to safety headway constraints of trip r and associated departure time t will also be eliminated from the column pool if branching rule $\bar{x}_{r,t}$ is fixed to 1.

5.6.4 Column and cut initialization (CCI)

During the solution process, we observe that some of the generated columns and Benders cuts could still be valid at their child nodes. Thus, we could warm-start the Benders decomposition and column generation procedures by initializing columns and Benders cuts to avoid some repetitive column generations. Specifically, at a new B&B node, all valid columns with strict positive column selection solution values at their parent node will be inherited and reserved as the initial column pools $\overline{\mathcal{P}}_0^R$ and $\overline{\mathcal{P}}_0^C$ for rolling stock units and crew groups respectively, i.e., $\overline{\mathcal{P}}_0^R = \{p \mid \overline{\lambda}_p > 0, p \in \overline{\mathcal{P}}^R\}$, $\overline{\mathcal{P}}_0^C = \{p \mid \overline{\mu}_p > 0, p \in \overline{\mathcal{P}}^C\}$. Next, the method to inherit Benders cuts is presented in Proposition 3 to identify those valid Benders cuts. After applying this proposition, we let the initial set of Benders cuts include all Benders cut from nodes with only branching on variables $\overline{\mathbf{u}}, \overline{\mathbf{x}}$ and their parent nodes with branching on variables $\overline{\mathbf{n}}, \overline{\mathbf{z}}$.

Proposition 3. *For two child nodes n_1 and n_2 of parent node n_0 : (a) When one of the branching rules regarding auxiliary variables $\overline{\mathbf{u}}, \overline{\mathbf{x}}$ in the relaxed BMP M4 of node n_0 is selected for generating nodes n_1 and n_2 , the Benders cuts generated at node n_1 are still valid at both nodes n_1 and n_2 with their descendent nodes. At the same time, the Benders cuts generated at node n_2 are also valid at both nodes n_1 and n_2 with their descendent nodes. However, (b) when one of the branching rules regarding auxiliary variables $\overline{\mathbf{n}}, \overline{\mathbf{z}}$ in the BSP M2 of node n_0 is selected for generating nodes n_1 and n_2 , the Benders cuts generated at node n_1 (n_2) are only valid at the descendent nodes of node n_1 (n_2) with the same branch.*

5.6.5 Priority heuristic rule (PHR)

When the proposed BPC algorithm can find a feasible joint rolling stock and crew schedule solution with same global lower and upper bounds, this solution must be the optimal solution to the model M1. However, it is difficult to obtain a feasible joint solution in the early stages of the B&B process due to a large number of columns, especially in some large-scale instances. Given this concern, we design the priority rule to transform the linear lower bound solution at the B&B node into a feasible integer solution to update the (integer) upper bound and prune unnecessary nodes. In detail, we use the priority rule to select and fix columns of rolling stock units according to the descending order of the value of the column selection solution (i.e., λ_p) greedily. Then, for those rolling stock units without selected columns, we generate paths based on the LC algorithm by avoiding the marked space-time resources of fixed columns. Lastly, we select and fix columns of crew groups according to the descending order of the column selection solution (i.e., μ_p) in priority, and assign extra available crew members for residual uncovered trip tasks. For clarity, the detailed pseudocode of the PHR is displayed in Algorithm 3 of the electronic companion EC. D.

5.6.6 Parallel B&B computing (PC) technique

The parallel B&B computing technique can be implemented to improve computational efficiency by collaboratively exploring multiple-core CPU computation resources. Specifically, in the breadth first search (BFS) implementation, the nodes on one branching layer will be first allocated averagely across multiple processors in accordance with the number of CPU threads. After solving these nodes independently across different CPU threads in a parallel manner, the BFS lower bound is updated with the minimum objective value of involved nodes on the same layer. Obviously, the parallel B&B computing technique can perform faster lower and upper bound updating processes than the serial B&B process. The detailed pseudocode of the parallel B&B algorithm is displayed in Algorithm 4 of the electronic companion EC. E.

6 Numerical experiments

In this section, we conduct a series of numerical experiments, including small-scale and real-world case studies, to assess the effectiveness of the proposed methods. All numerical experiments are implemented

in C++ and CPLEX 20.10 on a computer with an Intel(R) Core(R) i7-13700 processor running at 2.10GHz CPU and 32GB of RAM. For clarity, the small-scale instances are solved using both the CPLEX solver and the proposed BPC algorithm to validate the correctness and the performance of our method. The large-scale instances are only solved using the proposed BPC algorithm since it is difficult to solve the primal model M6 (see electronic companion EC. A for the details of this baseline model) by using the commercial solvers directly. The instances and detailed computational results related to this research are available on the GitHub website https://github.com/EntaiWang99/BPC_JRCSP.

6.1 Small-scale case study

In this experiment, we consider a small-scale “Y-shaped” urban rail network with four operating stations S_1 – S_4 and three depots D_1 – D_3 connecting with stations S_1 , S_3 and S_4 respectively, as shown in Fig. 6. For clarity, the traveling times with respect to the physical segments are also labeled in Fig. 6(a). Then, a total of 2 to 8 rolling stock units are stored at the three depots in different instances, and three types of train routes in Fig. 6(b) are considered for scheduling rolling stock units from different depots. All rolling stock units need to leave and enter the same home depots at the beginning and the end of the planning horizon. Lastly, to ensure the consistency and feasibility of the trip frequency requirement constraints (4) under different numbers of rolling stock units (especially in some instances with a small number of rolling stock units), at least one trip is required to pass each physical segment within the total planning horizon. The maximum number of coupled rolling stock units is set to 2 (i.e., $N_a^{\max} = 2, \forall a \in \mathcal{A}$) for all physical segments.

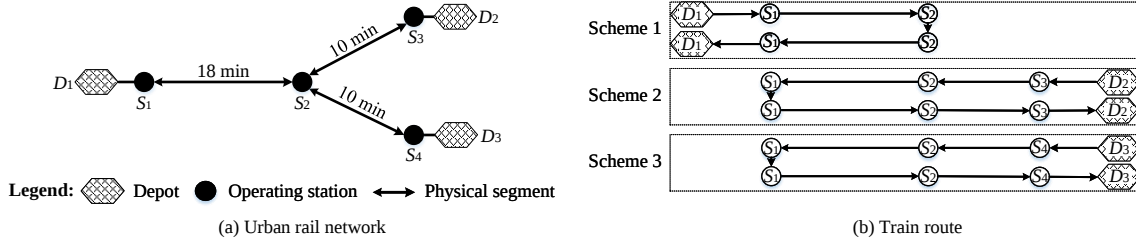


Figure 6: A small “Y-shape” bi-directional urban rail network with associated train routes

In the small-scale instances, the planning horizon is set to be the time interval $[0 \text{ min}, 60 \text{ min}]$. Regarding the input data of crew groups and crew members, we consider only one crew group, which is all available during the planning horizon (i.e., $[0 \text{ min}, 60 \text{ min}]$). All crew members can change their trips flexibly on different rolling stock units. Because we assume that the number of crew members in one crew group g (i.e., $|C_g|$) is not binding (see Assumption 4), for the convenience of implementing the mathematical model with the CPLEX solver, we set the maximum number of crew members equal to the number of trips to ensure the feasibility of the model (i.e., one trip is assigned to one individual crew member in the worst case). Lastly, the fixed operational costs for one rolling stock unit and one crew member are set to ¥100/unit and ¥404/member, and the variable costs for scheduling trips and connection operations are set to ¥35/min and ¥70/min, respectively. Based on the input data for rolling stock units and crew members, the corresponding path-based model M1 and the primal model M6 can be formulated.

6.1.1 Experimental results

In this section, we conduct experiments under various settings to evaluate the performance of our proposed methods. To this end, we consider a set of 7 instances and a total of 70 experiments in different settings, which are solved using both the CPLEX solver and the proposed BPC algorithm to validate the correctness and performance of our method. For convenience of description, we here introduce a notation with the format $(|\mathcal{K}|, |\mathcal{L}|, |\mathcal{R}|)$ to denote the experiment features, in which “ $|\mathcal{K}|$ ”

represents the number of rolling stock units, “ $|\mathcal{L}|$ ” represents the number of train routes, and “ $|\mathcal{R}|$ ” represents the number of trips. Based on some preliminary experiments, we observed that both upper and lower bounds by using the CPLEX solver or the BPC algorithm tend to be stable with only tiny improvements (or even no improvements) after 3600s of computing time. Thus, to balance the solution quality and the computing time, we use both the CPLEX solver and our BPC algorithm to solve these instances with a maximum computing time of 60 min (3600s).

The computational results are listed in Table 3 for all different instances, including acceleration techniques, lower bounds, upper bounds, gaps, and computing times. The BPC with all the acceleration techniques clearly outperforms the primal model M6 solved by CPLEX, and it could determine optimal or near-optimal solutions for all the instances. With the same termination condition, the average lower bound and computing time by using the BPC algorithm with all acceleration techniques (i.e., 10,972, 1278.6s in column “BPC algorithm with all techniques” of Table 3) are improved by 13.10% and 38.78%, respectively, compared to those of the primal model M6 solved by the CPLEX solver (i.e., 9701, 2088.7s in column “CPLEX solver” of Table 3).

6.1.2 Performance analyses regarding acceleration techniques

To further investigate the computing performance of our proposed models and algorithms, we implement numerical experiments with the absence of all or one acceleration technique(s), as shown in Table 3 with the computational results. Overall, the average lower bound and computing time by using the BPC algorithm with all acceleration techniques (i.e., 10,972, 1278.6s, see column “BPC algorithm with all techniques”) is improved by 0.82% and 50.40%, respectively, compared to the basic version (i.e., 10,883, 2577.6s, see column “BPC algorithm”). Particularly, when the computing time reaches 3600s, both the CPLEX solver and the standard BPC algorithm cannot find feasible upper bound solutions without the help of acceleration techniques for instances 6 and 7. We also observe that each acceleration technique can yield decreases in relative gaps and computing time by 53%–67% and 9%–46%, respectively. Lastly, we observe that some performances with respect to the individual acceleration techniques are similar in seven instances of Table 3. We present the convergence of instance 3 of Table 3 in Fig. (7) as an example to illustrate the improvement based on the Benders decomposition process at the root node and the B&B process. Specifically, we observe that both LBL and POC mainly accelerate the Benders decomposition process by improving lower bounds with tighter Benders cuts in the initial iterations. While the DCG, DMC, CCI, PHR, and PC techniques mainly accelerate the B&B process by generating constraints dynamically, excluding unnecessary columns, inheriting columns, generating upper bounds, and parallel node computation, respectively.

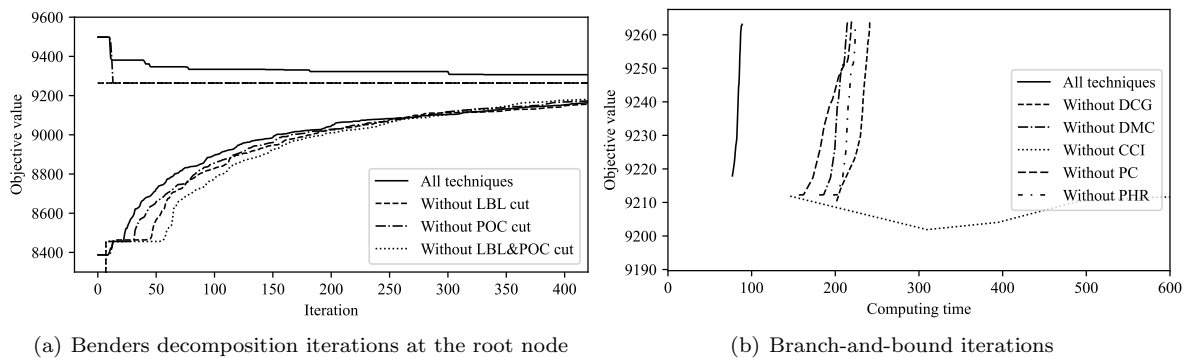


Figure 7: Illustrations of the convergence of instance 3 (as an example)

Table 3: Optimization results with different acceleration techniques in small-scale instances

ID	($\mathcal{K}$, $\mathcal{L}$, $\mathcal{R}$)	CPLEX solver				BPC with all techniques				BPC algorithm				Without PC				Without DCG			
		LB	UB	Gap (%)	CT (s)	LB	UB	Gap (%)	CT (s)	LB	UB	Gap (%)	CT (s)	LB	UB	Gap (%)	CT (s)	LB	UB	Gap (%)	CT (s)
1	(2,2,8)	5736	5736	0.0	0.6	5736	5736	0.0	0.2	5736	5736	0.0	0.1	5736	5736	0.0	0.2	5736	5736	0.0	0.3
2	(3,4,14)	7500	7500	0.0	3.6	7500	7500	0.0	3.7	7500	7500	0.0	43.1	7500	7500	0.0	15.2	7500	7500	0.0	12.8
3	(4,6,20)	9264	9264	0.0	216.4	9264	9264	0.0	88.7	9208	9264	0.6	3600.0	9264	9264	0.0	219.5	9264	9264	0.0	241.1
4	(5,8,26)	10220	11028	7.3	3600.0	11028	11028	0.0	299.9	10961	11028	0.6	3600.0	11028	11028	0.0	929.8	11028	11028	0.0	891.0
5	(6,10,32)	10368	12792	18.9	3600.0	12792	12792	0.0	1357.8	12741	-	-	3600.0	12777	12792	0.1	3600.0	12792	12792	0.0	3315.8
6	(7,12,38)	11728	-	-	3600.0	14529	14556	0.2	3600.0	14291	-	-	3600.0	14269	14980	4.7	3600.0	14185	14980	5.3	3600.0
7	(8,14,44)	13088	-	-	3600.0	15955	16754	4.8	3600.0	15746	-	-	3600.0	15761	16754	5.9	3600.0	15648	16764	6.7	3600.0
Ave.		9701	-	-	2088.7	10972	11090	0.7	1278.6	10883	-	-	2577.6	10905	11151	1.5	1709.2	10879	11152	1.7	1665.9
ID	($\mathcal{K}$, $\mathcal{L}$, $\mathcal{R}$)	Without DMC				Without CCI				Without PHR				Without LBL				Without POC			
		LB	UB	Gap (%)	CT (s)	LB	UB	Gap (%)	CT (s)	LB	UB	Gap (%)	CT (s)	LB	UB	Gap (%)	CT (s)	LB	UB	Gap (%)	CT (s)
1	(2,2,8)	5736	5736	0.0	0.2	5736	5736	0.0	0.2	5736	5736	0.0	0.1	5736	5736	0.0	0.2	5736	5736	0.0	0.2
2	(3,4,14)	7500	7500	0.0	12.5	7500	7500	0.0	113.5	7500	7500	0.0	16.2	7500	7500	0.0	29.9	7500	7500	0.0	19.5
3	(4,6,20)	9264	9264	0.0	214.7	9264	9264	0.0	2073.9	9264	9264	0.0	224.0	9264	9264	0.0	249.0	9264	9264	0.0	123.9
4	(5,8,26)	11028	11028	0.0	842.1	10947	11028	0.7	3600.0	11028	11028	0.0	807.9	11028	11028	0.0	826.5	11028	11028	0.0	564.5
5	(6,10,32)	12792	12792	0.0	3114.9	12690	13206	3.9	3600.0	12792	12792	0.0	3417.3	12792	12792	0.0	3154.1	12792	12792	0.0	1924.2
6	(7,12,38)	14268	15000	4.9	3600.0	14265	14990	4.8	3600.0	14267	-	-	3600.0	14255	14980	4.8	3600.0	14323	14980	4.4	3600.0
7	(8,14,44)	15772	16754	5.9	3600.0	15881	16754	5.2	3600.0	15778	-	-	3600.0	15782	16764	5.9	3600.0	15799	16764	5.8	3600.0
Ave.		10909	11153	1.5	1626.3	10898	11211	2.1	2369.7	10909	-	-	1666.5	10908	11152	1.5	1637.1	10920	11152	1.5	1404.6

(1) ID: Instance index; LB: Lower bound; UB: Upper bound; Gap = $\frac{UB-LB}{UB} \times 100\%$; CT: Computing time.

(2) Acceleration technique: Parallel computing (PC); Dynamic constraint generation (DCG); Dynamic management of column pool (DMC); Column and cut initialization (CCI), Priority heuristic rule (PHR), lower-bound lifting valid inequalities (LBL); Pareto-optimal cuts (POC).

(3) Solution status: Unfounded feasible solution with infinity upper bound and relative gap ("-").

(4) Ave.: Average performance for instances without infinity upper bounds and gaps.

6.2 Real-world case study

We consider a real-world case based on the Beijing urban rail system, including four bi-directional physical lines (i.e., lines 1, 2, 4, and 5) with 76 stations, to further explore the performance of the proposed methods and provide managerial insights for urban rail operators. Details of the urban rail network illustration are shown in Fig. 8 with detailed physical segment and station layouts. We assume that there are seven depots in this network, in which turn-around, coupling, and decoupling operations are allowed at operating stations (including terminal stations associated with depots and transfer stations).

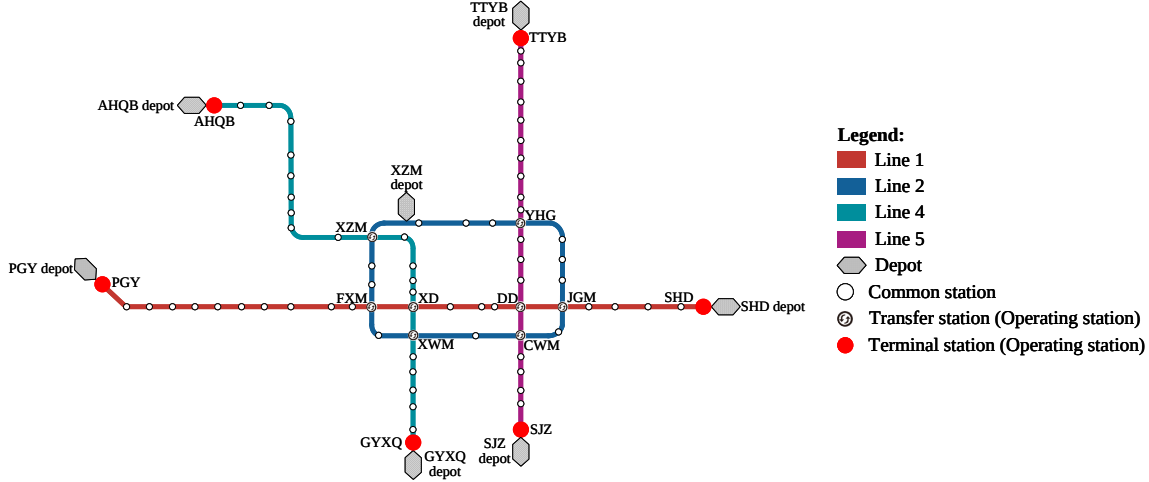


Figure 8: Overview of the Beijing urban rail network

6.2.1 Experimental settings

In this set of experiments, we consider a 16h planning horizon with the time interval [7:00, 23:00] and a total of 43 rolling stock units stored in seven depots. All rolling stock units need to leave and enter the same home depots at the beginning and the end of the planning horizon. Specifically, considering that the suburb depots (i.e., PGY, SHD, AHQB, GYXQ, TTYB, and SJZ) have larger spaces to store more rolling stock units with lower costs than the downtown depot XZM, it is possible for rolling stock units from lines 1, 4, and 5 to perform the cross-line operations, for satisfying the high trip frequency requirements of downtown line 2. Thus, as shown in Table 4, for rolling stock units at the suburb depots PGY, SHD, AHQB, GYXQ, TTYB, and SJZ associated with physical lines 1, 4, and 5, two types of candidate train routes are constructed for each rolling stock unit, including one local train route with only traveling on their local lines, and one cross-line train route with crossing line 2. Then, one local route is constructed for the rolling stock units at the downtown depot XZM with only traveling on line 2. Particularly, considering that one train route includes multiple loops of passing operating stations within the planning horizon, for clarity in column “TR sequence” of Table 4, for example, notation “ $\dots \rightarrow \text{Line 1} \rightarrow \dots$ ” represents that the train route passes stations on physical line 1 continuously without crossing to other physical lines. Next, based on the generated train routes for rolling stock units, a total of 685 trips can be generated by splitting the train routes at operating stations. Lastly, the maximum number of coupled rolling stock units is set to 2 (i.e., $N_a^{\max} = 2, \forall a \in \mathcal{A}$) in a train composition. The variable cost for scheduling trips and trip connection operations are set to ¥35/min and ¥70/min, respectively.

Regarding the trip frequency, we set the parameter $N_s^{\min}$ based on the average passenger demand on the involved segments in the Beijing urban rail system. Specifically, at least 8 min trip frequency time (i.e., 105 trips in one direction) on average should be guaranteed for busy downtown line 2, and

Table 4: Parameters for the rolling stock operations

ID	Depot	TR type	TR sequence	TR cost
1	PGY	Local line 1	PGY $\rightarrow$... $\rightarrow$ Line 1 $\rightarrow$... $\rightarrow$ PGY	500
2		Cross line 1	PGY $\rightarrow$ FXM $\rightarrow$ XZM $\rightarrow$... $\rightarrow$ Line 2 $\rightarrow$... $\rightarrow$ XWM $\rightarrow$ FXM $\rightarrow$ PGY	500
3	SHD	Local line 1	SHD $\rightarrow$... $\rightarrow$ Line 1 $\rightarrow$... $\rightarrow$ SHD	500
4		Cross line 1	SHD $\rightarrow$ JGM $\rightarrow$ YHG $\rightarrow$... $\rightarrow$ Line 2 $\rightarrow$... $\rightarrow$ CWM $\rightarrow$ JGM $\rightarrow$ SHD	500
5	XZM	Local line 2	XZM $\rightarrow$ FXM $\rightarrow$... $\rightarrow$ Line 2 $\rightarrow$... $\rightarrow$ YHG $\rightarrow$ XZM	1000
6		Local line 2	XZM $\rightarrow$ YHG $\rightarrow$... $\rightarrow$ Line 2 $\rightarrow$... $\rightarrow$ FXM $\rightarrow$ XZM	1000
7	AHQB	Local line 4	AHQB $\rightarrow$... $\rightarrow$ Line 4 $\rightarrow$... $\rightarrow$ AHQB	500
8		Cross line 4	AHQB $\rightarrow$ XZM $\rightarrow$ YHG $\rightarrow$... $\rightarrow$ Line 2 $\rightarrow$... $\rightarrow$ FXM $\rightarrow$ XZM $\rightarrow$ AHQB	500
9	GYXQ	Local line 4	GYXQ $\rightarrow$... $\rightarrow$ Line 4 $\rightarrow$... $\rightarrow$ GYXQ	500
10		Cross line 4	GYXQ $\rightarrow$ XWM $\rightarrow$ CWM $\rightarrow$... $\rightarrow$ Line 2 $\rightarrow$... $\rightarrow$ FXM $\rightarrow$ XWM $\rightarrow$ GYXQ	500
11	TTYB	Local line 5	TTYB $\rightarrow$... $\rightarrow$ Line 5 $\rightarrow$... $\rightarrow$ TTYB	500
12		Cross line 5	TTYB $\rightarrow$ YHG $\rightarrow$ JGM $\rightarrow$... $\rightarrow$ Line 2 $\rightarrow$... $\rightarrow$ XZM $\rightarrow$ YHG $\rightarrow$ TTYB	500
13	SJZ	Local line 5	SJZ $\rightarrow$... $\rightarrow$ Line 5 $\rightarrow$... $\rightarrow$ SJZ	500
14		Cross line 5	SJZ $\rightarrow$ CWM $\rightarrow$ JGM $\rightarrow$... $\rightarrow$ Line 2 $\rightarrow$... $\rightarrow$ XWM $\rightarrow$ CWM $\rightarrow$ SJZ	500

ID: Train route ID; Depot: Home depot of train route; TR type: Train route type; TR sequence: Passing operating station sequence of train route; TR cost: Fixed cost of train route.

at least 10 min trip frequency time (i.e., 84 trips in one direction) on average should be guaranteed for unbusy lines 1, 4, and 5 within time window [8:00, 22:00] by eliminating rolling stock warm-up time window [7:00, 8:00] and finalization time window [22:00, 23:00]. Lastly, as for the input data of crew groups and crew members, we consider two crew groups that are available during time windows [7:00, 15:00] and [15:00, 23:00], respectively. All crew members can change their trips flexibly on different rolling stock units and different physical segments. In addition, two meal/rest tasks are available to start within time windows [10:30, 13:00] and [17:30, 20:00]. The fixed operational costs for one crew member are set to ¥404/member.

6.2.2 Experimental results

In the experiment, to avoid the memory overflow with a large number of B&B nodes in latter B&B iterations, the termination criteria are set with a maximum of 10 branching layers (i.e., maximum of 1024 B&B nodes per branching layer) and a maximum computing time 24h (1440 min). We could determine a solution for this instance with a lower bound of 311,842, an upper bound objective value of 336,802, and a relative gap of 7.41%. For clarity, the optimized train schedule and the corresponding Gantt charts for rolling stock units and crew members are displayed in Figs. 9–11, respectively. Note that one trip with the bold line represents a long composition with two coupled rolling stock units. We observe that when higher trip frequencies are required on line 2 with larger passenger demand, 12 rolling stock units are scheduled to cross from lines 1, 4, and 5 to line 2, which could satisfy the higher trip frequency requirement of busy line 2. In addition, 38 trips are scheduled with long compositions to further improve the flexibility and reduce the cost of operations. Lastly, in the visualization of the crew schedule with Fig. 11, 180 crew members are assigned and scheduled to cover all in-service rolling stock units and trips. Particularly, some crew members (e.g., crew members 68–89) are scheduled to start within the meal/rest task time window (e.g., [10:30, 11:30] instead of the beginning of the planning horizon) due to the crew member resource shortage during the meal/rest period. Overall, most crew members can work 6-8 hours per day, which corresponds to the reality of crew operations. Typically, all crew members are scheduled with meal/rest tasks during time windows [10:30, 13:00] and [17:30, 20:00] within the trip task sequences throughout the day. Then, most crew members are scheduled to work consistently with other rolling stock units and trips after the meal/rest tasks.

6.2.3 Performance analyses in different experiment instances

To further investigate the performance of our proposed methods, we further consider 24 instances based on different parameter settings with the index form of $(|\mathcal{N}|, |\mathcal{K}|, |\mathcal{R}|)$, in which “ $|\mathcal{N}|$ ” represents the number of considered physical lines, “ $|\mathcal{K}|$ ” represents the number of rolling stock units, and “ $|\mathcal{R}|$ ”

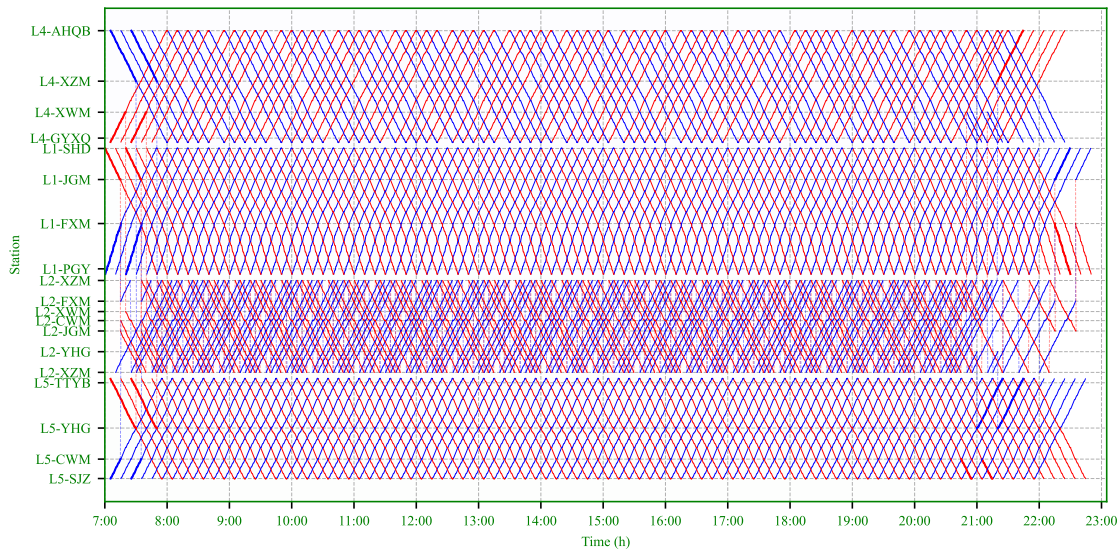


Figure 9: Visualization for rolling stock schedule in the Beijing urban rail network

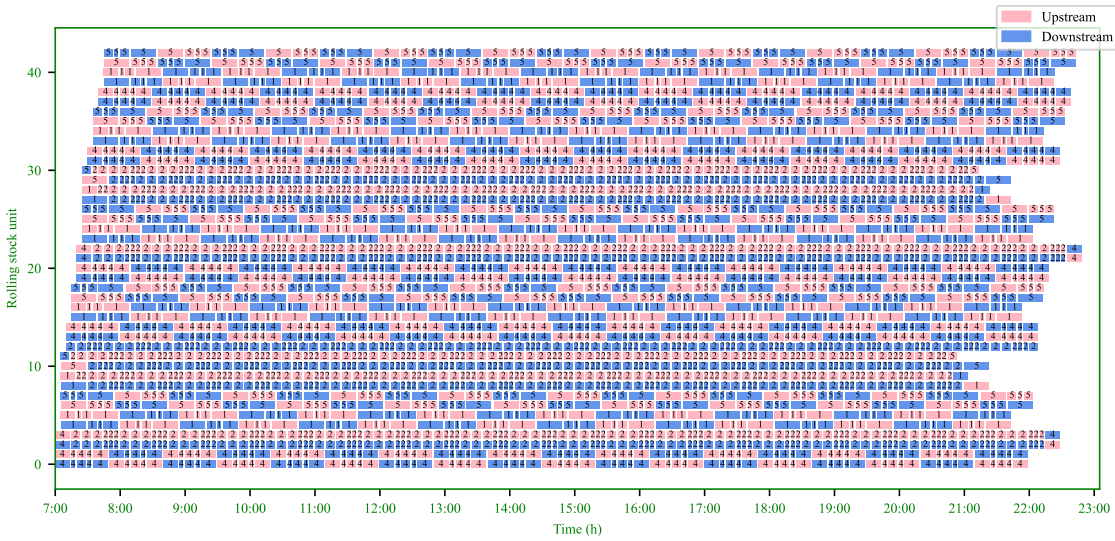


Figure 10: Visualization for rolling stock Gantt chart in the Beijing urban rail network

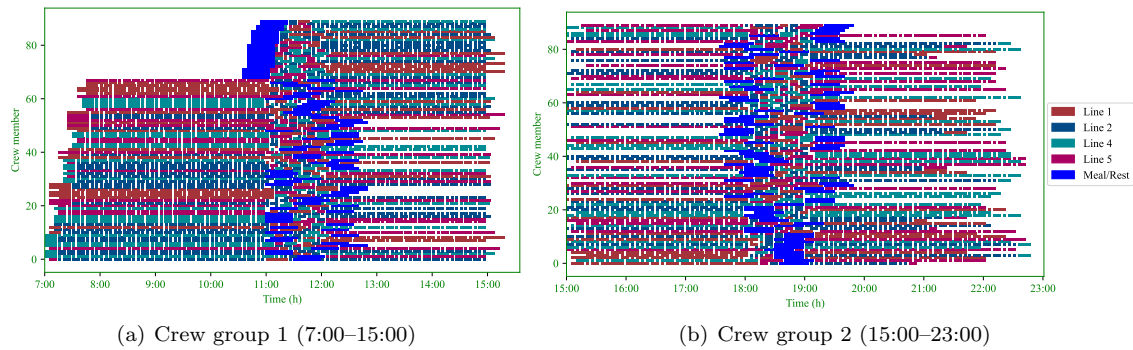


Figure 11: Visualization for crew Gantt chart in the Beijing urban rail network

represents the number of trips. Specifically, we perform 154 experiments with different features and acceleration technique combinations under a maximum of 10 branching layers and 24h (1440 min) of computing time as the algorithm termination criteria. To balance the distributions of rolling stock units in different physical lines, we start by allocating one rolling stock unit to one physical line for instance 1 with a 2-line network, instance 9 with a 3-line network, and instance 17 with a 4-line network (see column “ $(|\mathcal{N}|, |\mathcal{K}|, |\mathcal{R}|)$ ” of Table 5). Then, we increase one unit once for one physical line as the rate of increment to generate instances 2–8, 10–16, and 18–24, respectively. Lastly, to ensure the consistency and feasibility of the trip frequency requirement constraints (4) under different number of rolling stock units, at least one trip is required to pass each physical segment within the total planning horizon (as in Section 6.1).

Table 5: Computational results for different acceleration techniques in real-world instance

ID ($ \mathcal{N} , \mathcal{K} , \mathcal{R} $)	Only with PHR							All techniques						
	#UC	#CC	#C	LB	UB	Gap (%)	CT (min)	#UC	#CC	#C	LB	UB	Gap (%)	CT (min)
1 (2,2,39)	2396	64033	493	16215.0	16524	1.9	0.7	115	15947	50	16524.0	16524	0.0	0.1
2 (2,4,78)	12805	416721	1420	32255.9	34360	6.1	14.2	600	94729	64	32380.8	34310	5.6	0.7
3 (2,6,117)	37680	1318485	2651	48412.5	52500	7.8	102.6	1295	200082	114	48541.6	52146	6.9	4.1
4 (2,8,158)	99915	3828993	4957	64488.5	69378	7.0	443.2	1811	312582	112	64598.5	69074	6.5	15.0
5 (2,10,195)	77547	3389506	3112	80553.3	88376	8.6	1440.0	2974	425182	168	80682.1	88022	8.3	38.2
6 (2,12,233)	36701	1779257	1209	96640.1	106466	9.2	1440.0	4243	598752	266	96777.3	104446	7.3	85.5
7 (2,14,271)	23668	1177276	651	112760.0	125364	10.1	1440.0	5592	725194	282	112894.0	124706	9.5	199.9
8 (2,16,308)	17131	911518	438	128855.0	143150	10.0	1440.0	16406	2569778	504	129024.0	142038	9.2	456.0
9 (3,3,55)	6093	152026	858	23861.3	24142	1.2	2.8	235	34048	58	23911.0	24042	0.5	0.2
10 (3,6,107)	39118	1244528	2775	47643.6	50054	4.8	54.3	1599	218447	172	47766.3	49750	4.0	3.5
11 (3,9,159)	105937	4348369	5169	71335.6	76116	6.3	765.7	2371	387713	162	71536.3	75358	5.1	16.1
12 (3,12,213)	51358	2258036	1687	95064.8	103340	8.0	1440.0	3464	592938	218	95287.5	102582	7.1	60.1
13 (3,15,263)	40382	1955863	1081	118868.0	129756	8.4	1440.0	5291	697377	286	118990.0	129352	8.0	143.8
14 (3,18,317)	25821	1286206	570	142607.0	158092	9.8	1440.0	7988	1066267	452	142661.0	156626	8.9	711.5
15 (3,21,367)	14188	737429	270	166388.0	-	-	1440.0	10960	1475930	642	166480.0	181576	8.3	1037.2
16 (3,24,419)	8378	430137	149	188258.0	-	-	1440.0	15340	1657586	1186	189227.0	210520	10.1	1440.0
17 (4,4,70)	11820	333882	1282	31654.0	33316	5.0	8.9	682	96546	90	31755.2	32862	3.4	0.8
18 (4,8,137)	91628	3220352	4807	63216.7	67644	6.5	430.8	1980	333215	118	63367.6	67644	6.3	8.8
19 (4,12,206)	2574	3145960	2574	94688.3	101868	7.0	1440.0	3708	556508	218	94896.8	101414	6.4	52.9
20 (4,16,273)	41154	1906007	1014	126284.0	137358	8.1	1440.0	5845	913678	342	126313.0	135842	7.0	185.1
21 (4,20,340)	17424	904298	359	157873.0	173602	9.1	1440.0	8947	1272577	474	157902.0	171936	8.2	748.0
22 (4,24,406)	11067	590530	200	187949.0	207022	9.2	1440.0	17157	2127208	982	189495.0	206968	8.4	1440.0
23 (4,28,473)	9667	490230	152	217718.0	-	-	1440.0	32357	3169964	2058	218503.0	243420	10.2	1440.0
24 (4,32,537)	8360	402398	117	246330.7	-	-	1440.0	30491	2926492	1886	246336.0	275424	10.6	1440.0
Ave.	33033.8	1512168.3	1583.1	106663.3			976.0	7560.5	936197.5	454.3	106910.4	116524.3	6.9	397.0

(1) Acceleration techniques: Priority heuristic rule (PHR).

(2) Solution status: Unfounded feasible solution with infinity upper bound and relative gap (“-”).

(3) ID: Instance index; #UC: number of unit columns; #CC: number of crew columns; #C: number of cuts; LB: Lower bound; UB: Upper bound; Gap = $\frac{UB-LB}{UB} \times 100\%$; CT: Computing time; Ave.: Average.

(1) Performance analyses regarding acceleration techniques We observe that the performance improvements using the acceleration techniques in real-world case studies are similar to those in small-scale instances. In addition, we observed that feasible solutions could not be found in most of instances when using the standard BPC algorithm. Thus, it is necessary to apply the PHR acceleration technique to find feasible solutions. We present the performance results in Table 5 with respect to the PHR and other acceleration techniques. From the results, the average lower bound with all acceleration techniques (i.e., 106,910.4) can be improved by 0.23% compared to using only the PHR technique (i.e., 106,663.3), and the improvement is as high as 1.87% in some instances. In addition, the average number of rolling stock columns, the number of crew columns, the number of cuts, and the computing time with all acceleration techniques (i.e., 7560.5, 936,197.5, 454.3, 23,818.8) can be reduced by 77.11%, 38.09%, 71.30% and 51.32% compared to using only the PHR technique (i.e., 33,033.8, 1,512,168.3, 1583.1, 58,557.8), respectively.

(2) Performance analyses regarding sequential/joint rolling stock and crew scheduling schemes To demonstrate the benefits of the joint rolling stock scheduling and crew scheduling schemes, we define the sequential rolling stock and crew scheduling problem (SRCSP) with the mathematical model M5 in Eq. (23) for the comparisons with the JRCSP. Specifically, in this model, set Λ indicates the set of vectors for the rolling stock path selection variables that satisfy $0 \leq \lambda_p \leq 1$ and constraints (2)–(5), i.e., $\Lambda = \left\{ \lambda = (\lambda_1, \dots, \lambda_p, \dots, \lambda_{|\mathcal{P}_k^R|})^T \mid 0 \leq \lambda_p \leq 1, \text{ constraints (2)–(5), } p \in \mathcal{P}_k^R \right\}$, and set $\mathcal{M}(\lambda)$ indicates the set of vectors for the crew task sequence selection variables that satisfy $0 \leq \mu_p \leq 1$ and constraints (6) based on rolling stock path solution λ , i.e.,

$$\mathcal{M}(\lambda) = \left\{ \mu = (\mu_1, \dots, \mu_p, \dots, \mu_{|\mathcal{P}^C|})^T \mid 0 \leq \mu_p \leq 1, \text{ constraints (6), } p \in \mathcal{P}^C \right\}.$$

We first optimize the rolling stock scheduling model (i.e., $\min_{\lambda \in \Lambda} \left\{ \sum_{p \in \mathcal{P}^R} C_p^C \cdot \lambda_p \right\}$), and then optimize the crew scheduling model based on given rolling stock path selection variable $\bar{\lambda} \in \Lambda$ (i.e., $\min_{\mu \in \mathcal{M}(\bar{\lambda})} \left\{ \sum_{p \in \mathcal{P}^C} C_p^C \cdot \mu_p \right\}$). Both the rolling stock and crew scheduling models are tackled by the standard B&P algorithm, respectively.

$$\text{M5: } \min_{\mu \in \mathcal{M}(\lambda)} \left\{ \sum_{p \in \mathcal{P}^C} C_p^C \cdot \mu_p + \min_{\lambda \in \Lambda} \left\{ \sum_{p \in \mathcal{P}^R} C_p^C \cdot \lambda_p \right\} \right\} \quad (23)$$

To our knowledge, salary levels vary widely in different countries around the world. Based on the preliminary experiment, we observed that different salary levels for the crew members have a crucial impact on the balance of the rolling stock and crew scheduling solutions. Thus, we test the SRCSP and the JRCSP instances by considering two cases: case A with ¥404 per member per day and case B with ¥808 per member per day. Then, Table 6 displays the computational results with respect to the operational costs for rolling stock units and crew members by implementing sequential/joint scheduling schemes on different salary levels. From these results, the solutions of case A obtained by the joint

Table 6: Summary of optimization results in sequential/joint rolling stock and crew scheduling schemes

ID ($ \mathcal{N} , \mathcal{K} , \mathcal{R} $)	Case A						Case B					
	SRCSP			JRCSP			SRCSP			JRCSP		
	RSC	CC	TC	RSC	CC	TC	RSC	CC	TC	RSC	CC	TC
1 (2,2,39)	14100	2424	16524	14100	2424	16524	14100	4848	18948	14100	4848	18948
2 (2,4,78)	28200	6464	34664	28250	6060	34310	28200	12928	41128	28300	11312	39612
3 (2,6,117)	42300	10504	52804	42450	9696	52146	42300	21008	63308	42400	16968	59368
4 (2,8,158)	56400	15756	72156	56550	12524	69074	56400	31512	87912	56600	26664	83264
5 (2,10,195)	70500	18988	89488	70650	17372	88022	70500	37976	108476	70600	35552	106152
6 (2,12,233)	84600	22220	106820	84650	19796	104446	84600	44440	129040	84800	41208	126008
7 (2,14,271)	98700	26664	125364	98850	25856	124706	98700	53328	152028	99000	47672	146672
8 (3,3,55)	20760	6868	27628	20810	3232	24042	20760	13736	34496	20760	7272	28032
9 (3,6,107)	41520	8484	50004	41670	8080	49750	41520	16968	58488	41620	16160	57780
10 (3,9,159)	62280	15756	78036	62430	12928	75358	62280	31512	93792	62380	27472	89852
11 (3,12,213)	83040	19796	102836	83190	19392	102582	83040	39592	122632	83340	38784	122124
12 (3,15,263)	103800	26260	130060	103900	25452	129352	103800	52520	156320	104000	46056	150056
13 (3,18,317)	124560	35956	160516	124710	31916	156626	124560	71912	196472	125060	60600	185660
14 (3,21,367)	145320	39188	184508	145620	35956	181576	145320	78376	223696	145570	76760	222330
15 (4,4,70)	27560	8080	35640	27610	5252	32862	27560	16160	43720	27710	10504	38214
16 (4,8,137)	55120	15352	70472	55120	12524	67644	55120	30704	85824	55420	24240	79660
17 (4,12,206)	82680	23028	105708	82830	18584	101414	82680	46056	128736	82880	37976	120856
18 (4,16,273)	110240	27472	137712	110390	25452	135842	110240	54944	165184	110540	51712	162252
19 (4,20,340)	137800	38784	176584	138000	33936	171936	137800	77568	215368	138000	64640	202640
Ave.	73130.5	19370.7	92501.3	73251.6	17180.6	90432.2	73130.5	38741.5	111872.0	73320.0	34021.1	1107341.1

ID: Instance index; SRCSP: Sequential rolling stock and crew scheduling problem; JRCSP: Joint rolling stock and crew scheduling problem; RSC: Rolling stock cost; CC: Crew cost; TC: Total cost; Ave.: Average.

scheduling scheme have an average objective function value of 90,432.2, resulting in an average reduction of 2.24% as opposed to the solution obtained by the sequential scheduling scheme (the reduction is as high as 12.98% in some instances). The reduction mainly comes from the significant reduction in crew scheduling costs (11.31% on average) and the slight increase (0.17% on average) in rolling stock scheduling costs. In case B with higher salary, the solutions obtained by the joint scheduling scheme have an average objective function value of 107,341.1, resulting in an average reduction of 4.05% compared to the solution obtained by the sequential scheduling scheme (the reduction is as high as 18.74% in some instances). The reduction mainly comes from the significant reduction in crew scheduling costs (12.18% on average) and the slight increase (0.26% on average) in rolling stock scheduling costs between the joint and sequential schemes.

(3) Performance analyses regarding the flexible operation schemes Compared with the single train composition operation scheme on urban rail lines, the multi-train composition and cross-line operation (i.e., rolling stock units can travel flexibly by using connection tracks among different physical rail lines) schemes might improve the flexibility of scheduling rolling stock and crew members in rail networks. We next test 10 extra instances with 30 experiments based on different multi-train compositions and cross-line operation schemes. The computational results for different tested instances are displayed in Table 7, including the costs of rolling stock units, crew members, and total operations. Particularly, the column with “Basic operation” indicates that both the multi-train composition (MTC) and cross-line operation schemes are not involved; the column with “CL” indicates that only the cross-line operation scheme is involved; and the column with “CL+MTC” indicates that both the cross-line and MTC operation schemes are involved for scheduling rolling stock units.

From these results, with the cross-line operation scheme (see column “CL”), the average objective value (i.e., 110,782.4) decreased by 5.94% compared to the common scheme without both the cross-line and MTC operations (i.e., 117,772.8, see column “Basic operation”). This is mainly because some rolling stock units can be scheduled from the suburb depots with lower train route costs, and then service downtown trips by using cross-line train routes. Specifically, rolling stock units can cross to downtown line 2 from other suburb lines 1, 4, 5 with the cross-line operations, to satisfy the higher trip frequency requirement of line 2. In comparison, in the case without the cross-line operation scheme, some rolling stock units have to be allocated and depart from the downtown depot with higher train route costs. Furthermore, with both the MTC and cross-line operation schemes (see column “CL+MTC”), the average objective value (i.e., 104,849.7) decreases by 10.97% compared to the common scheme without both the cross-line and MTC operations (i.e., 117,772.8, see column “Basic operation”), since multiple rolling stock units can further share one trip cost in a long train composition on the basis of the cross-line operation. Thus, when the trip frequency requirements (i.e., passenger demand) vary in different urban rail lines, it is helpful to match passenger demand distributions with rolling stock units flexibly crossing multiple lines, so as to maximize the efficiency of rolling stock units and reduce the operation costs.

7 Conclusions and future research

In this study, with the aim of scheduling joint rolling stock and crew operation schemes in urban rail networks, we formulated the JRCSP into a path-based model with considering both multi-train composition and cross-line operations. To handle real-world instances, an exact branch-and-price-and-cut (BPC) solution algorithm was designed based on Benders decomposition and column generation methods. Then, the proposed BPC algorithm was unbraced through several acceleration techniques. Finally, we conducted the performance of the proposed methods based in small-scale instances and real-world instances. The proposed methodology can adequately schedule rolling stock units and crew members flexibly in rail networks, which might allows rail dispatchers to make consistent rolling stock and crew scheduling plans. The proposed approach and enhancements can be generalized and extended for solving other integrated scheduling problems in railway systems. In general, most rail

Table 7: Summary of optimization results in instances with different operation schemes

ID	$(\mathcal{N} , \mathcal{K})$	Basic operation			CL			CL+MTC		
		RSC	CC	TC	RSC	CC	TC	RSC	CC	TC
1	(2,4)	28250	6060	34310	26360	6060	32420	25834	5656	31490
2	(2,8)	56550	12524	69074	52720	12524	65244	50076	13332	63408
3	(2,12)	84650	19796	104446	79080	20604	99684	72674	21412	94086
4	(2,16)	112950	29088	142038	105440	25856	131296	92058	28280	120338
5	(3,6)	41670	8080	49750	39190	9696	48886	37524	7676	45200
6	(3,12)	83190	19392	102582	78240	17372	95612	74188	19796	93984
7	(3,18)	124710	31916	156626	117430	27472	144902	107551	30704	138255
8	(3,24)	166080	44440	210520	156480	38380	194860	138080	42824	180904
9	(4,12)	82830	18584	101414	79570	18180	97750	75763	19392	95155
10	(4,24)	165760	41208	206968	158790	38380	197170	144873	40804	185677
Ave.		94664.0	23108.8	117772.8	89330.0	21452.4	110782.4	81862.1	22987.6	104849.7

(1) ID: Instance index; RSC: Rolling stock cost; CC: Crew cost; TC: Total cost; Ave.: Average.

(2) Basic operation: Without the cross-line and multiple train composition operation schemes; CL: With only the cross-line operation scheme; CL+MTC: With both cross-line and multi-train composition operation schemes.

operational problems are solved separately and sequentially in different decision stages, such as line planning and train scheduling (Yao et al., 2023), train scheduling and routing (Wang et al., 2023). Thus, our proposed BPC solution algorithm can be applied to solving these integrated problems by decomposing them into BMP and BSP models and then employing the column generation method in the corresponding pricing sub-problems.

From the managerial perspective of the urban rail operations, the operational efficiency of scheduling rolling stock units and crew members can be improved in two ways: (1) Compared with the sequential rolling stock and crew scheduling scheme, the joint scheduling scheme benefits urban rail dispatchers by reducing the total operational costs for rolling stock units and crew members; (2) Compared to the operation scheme with the fixed train composition mode and without crossing physical rail lines, the multi-train-composition and cross-line operation schemes can improve the solution flexibility to satisfy passenger demand requirements with the limited rolling stock unit and crew member resources. Further research will focus on the following aspects: (1) A robust joint operation plan could be implemented to schedule rolling stock units and crew members based on spatio-temporal dynamic (e.g., OD-based time-dependent) and uncertain passenger demand. A promising research direction is to develop stochastic/robust optimization methods to tackle such a JRCSPP based on the uncertain passenger demand over multiple scenarios; (2) The proposed models and algorithms from this paper can be extended to solve long-term JRSCP (e.g., one week/month) that incorporates rolling stock maintenance and crew rostering (Feng et al., 2024).

Electronic companion

A Equivalent primal model of the JRCSPP

In this section, we present a compact mathematical optimization model of the JRCSPP for the validation and the comparison with the proposed approach, which is equivalent to the path-based model M1 in Section 4 and can be solved by a general-purpose solver (e.g., CPLEX) directly. Specifically, we introduce the notation, objective function, and system constraints in Sections A.1 and A.2, respectively.

A.1 Notation

For convenience, the additional notations used in this primal model M6 is summarized in Table 8 for reference. Specifically, the following binary decision variables are used to model the problem: train route selection variable $u_{k,l}$ taking value 1 if train route l is selected by rolling stock unit k in the rail network (otherwise $u_{k,l} = 0$), trip departure time selection variable $x_{r,t}$ taking value 1 if trip r starts at timestamp t in the train schedule (otherwise $x_{r,t} = 0$), crew task starting variable $z_{c,0,r}$ taking value 1 if trip/task r is connected from the virtual source for crew member c (otherwise $z_{c,0,r} = 0$), crew task connection variable $z_{c,r,r'}$ taking value 1 if trip/task r is connected with trip/task r' for crew member c (otherwise $z_{c,r,r'} = 0$), and crew task ending variable $z_{c,r,0}$ taking value 1 if trip/task r is connected to the virtual sink for crew member c (otherwise $z_{c,r,0} = 0$). In addition, we define the following positive integer variables: trip connection time $v_{k,\sigma(r),r}$ indicating the connection time between trips $\sigma(r)$ and r for rolling stock k , and total crew number m_g indicating the number of assigned crew members for each crew group g .

Table 8: Additional notations in the primal model

Notation	Definition
Indices	
c	Index of crew members
Sets	
$\mathcal{R}_r^{T+}, \mathcal{R}_r^{T-}$	Set of possible adjacent next/previous trips of trip r
Parameters	
$T_{\sigma(r),r}^{R\min}, T_{\sigma(r),r}^{R\max}$	Minimum/Maximum connection time between trips $\sigma(r)$ and r for rolling stock units
$T_{r,r'}^{C\min}, T_{r,r'}^{C\max}$	Minimum/Maximum connection time between trips r and r' for crew members
Decision variables	
$u_{k,l}$	Binary variables, =1 if train route l is selected by rolling stock unit k ; =0 otherwise
$x_{r,t}$	Binary variables, =1 if trip r starts at timestamp t ; =0 otherwise
$v_{k,\sigma(r),r}$	Integer variables, representing connection time between trips $\sigma(r)$ and r by rolling stock unit k ; =0 otherwise
$z_{c,0,r}$	Binary variables, =1 if trip r is assigned as the first task from the virtual source by crew member c ; =0 otherwise
$z_{c,r,r'}$	Binary variables, =1 if trips r and r' are connected by crew member c ; =0 otherwise
$z_{c,r,0}$	Binary variables, =1 if trip r is assigned as the last task to the virtual sink by crew member c ; =0 otherwise
m_g	Integer variables, representing total number of used crew members in crew group g

A.2 Objective function and system constraints

We first formulate the objective function in Eq. (24), which aims to minimize the total cost of the rolling stock and crew scheduling plans and is equivalent to Eq. (1) in the path-based model M1. Specifically, the first term represents the total fixed cost of the selected train routes for rolling stock units. The second term represents the fixed cost for the trips. The third term represents the total cost of the trip connection operations for rolling stock units. The last term represents the total cost of assigning crew members.

$$\min \sum_{k \in \mathcal{K}} \sum_{l \in \mathcal{L}_k} c_{k,l}^1 u_{k,l} + \sum_{k \in \mathcal{K}} \sum_{l \in \mathcal{L}_k} \sum_{r \in \mathcal{R}_l^T} \sum_{t \in \mathcal{T}_r} c_{r,t}^2 x_{r,t} + \sum_{k \in \mathcal{K}} \sum_{l \in \mathcal{L}_k} \sum_{r \in \mathcal{R}_l^T \setminus \{r_0(l)\}} c_{\sigma(r),r}^3 v_{k,\sigma(r),r} + \sum_{g \in \mathcal{G}} c_g^4 m_g \quad (24)$$

The feasibility and consistency of the model are guaranteed by three groups of system constraints formulated below, where the rolling stock scheduling related constraints, crew scheduling related constraints, and the range of the decision variables are expressed in Groups I, II, and III, respectively. For clarity, we first discuss the Group I system constraints in the following.

Group I: Rolling stock scheduling constraints

$$\sum_{l \in \mathcal{L}_k} u_{k,l} = 1 \quad \forall k \in \mathcal{K} \quad (25)$$

$$\sum_{t \in \mathcal{T}_r} x_{r,t} = u_{k,l} \quad \forall r \in \mathcal{R}_l^T, l \in \mathcal{L}_k, k \in \mathcal{K} \quad (26)$$

$$\sum_{t' \in \mathcal{T}_r \cap \{t\}} x_{r,t'} + \sum_{t' \in \mathcal{T}_{r'} \cap \mathbb{Z} \left[t+1, \min \left\{ t+h_{r,r'}^{\min}-1, |\mathcal{T}| \right\} \right]} x_{r',t'} \leq 1 \quad \forall t \in \mathcal{T}, (r, r') \in \mathcal{V} \quad (27)$$

$$\sum_{r \in \mathcal{R}^T} \sum_{t \in \mathcal{T}_s \cap \mathcal{T}_r} x_{r,t} \cdot \delta_{r,a} \geq N_s^{\min} \quad \forall s \in \mathcal{S}_a, a \in \mathcal{A} \quad (28)$$

$$\sum_{r \in \mathcal{R}^T} x_{r,t} \cdot \delta_{r,a} \leq N_a^{\max} \quad \forall a \in \mathcal{A}, t \in \mathcal{T} \quad (29)$$

Constraints (25) require one suitable train route l to be selected from the corresponding train route set $\mathcal{L}_k$ for scheduling each rolling stock unit k . Then, constraints (26) ensure that all trips that belong to train route l (i.e., $r \in \mathcal{R}_l^T$) start at one timestamp within their given time window (i.e., $\mathcal{T}_r$) if train route l is selected by rolling stock unit k (i.e., $u_{k,l} = 1$). Constraints (27) are the safety headway constraints among trips, which are equivalent to constraints (3). Constraints (28) impose the trip frequency constraints, which are equivalent to constraints (4). Particularly, we define parameter $\delta_{r,a}$, which is equal to 1 if trip r passes through physical segment a . Constraints (29) are the covering constraints of rolling stock units for physical segment a at timestamp t , which is equivalent to constraints (5).

$$\sum_{t \in \mathcal{T}_r} t \cdot x_{r,t} \geq \sum_{t \in \mathcal{T}_{\sigma(r)}} t \cdot x_{\sigma(r),t} + \left(T_{\sigma(r)}^R + T_{\sigma(r),r}^{\min} \right) \cdot u_{k,l} \quad \forall r \in \mathcal{R}_l^T \setminus \{r_0(l)\}, l \in \mathcal{L}_k, k \in \mathcal{K} \quad (30)$$

$$\sum_{t \in \mathcal{T}_r} t \cdot x_{r,t} \leq \sum_{t \in \mathcal{T}_{\sigma(r)}} t \cdot x_{\sigma(r),t} + \left(T_{\sigma(r)}^R + T_{\sigma(r),r}^{\max} \right) \cdot u_{k,l} \quad \forall r \in \mathcal{R}_l^T \setminus \{r_0(l)\}, l \in \mathcal{L}_k, k \in \mathcal{K} \quad (31)$$

$$v_{k,\sigma(r),r} = \sum_{t \in \mathcal{T}_r} t \cdot x_{r,t} - \sum_{t \in \mathcal{T}_{\sigma(r)}} t \cdot x_{\sigma(r),t} - \left(T_{\sigma(r)}^R + T_{\sigma(r),r}^{\max} \right) \cdot u_{k,l} \quad \forall r \in \mathcal{R}_l^T \setminus \{r_0(l)\}, l \in \mathcal{L}_k, k \in \mathcal{K} \quad (32)$$

Constraints (30)–(31) ensure that one rolling stock unit connects trip r (except the first trip $r_0(l)$) with train route l and its adjacent previous trip $\sigma(r)$ with the given minimum connection time $T_{\sigma(r),r}^{\min}$ and maximum connection time $T_{\sigma(r),r}^{\max}$, respectively, if rolling stock unit k selects train route l (i.e., $u_{k,l} = 1$). Then, we calculate the actual connection time between two adjacent trips $\sigma(r)$ and r for rolling stock unit k in constraints (32).

With constraints (25)–(32), we can optimize the rolling stock scheduling problem in the rail network. At the same time, the feasibility and consistency constraints for scheduling crew members are also needed to generate joint optimization solutions, which will be discussed in the following group.

Group II: Crew scheduling constraints

$$\sum_{r \in \mathcal{R}^T \cup \mathcal{R}_c^M} z_{c,0,r} \leq 1 \quad \forall c \in \mathcal{C} \quad (33)$$

In constraints (33), each crew member c has at most one starting task in the planning horizon, including a trip in set $\mathcal{R}^T$, or a meal task in set $\mathcal{R}_c^M$ associated with crew member c .

$$z_{c,0,r} + \sum_{r' \in \mathcal{R}_r^{T-} \cup \mathcal{R}_c^M \cup \mathcal{R}_c^R} z_{c,r',r} = z_{c,r,0} + \sum_{r' \in \mathcal{R}_r^{T+} \cup \mathcal{R}_c^M \cup \mathcal{R}_c^R} z_{c,r,r'} \quad \forall r \in \mathcal{R}^T \cup \mathcal{R}_c^M \cup \mathcal{R}_c^R, c \in \mathcal{C} \quad (34)$$

$$z_{c,0,r} + \sum_{r' \in \mathcal{R}_r^{T-}} z_{c,r',r} = z_{c,r,0} + \sum_{r' \in \mathcal{R}_r^{T+}} z_{c,r,r'} \quad \forall r \in \mathcal{R}_c^M \cup \mathcal{R}_c^R, c \in \mathcal{C} \quad (35)$$

$$\sum_{r \in \mathcal{R}^T \cup \mathcal{R}_c^M} z_{c,0,r} = \sum_{r \in \mathcal{R}^T \cup \mathcal{R}_c^M} z_{c,r,0} \quad \forall c \in \mathcal{C} \quad (36)$$

Constraints (34)–(36) refer to task flow balance constraints for each crew member c . Two sets, i.e., $\mathcal{R}_r^{T+}$ and $\mathcal{R}_r^{T-}$, denote the sets of possible next and previous trips for trip r . Particularly, one trip

(i.e., $r \in \mathcal{R}^T$) can be connected with two trips, meal tasks, or rest tasks as its next/previous adjacent tasks (see constraints (34)). While one meal/rest task (i.e., $r \in \mathcal{R}_c^M \cup \mathcal{R}_c^R$ for crew member c) can only be connected with trips as its next/previous adjacent tasks (see constraints (35)), as two consecutive meal/rest tasks are not permitted in the JRCSP as well as real-world situations. Lastly, based on constraints (33) and (36), the crew members are only allowed to start or end with trip $r \in \mathcal{R}^T$, or meal task $r \in \mathcal{R}^M$, but not allowed to start or end with rest task $r \in \mathcal{R}^R$, which are consistent with the real-world labor requirements.

$$\sum_{t \in \mathcal{T}_{r'}} tx_{r',t} \geq \sum_{t \in \mathcal{T}_r} tx_{r,t} + T_r^R + T_{r,r'}^{C \min} + M_{r,r'}(z_{c,r,r'} - 1) \quad \forall r, r' \in \mathcal{R}^T \cup \mathcal{R}_c^M \cup \mathcal{R}_c^R, r \neq r', c \in \mathcal{C} \quad (37)$$

$$\sum_{t \in \mathcal{T}_{r'}} tx_{r',t} \leq \sum_{t \in \mathcal{T}_r} tx_{r,t} + T_r^R + T_{r,r'}^{C \max} + M_{r,r'}(1 - z_{c,r,r'}) \quad \forall r, r' \in \mathcal{R}^T \cup \mathcal{R}_c^M \cup \mathcal{R}_c^R, r \neq r', c \in \mathcal{C} \quad (38)$$

Constraints (37)–(38) refer to the minimum connection time (i.e., $T_{r,r'}^{C \min}$) and maximum connection time (i.e., $T_{r,r'}^{C \max}$) requirements, if two consecutive tasks r, r' are performed by one crew member c (i.e., $z_{c,r,r'} = 1$).

$$M_{r,r'} = \begin{cases} \max \{0, \bar{T}_{r'} - \underline{T}_r + T_{r,r'}^{C \max} + T_r^R\}, & \text{if } r' \in \mathcal{R}^T \\ \bar{T}_{r'} + T_{r,r'}^{C \max} + T_r^R, & \text{if } r' \in \mathcal{R}^M \cup \mathcal{R}^R \end{cases} \quad \forall r, r' \in \mathcal{R}^T \cup \mathcal{R}^M \cup \mathcal{R}^R, r \neq r' \quad (39)$$

Moreover, to tighten the ILP model of the JRCSP, constraints (37)–(38) should be set with smallest feasible big- M values. To this end, let $M_{r,r'}$ be the big- M value associated with trips/tasks r and r' in Eq. (39), where $\underline{T}_r$ represent the earliest start time of trip/task r and $\bar{T}_{r'}$ represent the latest start time of trip/task r' .

$$z_{c,0,r} + \sum_{r' \in \mathcal{R}_r^{T-}} z_{c,r',r} = \sum_{r' \in \mathcal{R}^T \cup \mathcal{R}_c^M} z_{c,0,r'} \quad \forall r \in \mathcal{R}_c^M \cup \mathcal{R}_c^R, c \in \mathcal{C} \quad (40)$$

$$\sum_{t \in \mathcal{T}_r} x_{r,t} = \sum_{r' \in \mathcal{R}^T \cup \mathcal{R}_c^M} z_{c,0,r'} \quad \forall r \in \mathcal{R}_c^M \cup \mathcal{R}_c^R, c \in \mathcal{C} \quad (41)$$

Constraints (40)–(41) ensure that each meal/rest task $r \in \mathcal{R}_c^M \cup \mathcal{R}_c^R$ is planned for a crew member c that works in the planning horizon (i.e., $\sum_{r' \in \mathcal{R}^T \cup \mathcal{R}_c^M} z_{c,0,r'} = 1$), and it should start with the given time window $\mathcal{T}_r$ (e.g., lunch/dinner time windows).

$$x_{r,t} \leq \sum_{c \in \mathcal{C}} z_{c,0,r} + \sum_{c \in \mathcal{C}} \sum_{r' \in \mathcal{R}_r^{T-} \cup \mathcal{R}_c^M \cup \mathcal{R}_c^R} z_{c,r',r} \quad \forall t \in \mathcal{T}_r, r \in \mathcal{R}^T \quad (42)$$

$$m_g = \sum_{c \in \mathcal{C}_g} \sum_{r \in \mathcal{R}^T \cup \mathcal{R}_c^M \cup \mathcal{R}_c^R} z_{c,0,r} \quad \forall g \in \mathcal{G} \quad (43)$$

Constraints (42) represent the coupling constraints between rolling stock units and crew members on trips. Specifically, if trip r starts at timestamp t (i.e., $x_{r,t} = 1$), it must be assigned to one crew from the source (i.e., $\sum_{c \in \mathcal{C}} z_{c,0,r} = 1$) as the first task or other tasks (i.e., $\sum_{c \in \mathcal{C}} \sum_{r' \in \mathcal{R}_r^{T-} \cup \mathcal{R}_c^M \cup \mathcal{R}_c^R} z_{c,r',r} = 1$). Lastly, we calculate the total number of crew members required for each crew group g in constraints (43).

Group III: Range of the decision variables

All the train route selection, trip departure time selection, and crew task connection decision variables are binary variables, and all decision variables for trip connection times, and number of required crew members are positive integer variables, as formulated in the following constraints.

$$u_{k,l} \in \{0, 1\} \quad \forall l \in \mathcal{L}_k, k \in \mathcal{K} \quad (44)$$

$$x_{r,t} \in \{0, 1\} \quad \forall t \in \mathcal{T}_r, r \in \mathcal{R}^T \cup \mathcal{R}^M \cup \mathcal{R}^R \quad (45)$$

$$z_{c,0,r}, z_{c,r,r'}, z_{c,r,0} \in \{0, 1\} \quad \forall r, r' \in \mathcal{R}^T \cup \mathcal{R}_c^M \cup \mathcal{R}_c^R, r \neq r', c \in \mathcal{C} \quad (46)$$

$$v_{k,\sigma(r),r} \in \mathbb{Z}^+ \quad \forall r \in \mathcal{R}_l^T \setminus \{r_0(l)\}, l \in \mathcal{L}_k, k \in \mathcal{K} \quad (47)$$

$$m_g \in \mathbb{Z}^+ \quad \forall g \in \mathcal{G} \quad (48)$$

With Eqs. (24)–(48), the primal mathematical formulation M6 is modeled for the JRCSP. Then, a general-purpose solver (e.g., CPLEX) can be employed to solve the primal model M6 in small-scale instances (see Section 6.1 for numerical experiments). However, owing to a large number of system constraints and integer decision variables with their complex coupling relationships, it is difficult to solve the model for large-scale instances. Specifically, in order to find feasible trip sequences and task sequences for rolling stock units and crew members, a large number of constraints (i.e., constraints (25)–(26), (30)–(41)) are required to enforce the flow conservation for rolling stock units and crew members, which could result in poor computational efficiency.

B Mathematical model properties and proofs

Proposition 1. *The Benders sub-problem M2 and Benders dual sub-problem (DSP) M7 in Eq. (49) are always feasible and bounded for any given feasible vector $\bar{\lambda} \in \Lambda$.*

$$\text{M7: } \begin{cases} \min & \sum_{r \in \mathcal{R}^T} \sum_{t \in \mathcal{T}_r} \sum_{p \in \mathcal{P}^R} \bar{\lambda}_p \alpha_{p,r,t} \cdot \theta_{r,t}^1 + \sum_{p \in \mathcal{P}^C} \theta_p^2 & (49a) \\ \text{s.t.} & \sum_{r \in \mathcal{R}^T} \sum_{t \in \mathcal{T}_r} \gamma_{p,r} \cdot \theta_{r,t}^1 + \theta_p^2 \leq C_p^C \quad \forall p \in \mathcal{P}^C & (49b) \\ & \theta_{r,t}^1 \leq 0 \quad \forall t \in \mathcal{T}_r, r \in \mathcal{R}^T & (49c) \\ & \theta_p^2 \leq 0 \quad \forall p \in \mathcal{P}^C & (49d) \end{cases}$$

Proof. We first prove that the BSP M2 is always feasible and bounded, and then the DSP M7 is also always feasible and bounded based on the complementary slackness condition. Without loss of generality, given a feasible solution $\bar{\lambda} = \{\bar{\lambda}_p\}_{p \in \mathcal{P}^R} \in \Lambda$ of the relaxed BMP M4, we generate a set of trips $\bar{\mathcal{R}}_{\bar{\lambda}}^C = \{(r, t) \mid \sum_{p \in \mathcal{P}^R} \bar{\lambda}_p \alpha_{p,r,t} \geq 1, \forall t \in \mathcal{T}_r, r \in \mathcal{R}^T\}$ that are required to be covered by crew members. For convenience, virtual source vertex (d_{ori}^C, t_0) and virtual sink vertex $(d_{des}^C, t_{|\mathcal{T}|})$ are used to indicate the source and sink for crew members. Then, we construct an artificial feasible solution by assigning one available crew member $c \in \mathcal{C}_g, g \in \mathcal{G}_t$ to cover only one trip $(r, t) \in \bar{\mathcal{R}}_{\bar{\lambda}}^C$, which corresponds to a feasible task sequence p with the satisfaction of meal/rest task requirements (e.g., $p = (d_{ori}^C, t_0) \rightarrow (r, t) \rightarrow (d_{des}^C, t_{|\mathcal{T}|})$). Considering that the total number of crew members is not limited (see Assumption 4) and the total available time horizon for available crew groups lasts for the whole time horizon (i.e., $\bigcup_{g \in \mathcal{G}} \mathcal{T}_g = \mathcal{T}$), it is always feasible to construct a solution μ^* of the BSP M2 with the bounded operational cost as $\sum_{g \in \mathcal{G}} c_g^4 \varphi_{g,p(r)}$, where $\varphi_{g,p(r)}$ is equal to 1 if and only if crew task sequence $p(r)$ associated with trip r is assigned to one crew member in crew group g . Then, the feasibility and bounded conditions are guaranteed for BSP M2 and DSP M7. $\square$

Proposition 2. *The lower-bound lifting (LBL) valid inequalities in Eq. (15) is effective for the Benders master problem M4.*

Proof. Considering the BMP M4 with LBL cuts, based on the given column selection solution $\bar{\lambda} = \{\bar{\lambda}_p\}_{p \in \mathcal{P}^R} \in \Lambda$ for rolling stock units, the objective value of LBL cuts with respect to variable η

(denoted by $\eta_{LBL}^*(\bar{\lambda})$) could be calculated based on Eq. (50).

$$\eta_{LBL}^*(\bar{\lambda}) \geq \sum_{p \in \mathcal{P}^R} \sum_{r \in \mathcal{R}^T} \sum_{t'=t}^{\min\{\underline{T}_{r,t}^D, \bar{T}_{g(r,t)}\}} c_{g(r,t)}^4 \alpha_{p,r,t} \bar{\lambda}_p \quad \forall \bar{\lambda} \in \Lambda \quad (50)$$

At the same time, considering the BMP M3 without LBL cuts, we denote by $\eta_{BMP}^*(\bar{\lambda})$ as the optimal objective value of variable η with the extreme point set $\mathcal{O}_\Delta$, based on the given column selection solution $\bar{\lambda}$ for rolling stock units. To prove that the LBL cuts are valid to lift valid lower bounds to the BMP M4, if and only if we could prove the condition in Eq. (51) equivalently.

$$\eta_{LBL}^*(\bar{\lambda}) \leq \eta_{BMP}^*(\bar{\lambda}) \quad \forall \bar{\lambda} \in \Lambda \quad (51)$$

On the one hand, we denote by $\mu^*(\bar{\lambda}) = \{\mu_p^*(\bar{\lambda})\}_{p \in \mathcal{P}^C}$ as the optimal solution of crew columns with respect to the BSP M2 with given solution $\bar{\lambda}$ for rolling stock units. Then, the following Eq. (52) always holds, where $\eta_{BSP}^*(\bar{\lambda})$ denotes the optimal value of the BSP M2, where $\varphi_{g,p}$ is equal to 1 if and only if path p is assigned to one crew member in crew group g .

$$\eta_{BMP}^*(\bar{\lambda}) \geq \eta_{BSP}^*(\bar{\lambda}) = \sum_{g \in \mathcal{G}} \sum_{p \in \mathcal{P}^C} c_g^4 \mu_p^* \varphi_{g,p} \quad (52)$$

On the other hand, for one trip r starting at timestamp t with respect to rolling stock column $p \in \mathcal{P}^R$, we observe that there is at most one trip can be covered by one crew member in crew group $g(r,t)$ during the time window $[t, \min\{\underline{T}_{r,t}^D, \bar{T}_{g(r,t)}\}]$, then the following Eq. (53) holds. Particularly, to guarantee the uniqueness of the associated crew group $g(r,t)$ of trip r with starting from timestamp t , we assume that the intersections of the available time windows between any two crew groups are empty sets (i.e., $\mathcal{T}_g \cap \mathcal{T}_{g'} = \emptyset, \forall g, g' \in \mathcal{G}, g \neq g'$).

$$\sum_{t'=t}^{\min\{\underline{T}_{r,t}^D, \bar{T}_{g(r,t)}\}} \alpha_{p,r,t'} \leq 1 \quad p \in \mathcal{P}^R, t \in \mathcal{T}_r, r \in \mathcal{R}^T \quad (53)$$

Since $\mu_p^* \geq 0, p \in \mathcal{P}^C$ and $0 \leq \gamma_{p,r} \leq 1, r \in \mathcal{R}^T, p \in \mathcal{P}^C$, we could formulate Eq. (54) by multiplying each term in Eq. (53) with $c_{g(r,t)}^4, \mu_p^*$ and $\gamma_{p,r}$, and then summarize them over path $p \in \mathcal{P}^C$.

$$\sum_{t'=t}^{\min\{\underline{T}_{r,t}^D, \bar{T}_{g(r,t)}\}} c_{g(r,t)}^4 \alpha_{p,r,t'} \sum_{p' \in \mathcal{P}^C} \gamma_{p',r} \mu_{p'}^* \leq \sum_{p' \in \mathcal{P}^C} c_{g(r,t)}^4 \mu_{p'}^* \quad p \in \mathcal{P}^R, t \in \mathcal{T}_r, r \in \mathcal{R}^T \quad (54)$$

By submitting Eq. (6) into Eq. (54) and summarizing them over $r \in \mathcal{R}^T$, we then have Eq. (55).

$$\sum_{r \in \mathcal{R}^T} \sum_{p \in \mathcal{P}^R} \sum_{t'=t}^{\min\{\underline{T}_{r,t}^D, \bar{T}_{g(r,t)}\}} c_{g(r,t)}^4 \alpha_{p,r,t'} \bar{\lambda}_p \leq \sum_{r \in \mathcal{R}^T} \sum_{p \in \mathcal{P}^C} c_{g(r,t)}^4 \mu_p^* \quad t \in \mathcal{T} \quad (55)$$

We observe that the left hand of Eq. (55) is equivalent to the objective value of variable η with respect to LBL cut (i.e., $\eta_{LBL}^*(\bar{\lambda})$), and the right hand of Eq. (55) is equivalent to the objective value of BSP (i.e., $\eta_{BSP}^*(\bar{\lambda})$). Then, the condition is guaranteed with combining Eqs. (51), (52), and (55), as given in the following Eq. (56). Therefore, the LBL valid equation in Eq. (15) is effective for the BMP M3.

$$\eta_{LBL}^*(\bar{\lambda}) \leq \eta_{BSP}^*(\bar{\lambda}) \leq \eta_{BMP}^*(\bar{\lambda}) \quad (56)$$

□

Proposition 3. Let z_{M4} and UB be the current lower bound and upper bound on the optimal value of the relaxed BMP $M4$, respectively. If λ_p is a non-basic variable in the optimal solution to the relaxed BMP $M4$ and $z_{M4} + RC_{k,l} > UB$, where $RC_{k,l}$ is the reduced cost associated with rolling stock unit $k = k(p)$ and train route $l = l(p)$, then $\lambda_p = 0$ in any optimal solution of the relaxed BMP $M4$.

Proof. The result follows from the fact that $z_{M4} + RC_{k,l}$ is a lower bound on the objective function value of the relaxed BMP $M4$ if one column associated with rolling stock unit k and train route l (i.e., $p \in \overline{\mathcal{P}}_{k,l}^R$) is selected. Therefore, if $z_{M4} + RC_{k,l} > UB$, then $\lambda_p = 0, \forall p \in \overline{\mathcal{P}}_{k,l}^R$ in any final optimal integer solution. $\square$

Proposition 4. For two child nodes n_1 and n_2 of parent node n_0 : (a) When one of the branching rules regarding auxiliary variables $\overline{\mathbf{u}}, \overline{\mathbf{x}}$ in the relaxed BMP $M4$ of node n_0 is selected for generating nodes n_1 and n_2 , the Benders cuts generated at node n_1 are still valid at both nodes n_1 and n_2 with their descendent nodes. At the same time, the Benders cuts generated at node n_2 are also valid at both nodes n_1 and n_2 with their descendent nodes. However, (b) when one of the branching rules regarding auxiliary variables $\overline{\mathbf{n}}, \overline{\mathbf{z}}$ in the BSP $M2$ of node n_0 is selected for generating nodes n_1 and n_2 , the Benders cuts generated at node n_1 (n_2) are only valid at the descendent nodes of node n_1 (n_2) with the same branch.

Proof. Let Δ denote the polyhedron defined by the dual solution from the DSP $M7$, as defined in Section 5.2. When one of the branching rules at node n_0 regarding auxiliary variables $\overline{\mathbf{u}}, \overline{\mathbf{x}}$ in the relaxed BMP $M4$ is selected, the polyhedrons Δ are same between two child nodes n_1 and n_2 , which means the generated Benders cuts at node n_1 are still valid at node n_2 , and the generated Benders cuts at node n_2 are also valid at node n_1 . However, when one of the branching rules at node n_0 regarding auxiliary variables $\overline{\mathbf{n}}, \overline{\mathbf{z}}$ in the relaxed BSP $M2$ is selected, the polyhedrons Δ are really different between two child nodes n_1 and n_2 due to the branching rule at node n_0 . Thus, the generated Benders cuts at node n_1 (n_2) are only valid at its child and descendent nodes with polyhedrons $\Delta' \subseteq \Delta$. $\square$

C Label correcting (LC) algorithm

C.1 LC algorithm for rolling stock units

To apply the LC algorithm for rolling stock units, a new space-time network $\mathcal{G}_{k,l}^R = \{\mathcal{V}_{k,l}^R, \mathcal{A}_{k,l}^R\}$ for train route l and rolling stock unit k is constructed based on the involved trips, where $\mathcal{V}_{k,l}^R$ is the set of related space-time vertices and $\mathcal{A}_{k,l}^R$ is the set of related space-time arcs. In particular,

$$\mathcal{V}_{k,l}^R = \{(r, t) \mid t \in \mathcal{T}_r, r \in \mathcal{R}_{k,l}\}$$

and

$$\mathcal{A}_{k,l}^R = \{(r, r', t, t') \mid (r, t) \in \mathcal{V}_l^R, (r', t') \in \mathcal{V}_{k,l}^R, T_{r,r'}^{\min} + T_r^R \leq t' - t \leq T_{r,r'}^{\max} + T_r^R\}, l \in \mathcal{L}_k, k \in \mathcal{K},$$

where notation (r, r', t, t') represent a space-time arc from space-time vertex (r, t) to vertex (r', t') . Lastly, each space-time arc (r, r', t, t') is associated with a cost $\tilde{c}_{r,r',t,t'}^R$, which is calculated based on Eqs. (57)–(58), where $a(r)$ and $a(r')$ are the associated physical segments of trips r and r' , respectively.

$$\tilde{c}_{r,r',t,t'}^R = \begin{cases} c_{k,l}^1 - \pi_k + \tilde{c}_{r',t'}^R, & \text{if } (r, r', t, t') = (d_{ori}(k), r', t_0, t'), t' \in \mathcal{T}_{r'}, r' \in \mathcal{R}_l^T \\ c_{r,r'}^3(t' - t - T_r^R) + \tilde{c}_{r',t'}^R, & \text{if } t \in \mathcal{T}_r, t' \in \mathcal{T}_{r'}, r, r' \in \mathcal{R}_l^T, r \neq r' \\ 0, & \text{if } (r, r', t, t') = (r, d_{des}(k), t, t_{|\mathcal{T}|}), t \in \mathcal{T}_r, r \in \mathcal{R}_l^T \\ 0, & \text{otherwise} \end{cases} \quad (57)$$

$$\begin{aligned}
\tilde{c}_{r',t'}^R = & c_{r',t'}^2 - \sum_{(r'',r''') \in \mathcal{V}} \sum_{\tau=\max\{t'-h_{r'',r'''}^{\min}, 1, 0\}}^{t'-1} \pi_{\tau,r',r'''}^2 \\
& - \sum_{(r'',r''') \in \mathcal{V}} \pi_{t',r'',r'''}^2 - \sum_{s \in \mathcal{S}_{a(r')}} \left(-N_{a(r')}^{\min}\right) \pi_{a(r'),t'}^3 \\
& - N_{a(r')}^{\max} \pi_{a(r'),t'}^4 + \sum_{(\theta^1, \theta^2) \in \mathcal{O}_{\Delta}} \pi_{(\theta^1, \theta^2)}^5 \theta_{r',t'}^1 + \sum_{\tau=\max\{\bar{T}_{r',t'}^D, \underline{T}_{g(r',t')}\}}^{t'} \pi_{\tau}^6
\end{aligned} \tag{58}$$

After the pricing problem of the relaxed BMP M4 is decomposed into a series of sub-problems for each rolling stock unit and train route (see Section 5.4.1), one pricing sub-problem $SPR_{k,l}$ for rolling stock unit k and train route l can be viewed as a time-dependent shortest path problem in the space-time network $\mathcal{G}_{k,l}^R$, which can be solved by the LC algorithm with the pseudocode in Algorithm 1.

Algorithm 1 LC algorithm for solving $SPR_{k,l}$ of rolling stock units

Input:

- 1: The space-time network $\mathcal{G}_{k,l}^R = (\mathcal{V}_{k,l}^R, \mathcal{A}_{k,l}^R)$ for rolling stock unit k and train route l ; current dual solutions $\pi^1 \sim \pi^6$ for constraints in the relaxed BMP M4.

Output:

- 2: One best space-time path $p_{k,l}^*$ for rolling stock unit k and train route l .
 - 3: *# parallel for* (each rolling stock unit k and train route l)
 - 4: **Note:** The hashtag “*# parallel for*” represents the number of parallel CPU processors, and all threads must wait at the joint point until all threads are completed.
 - 5: **Step 1 (Label Initialization):**
 - 6: • Initialize label cost $RC_{d_{ori}(k),t_0} = -\pi_k^1 + c_{k,l}^1$ at origin depot vertex $(d_{ori}(k), t_0)$.
 - 7: • Initialize label costs $RC_{r,t} = +\infty$ at other vertices $(r, t), (i, t) \in \mathcal{V}_{k,l}^R \setminus \{(d_{ori}(k), t_0)\}$.
 - 8: • Sequence labels (r, t) in an order from origin depot vertex $(d_{ori}(k), t_0)$ by using the topological sorting process.
 - 9: • Initialize label predecessors, i.e., $pred_{r,t} = -1$.
 - 10: **Step 2 (Forward updating label dynamically):**
 - 11: **for** each label $b = (r, t), (r, t) \in \mathcal{V}_{k,l}^R$ **do**
 - 12: **for** each label $b' = (r', t'), (r', t') \in \mathcal{V}_{k,l}^R, t' \geq t$ **do**
 - 13: • Check branching rules
 - 14: **if** $\bar{x}_{r,t} == 0 \mid \bar{x}_{r',t'} == 0 \mid \bar{n}_{g(r,t),r} == 0 \mid \bar{n}_{g(r',t'),r'} == 0$ **then**
 - 15: **continue**
 - 16: **end if**
 - 17: **if** $(r == \sigma(r')) \ \& \ (r \neq r' \ \& \ T_r^R + T_{r,r'}^{R\min} \leq t' - t \leq T_r^R + T_{r,r'}^{R\max})$ **then**
 - 18: • Calculate new label cost $RC'_{r',t'} = RC_{r,t} + \tilde{c}_{r,r',t,t'}^R$ based on Eqs. (57)–(58).
 - 19: **if** $RC'_{r',t'} \leq RC_{r',t'}$ **then**
 - 20: • Update label $RC_{r',t'} \leftarrow RC'_{r',t'}$.
 - 21: • Update label predecessor $pred_{r',t'} = (r, t)$.
 - 22: **end if**
 - 23: **end if**
 - 24: **end for**
 - 25: **end for**
 - 26: **Step 3 (Backtrace):**
 - 27: • For rolling stock unit k and train route l , use the label predecessor $pred_{r,t}$ and label costs $RC_{r,t}$ to trace back from destination depot vertex $(d_{des}(k), |T|)$ to origin depot vertex $(d_{ori}(k), t_0)$, and obtain a new optimal column. Then, the generated column should be added to the column pool $\bar{\mathcal{P}}^R$ in subsequent iterations.
-

Specifically, at the beginning of the searching process, we first check the branching rule regarding the train route assignments with rolling stock units. That is, if train route l is forbidden for rolling stock unit k by using the branching rule of the current node (i.e., $\bar{u}_{k,l} = 0$), the whole labeling process will be skipped for the train route l and rolling stock unit k . Then, in the label initialization step, a topological sort is performed to sequence vertices in a non-decreasing order from the origin depot vertex $(d_{ori}(k), t_0)$ for rolling stock unit k . Next, in the forward labeling step, the total reduced cost $RC_{r,t}$ from depot vertex $(d_{des}(k), t_0)$ to vertex (r, t) is used as the label cost at vertex (r, t) . In addition, we use two branching rules to exclude infeasible trips, that is (a) if trip r (or r') is forbidden to start

from timestamp t (or t') (i.e., $\bar{x}_{r,t} = 0$ or $\bar{x}_{r',t'} = 0$), or (b) if trip r (or r') cannot be paired with crew members in crew group g (or g') (i.e., $\bar{n}_{g,r} = 0$ or $\bar{n}_{g',r'} = 0$), then the label updating process associated with trip r at timestamp t will be skipped. A new label cost $RC'_{r',t'}$ at vertex (r', t') is calculated based on label cost $RC_{r,t}$, trip costs, connection costs, and dual values of arc (r, r', t, t') in Eq. (57)–(58). Lastly, after applying the LC algorithm, the shortest path with the minimum negative label cost (i.e., reduced cost) can be found by using the back-tracing procedure from the destination depot vertex $(d_{des}(k), t_{|\mathcal{T}|})$ to the origin depot vertex $(d_{ori}(k), t_0)$ of the corresponding rolling stock unit.

Based on the constructed space-time network $\mathcal{G}_{k,l}^R$ and the description of the pseudocode in LC Algorithm 1, we discuss the time complexity in Proposition 5. Particularly, we observe that the time complexity of the pricing problem of the relaxed BMP M4 is dependent on the scales of space-time vertices (i.e., trips with their available departure timestamps) and their available outgoing arcs of trip connections.

Proposition 5. *Based on the space-time network $\mathcal{G}_{k,l}^R = \{\mathcal{V}_{k,l}^R, \mathcal{A}_{k,l}^R\}$ for train route l and rolling stock unit k , the shortest path p^* with the minimum reduced cost $\overline{C}_{k,l,p^*}^R$ can be found by the LC algorithm with the time complexity $O\left(\sum_{v \in \mathcal{V}_{k,l}^R} |\mathcal{A}_{v+}| \right)$, where $\mathcal{A}_{v+}$ is the set of outgoing arcs from vertex v after implementing the topological sorting process.*

C.2 Multi-stage LC algorithm for crew groups

Similarly, to apply the LC algorithm for crew members, a new space-time network $\mathcal{G}_g^C = \{\mathcal{V}_g^C, \mathcal{A}_g^C\}$ is constructed to implement the LC algorithm for specific crew group g , where $\mathcal{V}_g^C$ and $\mathcal{A}_g^C$ denote the sets of space-time vertices and arcs related to crew group g . Specifically,

$$\mathcal{V}_g^C = \overline{\mathcal{R}}_{\bar{\lambda}}^C \cup \mathcal{R}_g^M \cup \mathcal{R}_g^R \cup \{(d_{ori}^C, t_0), (d_{des}^C, t_{|\mathcal{T}|})\},$$

$$\mathcal{A}_g^C = \{(r, r', t, t') \mid r' \in \mathcal{R}_r^{T+}, T_{r,r'}^{C \min} \leq t' - t - T_r^R \leq T_{r,r'}^{C \max}, (r, t), (r', t') \in \mathcal{V}_g^C, (r, t) \neq (r', t')\},$$

where $\bar{\lambda}$ is the solution obtained from the relaxed BMP M4, (d_{ori}^C, t_0) and $(d_{des}^C, t_{|\mathcal{T}|})$ are the virtual source and sink vertices for showing the source and sink for crew members. Lastly, each space-time arc (r, r', t, t') is associated with a cost $\tilde{c}_{r,r',t,t'}^C$, which is calculated in Eq. (59).

$$\tilde{c}_{r,r',t,t'}^C = \begin{cases} c_g^4, & \text{if } (r, r', t, t') = (d_{ori}^C, r', t_0, t'), (r', t') \in \mathcal{V}_g^C \\ -\theta_{r,t}^1, & \text{if } (r, r', t, t') \in \mathcal{A}_g^C \\ -\theta_{r,t}^1, & \text{if } (r, r', t, t') = (r, d_{des}^C, t, t_{|\mathcal{T}|}), (r, t) \in \mathcal{V}_g^C \\ 0, & \text{otherwise} \end{cases} \quad (59)$$

As displayed in the pseudocode of the multi-stage LC algorithm for solving SPC_g of crew group g in Algorithm 2, based on the stage partition method in Section 5.4.2, the entire planning horizon is first partitioned into a finite number of stages based on the ascending order of meal/rest tasks, denoted by $\mathcal{H} = \{h_0, h_1, h_2, \dots, h_{|\mathcal{R}_g^M| + |\mathcal{R}_g^R|}\}$, where $|\mathcal{R}_g^M| + |\mathcal{R}_g^R|$ is the total number of meal/rest tasks for crew members in crew group g , as illustrated in Fig. 5. Then, the space-time network $\mathcal{G}_g^C$ can be decomposed into space-time sub-networks $\mathcal{G}_{g,h}^C = \{\mathcal{V}_{g,h}^C, \mathcal{A}_{g,h}^C\}$ for crew group g and stage h . Note that vertex set $\mathcal{V}_{g,h}^C$ of stage h contains all trip tasks that start after time $\underline{T}_h$ and before time $\overline{T}_h$ of stage h , which means that we cannot miss any feasible optimal solution with respect to the sub-task-sequence within the time window $[\underline{T}_h, \overline{T}_h]$. Lastly, each space-time vertex $(\tilde{r}, \tilde{t}) \in \mathcal{V}_{g,h}^C$ is associated with a label in stage h .

In the search process, we apply the LC algorithm to find all feasible sub-task-sequences of fixed trips within the time window $[\underline{T}_h, \overline{T}_h]$. Specifically, at the initialization of the LC algorithm in one

Algorithm 2 Multi-stage LC algorithm for solving SPC_g of crew groups

Input:

1: The space-time network $\mathcal{G}_g^C = (\mathcal{V}_g^C, \mathcal{A}_g^C)$ of crew group g ; current dual solutions $\theta^1 \sim \theta^2$ for constraints in BSP M2; meal tasks $\mathcal{R}_g^M$ and rest tasks $\mathcal{R}_g^R$ with respect to crew group g .

Output:

2: One best space-time task sequence p_g^* for crew group g with the requirements of meal/rest tasks.

3: # parallel for (each crew group g)

4: **Step 1 (Multi-stage network partition):**

5: • Initialize multiple stages $\mathcal{H} = \{h_0, h_1, h_2, \dots, h_{|\mathcal{R}_g^M| + |\mathcal{R}_g^R|}\}$.

6: **for** each stage $h = h_0, h_1, h_2, \dots, h_{|\mathcal{H}|-1}$ **do**

7: Construct space-time sub-network $\mathcal{G}_{g,h}^C = \{\mathcal{V}_{g,h}^C, \mathcal{A}_{g,h}^C\}$;

8: **Step 2 (Label Initialization):**

9: • Initialize label cost $RC_{d_{ori}, T_h}^C = 0$ at stage virtual source vertex (d_{ori}^C, T_h) .

10: • Initialize label costs $RC_{r,t} = +\infty$ at other vertices $(r, t) \in \mathcal{V}_g^C \setminus \{(d_{ori}^C, T_h)\}$.

11: • Sequence labels (r, t) in non-decreasing order of timestamp t .

12: • Initialize label predecessors, i.e., $pred_{r,t} = -1$.

13: **Step 3 (Forward updating label dynamically):**

14: **for** each label $b = (r, t) \in \mathcal{V}_{g,h}^C$ and label $b' = (r', t') \in \mathcal{V}_{g,h}^C, (r, t) \neq (r', t')$ **do**

15: • Check branching rules

16: **if** $\bar{n}_{g,r} == 0 \parallel \bar{n}_{g,r'} == 0 \parallel \bar{z}_{g,r,r'} == 0$ **then**

17: **continue**

18: **end if**

19: **if** $(r \neq r') \ \& \ (T_r^R + T_{r,r'}^{C \min} \leq t' - t \leq T_r^R + T_{r,r'}^{C \max})$ **then**

20: • Calculate new label cost $RC'_{r',t'} = RC_{r,t} + \tilde{c}_{r,r',t,t'}^C$ based on Eq. (59).

21: **if** $RC'_{r',t'} \geq RC_{r',t'}$ **then**

22: • Update label $RC_{r',t'} \leftarrow RC'_{r',t'}$.

23: • Update label predecessor $pred_{r',t'} = (r, t)$.

24: **end if**

25: **end if**

26: **end for**

27: **Step 4 (Backtrace):**

28: • For each crew group g , use the label predecessor $pred_{r,t}$ and label costs $RC_{r,t}$ to trace back from stage virtual sink vertex (d_{des}^C, T_h) and determine feasible sub-task-sequences. Then, the generated sub-task-sequences should be added to the column pool $\bar{\mathcal{P}}_{g,h}^C$ with their partial reduced cost $RC_{h,p}$ at current stage h .

29: **end for**

30: **Step 5 (Task sequence merging for multiple stages):**

31: **for** each stage $h = |\mathcal{H}| - 1, \dots, 1$ **do**

32: • Adjacent previous stage $h' = h - 1$

33: **for** $p = 1, 2, \dots, |\bar{\mathcal{P}}_{g,h}^C|$ and $p' = 1, 2, \dots, |\bar{\mathcal{P}}_{g,h'}^C|$ **do**

34: $r_0 \leftarrow$ first assigned trip task of task sequence p

35: $r_{end} \leftarrow$ last assigned trip task of task sequence p'

36: $r_{h'} \leftarrow$ associated meal/rest task of stage h'

37: $i_0 \leftarrow$ departure station of trip task r_0

38: $i_{end} \leftarrow$ arrival station of trip task r_{end}

39: **if** $(r_{end} \neq r_0) \ \& \ (t_{r_0} - t_{r_{end}} \geq T_{r'}^R + T_{r_{end}}^R + T_{r_{end},r'}^C + T_{r',r_0}^C) \ \& \ (i_0 == i_{end})$ **then**

40: • Check branching rules

41: **if** $\bar{n}_{g,r_{end}} == 0 \parallel \bar{n}_{g,r'} == 0 \parallel \bar{n}_{g,r_0} == 0 \parallel \bar{z}_{g,r_{end},r'} == 0 \parallel \bar{z}_{g,r',r_0} == 0$ **then**

42: **continue**

43: **end if**

44: • Construct a new task sequence p'' by merging sub-task-sequences p and p'

45: $p'' = p \cup p'; \bar{\mathcal{P}}_{g,h'}^C = \bar{\mathcal{P}}_{g,h'}^C \cup \{p''\}$.

46: **if** $h' == 1$ **then**

47: $RC_{h,p''} = RC_{h,p} + RC_{h,p'} + c_g^4$

48: **else**

49: $RC_{h,p''} = RC_{h,p} + RC_{h,p'}$

50: **end if**

51: **end if**

52: **end for**

53: **end for**

54: **Step 6 (Task sequence selection):**

55: • Select a new optimal column p^* from $\bar{\mathcal{P}}_{g,h=0}^C$ with the minimum reduced cost (i.e., $RC_{h=0,p^*}$). Then, the generated column should be added to the column pool $\bar{\mathcal{P}}^C$ for crew group g in the subsequent iterations.

stage, all tasks associated with fixed trips are first filtered within the effective working time windows of crew group g , then sorted with the increasing order sequence of their departure times. By this way, the searching sequence of the involved vertices and arcs can be guaranteed to generate feasible and optimal crew task sequences. Then, in the forward labeling step, we first check two branching rules to exclude infeasible trip tasks and task connections, that is, if (a) task r cannot be paired with crew members in crew group g (i.e., $\bar{n}_{g,r} = 0$), or (b) task r cannot be paired with other trips with crew members in crew group g (i.e., $\sum_{r' \in \mathcal{R}_r^{T+}} \bar{z}_{g,r,r'} = 0$ or $\sum_{r' \in \mathcal{R}_r^{T-}} \bar{z}_{g,r',r} = 0$), then the label updating process associated with task r will be skipped. After updating all involved labels in the network, a set of sub-task-sequences can be generated between the vertices within time windows of two adjacent meal/rest tasks. Next, after applying the LC algorithm at all stages, global feasible task sequences could be generated with merging sub-task-sequences at each stage with satisfying the minimum required time for each meal/rest task. Lastly, we select the optimal task sequence based on the minimum total label cost (i.e., reduced cost) along all stages from the virtual source to virtual sink vertices of crew group g .

Proposition 6. *Based on the space-time network $\mathcal{G}_g^C = \{\mathcal{V}_g^C, \mathcal{A}_g^C\}$ for crew group g , a shortest task sequence p^* with the minimum reduced cost $\bar{C}_{g,p^*}^C$ can be found by the multi-stage LC algorithm with the time complexity $O\left(\sum_{v \in \mathcal{V}_{g,h_0}^C} |\mathcal{A}_v^+|\right)$ for the first stage h_0 , $O\left(\sum_{h \in \mathcal{H} \setminus \{h_0, h_{|\mathcal{H}|-1}\}} \sum_{v \in \mathcal{V}_{g,h}^C} |\mathcal{A}_v^+|\right)$ for middle stages $h \in \mathcal{H} \setminus \{h_0, h_{|\mathcal{H}|-1}\}$, and $O\left(\sum_{v \in \mathcal{V}_{g,h_{|\mathcal{H}|-1}}^C} |\mathcal{A}_v^+|\right)$ for the last stage $|\mathcal{H}| - 1$, where $\mathcal{A}_v^+$ is the set of outgoing arcs from vertex v after partitioning trips and tasks into individual stages.*

Based on the description and pseudocode of the multi-stage LC Algorithm 2, we discuss the time complexity in Proposition 6. Particularly, we observe that the time complexity of the pricing problem of BSP M2 is related to not only the number of fixed trip tasks but also the number of stages associated with meal/rest tasks. Specifically, in the first or last stage, it only calls the LC algorithm once to determine sub-task-sequences from timestamp t_0 to vertices in the time window $[\underline{T}_{r_0}, \bar{T}_{r_0}]$ (associated with first meal/rest task r_0), or from vertices in the time window $[\underline{T}_{r_{|\mathcal{R}_g^M|+|\mathcal{R}_g^R|-1}}, \bar{T}_{r_{|\mathcal{R}_g^M|+|\mathcal{R}_g^R|-1}}]$ (associated with last meal/rest task $r_{|\mathcal{R}_g^M|+|\mathcal{R}_g^R|-1}$) to timestamp t_{end} , respectively. While in the middle stages, it needs call the LC algorithm with $(\bar{T}_r - \underline{T}_r)$ times, to determine all-to-all sub-task-sequences starting from vertices in time window $[\underline{T}_{\sigma(r)}, \bar{T}_{\sigma(r)}]$ (associated with previous meal/rest task $\sigma(r)$) and ending vertices in time window $[\underline{T}_r, \bar{T}_r]$ (associated with meal/rest task r), as shown in Fig. 5.

D Priority heuristic rule (PHR) algorithm

In this section, we aim to present a pseudocode for the priority heuristic rule in Algorithm 3 to transform the fractional solution of the relaxed BMP M4 and BSP M2 to a feasible joint rolling stock and crew schedule solution with satisfying all involved constraints. Specifically, we denote by $\bar{\lambda}$ and $\bar{\mu}$ as the optimal fractional solution associated with the LP relaxation of model M1 after processing Benders decomposition at a node in the B&B tree.

E Parallel B&B computing technique

In this section, we aim to present a pseudocode for the branch-and-bound process with the parallel B&B computing (PC) techniques in Algorithm 4. Note that hashtag “# parallel for” represents the number of parallel CPU processors, and all threads must wait at the joint point until all threads are completed. Then, tasks with the operator “—” are executed by multiple threads simultaneously.

Algorithm 3 Priority heuristic rule to generate a feasible solution

Input:
1: A linear optimal solution $(\bar{\lambda}, \bar{\mu})$ of the column selection variables with respect to the LP relaxation of the model M1.
Output:
2: An integer feasible joint optimal solution (λ^*, μ^*) (i.e., an upper bound solution).
3: **if** !ISINTEGER($\bar{\lambda}$) **then**
4: **for** each rolling stock unit $k = 1, 2, \dots, |\mathcal{K}|$ **do**
5: **Step 1.1:** Generate set $\bar{\mathcal{P}}_k^R$ by filtering columns associated with rolling stock unit k .
6: **Step 1.2:** Sort the indices of columns of rolling stock unit k as $p'_1, \dots, p'_q, \dots, p'_{|\bar{\mathcal{P}}_k^R|}$ according to the descending
7: order of the column selection solution (i.e., $\bar{\lambda}_p, p \in \bar{\mathcal{P}}_k^R$).
8: $Column_Flag \leftarrow False$
9: **for** each column $p'_q = p'_1, \dots, p'_{|\bar{\mathcal{P}}_k^R|}$ **do**
10: **Step 2:** Check conflicts between column p'_q and the marked space-time resources.
11: **if** CHECKFEASIBLE(constraints (12c)) **then**
12: **Step 3:** Mark the space-time occupation resources and related headway resources of column p'_q .
13: • **Note:** The space-time resource of one physical segment (e.g., segment a) can be occupied by at most $N_a^{\max}$
14: trips which pass segment a at one timestamp. While the space-time resource within the related headway
15: regions of trips cannot be occupied by any other subsequent columns again.
16: **if** CHECKFEASIBLE(constraints (12d)–(12e)) **then**
17: $Column_Flag \leftarrow True$
18: $\lambda_{p'_q}^* = 1$
19: $\lambda_p^* = 0, \forall p \in \bar{\mathcal{P}}_k^R \setminus \{p'_q\}$.
20: $Break$
21: **end if**
22: **end if**
23: **end for**
24: **if** $Column_Flag == False$ **then**
25: **Step 4:** Find and construct a new space-time shortest path $p_{|\bar{\mathcal{P}}_k^R|+1}$ by applying the LC algorithm (see Section
26: 5.4.1 and EC. C.1)
27: in network $\mathcal{G}_{k,l}^R$ within all train routes l of rolling stock k without passing the marked resources with respect to
28: other rolling stock units.
29: **if** CHECKFEASIBLE(constraints (12d)–(12e)) **then**
30: $Column_Flag \leftarrow True$
31: $\bar{\mathcal{P}}_k^R = \bar{\mathcal{P}}_k^R \cup \left\{ p_{|\bar{\mathcal{P}}_k^R|+1} \right\}$
32: $\lambda_{p_{|\bar{\mathcal{P}}_k^R|+1}}^* = 1$
33: $\lambda^* = \lambda^* \cup \left\{ \lambda_{p_{|\bar{\mathcal{P}}_k^R|+1}}^* \right\}$
34: $\lambda_p^* = 0, \forall p \in \bar{\mathcal{P}}_k^R \setminus \left\{ p_{|\bar{\mathcal{P}}_k^R|+1} \right\}$
35: **end if**
36: **end if**
37: **end for**
38: **Step 5:** Solve the BSP M2 again and update $\bar{\mu}$ by using the column generation method (see Section 5.4.2 and EC. C.2)
39: based on the newly selected and constructed rolling stock columns.
40: **end if**
41: **if** !ISINTEGER($\bar{\mu}$) **then**
42: **Step 6:** Sort the indices of columns for crew members as $p'_1, \dots, p'_q, \dots, p'_{|\bar{\mathcal{P}}^C|}$ according to the descending order of
43: the column selection solution (i.e., $\bar{\mu}_p, p \in \bar{\mathcal{P}}^C$).
44: **for** each column $p'_q = p'_1, \dots, p'_{|\bar{\mathcal{P}}^C|}$ **do**
45: **Step 7.1:** For column p'_q , calculate the number $N_{p'_q}$ of covered trip tasks which have not been marked for selected
46: columns.
47: **if** $N_{p'_q} > 0$ **then**
48: **Step 7.2:** Mark the newly generated trip tasks of column p'_q (The marked tasks cannot be occupied by any other
49: subsequent columns again).
50: $\mu_{p'_q}^* = 1$
51: **end if**
52: **end for**
53: **while** Not all trip tasks are covered **do**
54: **Step 8.1:** Find and construct a new shortest path $p_{|\bar{\mathcal{P}}^C|+1}$ by applying the multi-stage LC algorithm (see Section
55: 5.4.2 and EC. C.2) with assigning a new available crew member.
56: **Step 8.2:** Mark the trip tasks of column $p_{|\bar{\mathcal{P}}^C|+1}$ (The marked tasks cannot be occupied by any other subsequent
57: columns again).
58: $\bar{\mathcal{P}}^C = \bar{\mathcal{P}}^C \cup \left\{ p_{|\bar{\mathcal{P}}^C|+1} \right\}$
59: $\mu_{p_{|\bar{\mathcal{P}}^C|+1}}^* = 1$
60: $\mu^* = \mu^* \cup \left\{ \mu_{p_{|\bar{\mathcal{P}}^C|+1}}^* \right\}$
61: **end while**
62: **end if**

Algorithm 4 Parallel branch-and-bound algorithm

```

1: function BREADTH_FIRST_SEARCH (BFS)(origin_node)
2:   • Push back origin_node into the active node list ANL:  $ANL \leftarrow \text{PUSH}(ANL, origin\_node)$ .
3:   while True do
4:     if  $ANL == \phi$  or  $gap = (GUB - GLB)/GUB < \epsilon$  then
5:       Break
6:     end if
7:     • Initialize local upper bound  $LUB = GUB$  and local lower bound  $LLB = GLB$  on the current layer.
8:     • Distribute all nodes in ANL to  $m$  processors.
9:     # parallel for m processors (Inner parallel computation)
10:    — Solve nodes parallelly and averagely in each processor, and obtain upper bound / lower bound / child
11:    nodes on each node.
12:    • Update local upper bound  $LUB$  / local lower bound  $LLB$  with the minimum lower bound / upper bound
13:    within all solutions of nodes.
14:    if  $LUB < GUB$  then
15:       $GUB \leftarrow LUB$ 
16:    end if
17:    if  $LLB > GLB$  then
18:       $GLB \leftarrow LLB$ 
19:    end if
20:    • Push back all child nodes from different processors into ANL.
21:  end while
22: end function

```

References

- Abbink, E., van den Berg, B., Kroon, L., Salomon, M., 2004. Allocation of railway rolling stock for passenger trains. *Transportation Science* 38, 33–41.
- Abbink, E., Fischetti, M., Kroon, L., Timmer, G., Vromans, M., 2005. Reinventing crew scheduling at netherlands railways. *Interfaces* 35, 393–401.
- Abbink, E.J.W., Albino, L., Dollevoet, T., Huisman, D., Roussado, J., Saldanha, R.L., 2011. Solving large scale crew scheduling problems in practice. *Public Transport* 3, 149–164.
- Adulyasak, Y., Cordeau, J.F., Jans, R., 2015. Benders decomposition for production routing under demand uncertainty. *Operations Research* 63, 851–867.
- Amberg, B., Amberg, B., Kliwer, N., 2019. Robust efficiency in urban public transportation: Minimizing delay propagation in cost-efficient bus and driver schedules. *Transportation Science* 53, 89–112.
- Andrade-Michel, A., Ríos-Solís, Y.A., Boyer, V., 2021. Vehicle and reliable driver scheduling for public bus transportation systems. *Transportation Research Part B: Methodological* 145, 290–301.
- Bach, L., Dollevoet, T., Huisman, D., 2016. Integrating timetabling and crew scheduling at a freight railway operator. *Transportation Science* 50, 878–891.
- Borndörfer, R., Schulz, C., Seidl, S., Weider, S., 2017. Integration of duty scheduling and rostering to increase driver satisfaction. *Public Transport* 9, 177–191.
- Breugem, T., Dollevoet, T., Huisman, D., 2022a. Is equality always desirable? analyzing the trade-off between fairness and attractiveness in crew rostering. *Management Science* 68, 2619–2641.
- Breugem, T., van Rossum, B., Dollevoet, T., Huisman, D., 2022b. A column generation approach for the integrated crew re-planning problem. *Omega* 107, 102555.
- Cacchiani, V., Caprara, A., Toth, P., 2008. A column generation approach to train timetabling on a corridor. *4OR* 6, 125–142.
- Cacchiani, V., Caprara, A., Toth, P., 2010a. Non-cyclic train timetabling and comparability graphs. *Operations Research Letters* 38, 179–184.
- Cacchiani, V., Caprara, A., Toth, P., 2010b. Solving a real-world train-unit assignment problem. *Mathematical Programming* 124, 207–231.
- Cacchiani, V., Caprara, A., Toth, P., 2013. A Lagrangian heuristic for a train-unit assignment problem. *Discrete Applied Mathematics* 161, 1707–1718.
- Canca, D., Andrade-Pineda, J.L., De los Santos, A., Calle, M., 2018. The railway rapid transit frequency setting problem with speed-dependent operation costs. *Transportation Research Part B: Methodological* 117, 494–519.

- Caprara, A., Fischetti, M., Toth, P., 2002. Modeling and solving the train timetabling problem. *Operations Research* 50, 851–861.
- Caprara, A., Kroon, L., Monaci, M., Peeters, M., Toth, P., 2007. Chapter 3 passenger railway optimization, in: Barnhart, C., Laporte, G. (Eds.), *Transportation*. Elsevier. volume 14 of *Handbooks in Operations Research and Management Science*, pp. 129–187.
- Cordeau, J.F., Soumis, F., Desrosiers, J., 2001a. Simultaneous assignment of locomotives and cars to passenger trains. *Operations Research* 49, 531–548.
- Cordeau, J.F., Stojković, G., Soumis, F., Desrosiers, J., 2001b. Benders decomposition for simultaneous aircraft routing and crew scheduling. *Transportation Science* 35, 375–388.
- Correia Duarte, P.J., Dauzère-Pérès, S., Huisman, D., Mannino, C., Sartor, G., Weik, N., Widmann, P., 2025. 50 years of or in railway timetabling and rolling stock planning. *EURO Journal on Transportation and Logistics* 14, 100155.
- Dauzère-Pérès, S., De Almeida, D., Guyon, O., Benhizia, F., 2015. A Lagrangian heuristic framework for a real-life integrated planning problem of railway transportation resources. *Transportation Research Part B: Methodological* 74, 138–150.
- Desaulniers, G., Hickman, M.D., 2007. Chapter 2 public transit, in: Barnhart, C., Laporte, G. (Eds.), *Transportation*. Elsevier. volume 14 of *Handbooks in Operations Research and Management Science*, pp. 69–127.
- Feng, T., Lusby, R.M., Zhang, Y., Tao, S., Zhang, B., Peng, Q., 2024. A branch-and-price algorithm for integrating urban rail crew scheduling and rostering problems. *Transportation Research Part B: Methodological* 183, 102941.
- Gattermann-Itschert, T., Poreschack, L.M., Thonemann, U.W., 2023. Using machine learning to include planners’ preferences in railway crew scheduling optimization. *Transportation Science* 57, 796–812.
- Goel, A., Irnich, S., 2017. An exact method for vehicle routing and truck driver scheduling problems. *Transportation Science* 51, 737–754.
- Han, A.F., Li, E.C., 2014. A constraint programming-based approach to the crew scheduling problem of the taipei mass rapid transit system. *Annals of Operations Research* 223, 173–193.
- Heil, J., Hoffmann, K., Buscher, U., 2020. Railway crew scheduling: Models, methods and applications. *European Journal of Operational Research* 283, 405–425.
- Hoffmann, K., Buscher, U., Neufeld, J.S., Tamke, F., 2017. Solving practical railway crew scheduling problems with attendance rates. *Business & Information Systems Engineering* 59, 147–159.
- Huisman, D., Kroon, L.G., Lentink, R.M., Vromans, M.J.C.M., 2005. Operations research in passenger railway transportation. *Statistica Neerlandica* 59.
- Jütte, S., Müller, D., Thonemann, U.W., 2017. Optimizing railway crew schedules with fairness preferences. *Journal of Scheduling* 20, 43–55.
- Kidd, M.P., Lusby, R.M., Larsen, J., 2019. Passenger- and operator-oriented scheduling of large railway projects. *Transportation Research Part C: Emerging Technologies* 102, 136–152.
- Kroon, L., Fischetti, M., 2001. Crew Scheduling for Netherlands Railways “Destination: Customer”. Springer Berlin Heidelberg, Berlin, Heidelberg. pp. 181–201.
- Kroon, L., Huisman, D., Abbink, E., Fioole, P.J., Fischetti, M., Maróti, G., Schrijver, A., Steenbeek, A., Ybema, R., 2009. The new dutch timetable: The or revolution. *Interfaces* 39, 6–17.
- Kroon, L., Maróti, G., Nielsen, L., 2015. Rescheduling of railway rolling stock with dynamic passenger flows. *Transportation Science* 49, 165–184.
- Kroon, L.G., Peeters, L.W.P., Wagenaar, J.C., Zuidwijk, R.A., 2014. Flexible connections in PESP models for cyclic passenger railway timetabling. *Transportation Science* 48, 136–154.
- Lin, Z., Kwan, R.S., 2016. A branch-and-price approach for solving the train unit scheduling problem. *Transportation Research Part B: Methodological* 94, 97–120.
- Löbel, A., 1998. Vehicle scheduling in public transit and Lagrangean pricing. *Management Science* 44, 1637–1649.
- Lusby, R.M., Haahr, J.T., Larsen, J., Pisinger, D., 2017. A branch-and-price algorithm for railway rolling stock rescheduling. *Transportation Research Part B: Methodological* 99, 228–250.
- Lusby, R.M., Larsen, J., Bull, S., 2018. A survey on robustness in railway planning. *European Journal of Operational Research* 266, 1–15.
- Lusby, R.M., Larsen, J., Ehrgott, M., Ryan, D., 2011. Railway track allocation: models and methods. *OR Spectrum* 33, 843–883.

- Magnanti, T.L., Wong, R.T., 1981. Accelerating Benders decomposition: Algorithmic enhancement and model selection criteria. *Operations Research* 29, 464–484.
- Nguyen, S., Pallottino, S., Malucelli, F., 2001. A modeling framework for passenger assignment on a transport network with timetables. *Transportation Science* 35, 238–249.
- 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.
- Pan, H., Liu, Z., Yang, L., Liang, Z., Wu, Q., Li, S., 2021. A column generation-based approach for integrated vehicle and crew scheduling on a single metro line with the fully automatic operation system by partial supervision. *Transportation Research Part E: Logistics and Transportation Review* 152, 102406.
- Pan, H., Yang, L., Liang, Z., 2023. Demand-oriented integration optimization of train timetabling and rolling stock circulation planning with flexible train compositions: A column-generation-based approach. *European Journal of Operational Research* 305, 184–206.
- Park, T., Ryu, K.R., 2006. Crew pairing optimization by a genetic algorithm with unexpressed genes. *Journal of Intelligent Manufacturing* 17, 375–383.
- Potthoff, D., Huisman, D., Desaulniers, G., 2010. Column generation with dynamic duty selection for railway crew rescheduling. *Transportation Science* 44, 493–505.
- Päppler, P., Neufeld, J.S., Buscher, U., Scheffler, M., Wastian, M., Rosenberger, J., Joshi, K., Kocatürk, F., Ehmke, J., Kunovjanek, M., Schwab, N., Popper, N., 2025. Challenges and the need to integrate rolling stock and crew scheduling for efficient railway operations. *IFAC-PapersOnLine* 59, 445–450. 11th Vienna International Conference on Mathematical Modelling MATHMOD 2025.
- Rähmann, C., Wagener, F., Thonemann, U.W., 2021. Robust tactical crew scheduling under uncertain demand. *Transportation Science* 55, 1392–1410.
- Ruther, S., Boland, N., Engineer, F.G., Evans, I., 2017. Integrated aircraft routing, crew pairing, and tail assignment: Branch-and-price with many pricing problems. *Transportation Science* 51, 177–195.
- Sandhu, R., Klabjan, D., 2007. Integrated airline fleet and crew-pairing decisions. *Operations Research* 55, 439–456.
- Schettini, T., Jabali, O., Malucelli, F., 2022. Demand-driven timetabling for a metro corridor using a short-turning acceleration strategy. *Transportation Science* 56, 919–937.
- Tatsuhiro, S., Shuichiro, S., Toyohisa, M., Nobutaka, U., Murata, T., 2009. Crew and vehicle rescheduling based on a network flow model and its application to a railway train operation. *IAENG International Journal of Applied Mathematics* 39.
- Vaidyanathan, Balachandran and Ahuja, R.K., 2015. *Crew Scheduling Problem*. Springer US, Boston, MA. pp. 163–175.
- Van Lieshout, R.N., 2021. Integrated periodic timetabling and vehicle circulation scheduling. *Transportation Science* 55, 768–790.
- Van Lieshout, R.N., Bouman, P.C., van den Akker, M., Huisman, D., 2021. A self-organizing policy for vehicle dispatching in public transit systems with multiple lines. *Transportation Research Part B: Methodological* 152, 46–64.
- Veelenturf, L.P., Potthoff, D., Huisman, D., Kroon, L.G., 2012. Railway crew rescheduling with retiming. *Transportation Research Part C: Emerging Technologies* 20, 95–110.
- Wang, E., Yang, L., Li, P., Zhang, C., Gao, Z., 2023. Joint optimization of train scheduling and routing in a coupled multi-resolution space-time railway network. *Transportation Research Part C: Emerging Technologies* 147, 103994.
- Wang, E., Yang, L., Yin, J., Zhang, J., Gao, Z., 2024. Passenger-oriented rolling stock scheduling in the metro system with multiple depots: Network flow based approaches. *Transportation Research Part B: Methodological* 180, 102885.
- Wong, R.C.W., Yuen, T.W.Y., Fung, K.W., Leung, J.M.Y., 2008. Optimizing timetable synchronization for rail mass transit. *Transportation Science* 42, 57–69.
- Wu, L., Adulyasak, Y., Cordeau, J.F., Wang, S., 2022. Vessel service planning in seaports. *Operations Research* 70, 2032–2053.
- Yao, Z., Nie, L., Yue, Y., He, Z., Ke, Y., Mo, Y., Wang, H., 2023. Network periodic train timetabling with integrated stop planning and passenger routing: A periodic time-space network construct and ADMM algorithm. *Transportation Research Part C: Emerging Technologies* 153, 104201.

- Yin, J., D'Ariano, 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.
- Yuan, J., Gao, Y., Li, S., Liu, P., Yang, L., 2022. Integrated optimization of train timetable, rolling stock assignment and short-turning strategy for a metro line. *European Journal of Operational Research* 301, 855–874.
- Zhou, H., Qi, J., Yang, L., Shi, J., Pan, H., Gao, Y., 2022. Joint optimization of train timetabling and rolling stock circulation planning: A novel flexible train composition mode. *Transportation Research Part B: Methodological* 162, 352–385.