Transportation Research Part C 194 (2027) 105996

Contents lists available at ScienceDirect

Transportation Research Part C

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

A decomposition-based matheuristic for the passenger-oriented integrated passenger-freight transport problem

Siqiao Li [a,*], Patrick Stokkink [b], Oded Cats [a], Mahnam Saeednia [a]

[a] Department of Transport and Planning, Faculty of Civil Engineering and Geosciences, Delft University of Technology, Delft, the Netherlands
[b] Department of Engineering Systems and Services, Faculty of Technology, Policy and Management, Delft University of Technology, Delft, the Netherlands

A R T I C L E I N F O

Keywords:
Pickup and delivery problem
Integrated transport
Matheuristic
Adaptive large neighborhood search

A B S T R A C T

Integrating passenger and freight transport has become increasingly important for improving the efficiency of urban mobility systems, as planning them separately often results in underutilized capacity and redundant vehicle movements. Coordinating the two enables more effective use of existing transport resources for public transport operators, helps logistics providers maintain cost-effective and reliable deliveries, and assists public authorities in alleviating urban congestion. When implementing such integration, railway stations play a key role as urban hotspots with limited access, where road-rail connections can facilitate efficient intermodal operations. In this work, we introduce a passenger-oriented pickup and delivery problem with explicit road-rail connections and a fleet of shared-use light vehicles. The problem is formulated as a mixed-integer linear program to jointly determine fleet size, routing, and scheduling plans, ensuring a balanced service quality for both passengers and freight. A decomposition-based matheuristic is developed that iteratively coordinates a routing-oriented master problem and a scheduling-oriented subproblem, with feasibility partially restored through a schedule-shifting strategy. The master problem is tackled using an adaptive large neighborhood search enhanced with variable neighborhood search, featuring a novel adaptive selection mechanism based on the relative contributions of service quality indicators. The resulting routing decisions are then passed to the scheduling subproblem, which is solved efficiently using a propagation-based approach that guarantees local optimality. Extensive sensitivity and benchmark analyses demonstrate the benefits of integrated transport in reducing delays, particularly in resource-constrained settings and across different passenger–freight demand compositions.

1. Introduction

Public authorities are increasingly confronted with rising traffic volumes and their associated environmental concerns in urban areas. Among the major contributors are high-polluting trucks and vans used for last-mile deliveries, which exacerbate both congestion and emissions. To mitigate these impacts, many cities have introduced regulatory measures such as low-emission zones and access charges (EIT Urban Mobility, 2022; Watkins et al., 2024). While such policies help achieve mobility and air quality objectives, they also impose stricter operational constraints on logistics service providers, increasing delivery costs and complicating the provision of reliable service levels. Moreover, the independent scheduling of passenger and freight transport often leads to underutilized vehicle capacity and operational inefficiencies. The lack of coordination between the two systems limits the potential to exploit existing transport resources, resulting in redundant vehicle movements and elevated operational costs.

* Corresponding author.
E-mail address: S.Li-14@tudelft.nl (S. Li).

https://doi.org/10.1016/j.trc.2026.105996
Received 13 October 2025; Received in revised form 27 August 2026; Accepted 31 August 2026
Available online 7 September 2026

In response, integrating freight transport with passenger mobility systems has emerged as a promising strategy to improve urban transport efficiency. This approach is consistent with the European Commission's guidance that "local authorities need to consider all urban logistics related to passenger and freight transport together as a single logistics system" (European Commission, 2007). By utilizing a light passenger fleet to provide delivery services, logistics operators can mitigate access restrictions and reduce costs, while public transport providers can enhance vehicle utilization. Such a shared-use framework therefore offers a natural alignment between public objectives, including reducing traffic intensity and improving resource efficiency, and private incentives for cost-effective and reliable operations (Cleophas et al., 2019). However, realizing this concept in practice raises important challenges. Foremost among them is the preservation of passenger service quality, as highlighted by stakeholder analyses: integrating passenger and freight transport may introduce additional delays or in-vehicle congestion for passengers (Pogna et al., 2025). To encourage adoption, it is essential that passenger needs are prioritized and that service quality is not compromised (Savelsbergh and Van Woensel, 2016).

The multimodal nature of mobility and logistics systems makes railway stations a critical component within urban transportation networks, with many trips either originating or terminating at a railway station (International Union of Railways (UIC), 2019). In London alone, railway stations recorded on average more than one million passenger arrivals per day in 2024 (Department for Transport, 2025). This makes curbside capacity for vehicles feeding and distributing passengers and goods from train stations an unavoidable and challenging issue. First, multiple requests at the same station may require several vehicles to visit simultaneously, creating competition for limited curbside capacity. Second, heterogeneous time windows may necessitate repeated visits by the same vehicle, with each visit occupying the curbside for a certain duration. These factors render station curbside capacity a tightly coupled constraint that existing models cannot fully capture.

Although several studies have explored integrating passenger and freight flows in urban contexts within a Pickup and Delivery Problem (PDP) framework, they typically capture only a limited set of service indicators and overlook the critical role of railway stations. As a result, existing approaches provide limited value for many urban and regional mobility contexts, where passenger and freight service quality must be balanced while road–rail connections and station curbside constraints are explicitly considered.

To address these challenges, we develop a passenger-centric PDP with railway stations explicitly modeled as transfer hubs between road and rail. The fleet consists of environmentally friendly light vehicles that are assumed to comply with safety standards for integrated passenger-freight operations by design. The aim is to provide a more practical solution that can assist city authorities in fostering passenger-freight integration, while ensuring balanced service quality standards for both passengers and freight, and enhancing the efficiency of urban mobility systems. The contributions can be summarized as follows.

(1) We propose an integrated passenger-freight PDP with road-rail connections. A mixed-integer linear program (MILP) is developed that incorporates multiple service quality indicators and introduces a novel recursion-based approach for considering curbside capacity constraints. This local-to-aggregate coupling approach can be extended to settings in which multiple local node representations share a common facility-level capacity.
(2) We develop a matheuristic that decomposes the problem into a routing-oriented master problem and a scheduling-oriented subproblem solved to local optimality using a propagation-based approach. A schedule-shifting strategy is introduced to partially restore feasibility at the aggregate level.
(3) We design an enhanced Adaptive Large Neighborhood Search (eALNS) algorithm with problem-specific destroy operators exploiting marginal cost information from the subproblem, further strengthened by an adaptive cost-structure-driven selection scheme and Variable Neighborhood Search (VNS) intensification. This novel adaptive mechanism is broadly applicable to problems with multidimensional objectives, dynamically prioritizing operators that target the dominant cost components of the current solution.
(4) We provide insights that highlight the effectiveness of the proposed enhancement techniques and demonstrate the applicability and benefits of integrated transport in resource-constrained settings and across different passenger–freight demand compositions.

The remainder of this article is organized as follows. Section 2 reviews the relevant literature. Section 3 introduces the problem setting, and Section 4 presents its MILP formulation. Section 5 describes a decomposition-based matheuristic framework for its solution, while Section 6 reports the case study and numerical results. Finally, Section 7 concludes this study.

2. Literature review

Integrated transport of passengers and freight has long been applied in long-haul services to exploit shared-use infrastructure (Savelsbergh and Van Woensel, 2016). Examples include applications in the rail (Ursavas and Zhu, 2018; Li et al., 2023; Zhen et al., 2023), maritime (Bruzzone et al., 2021), and air transport (Zhang et al., 2004; Tang et al., 2008; Huang et al., 2024) sectors. More recently, the concept has been extended to urban contexts, where public transport systems are leveraged to accommodate last-mile deliveries by utilizing their residual capacity (Fatnassi et al., 2015; Delle Donne et al., 2023), with metro systems receiving particular attention. Related studies have addressed train service scheduling (Ozturk and Patrick, 2018; Behiri et al., 2018; Hörsting and Cleophas, 2023), train configuration (Di et al., 2022), train unit scheduling (Li et al., 2024), as well as multimodal integration involving metro-based freight deliveries (Stokkink and Geroliminis, 2025; Lu et al., 2026).

Beyond infrastructure-based integrated transport, a growing body of research has examined integrated passenger-freight transport using road-bound transport to provide flexible, tailored services. These problems are typically formulated as variants of the PDP, a classic extension of the Vehicle Routing Problem (VRP). For example, Ghilas et al. (2016) study a system where freight requests are transferred at public transport stations and subsequently carried by passenger services between stops. Regarding passenger service quality, they assume that passenger capacity is unaffected and thus exclude passenger demand from consideration. The problem is formulated as a PDP and solved using an Adaptive Large Neighborhood Search (ALNS) heuristic. Along similar lines, Machado et al.

(2023) analyze the integration of freight deliveries into bus networks from a fleet optimization perspective. Their objective is to select a minimum number of buses that strategically cover uncertain demand scenarios. Passenger demand is not explicitly modeled, as the integration is simplified by assigning freight to designated space in the vehicles. Extending the multi-tiered setting, Mandal and Archetti (2025) propose a three-tier delivery system in which public vehicles operate in the middle tier to transport freight between stops, while freight vehicles cover the first- and last-mile legs. Their focus lies on minimizing the routing costs of freight vehicles, and they develop a decomposition-based matheuristic framework with three variants, each variant emphasizing optimization in a different tier.

Building on the flexibility of shared vehicles, the idea of modularity has been recently introduced to enhance integrated passenger-freight transport. Hatzenbühler et al. (2023) extend road-based integration by introducing modular vehicles with platooning capabilities, formulating the problem as a PDP and solving it with an ALNS algorithm to minimize costs related to travel distance, travel time, fleet size, and unserved requests. In contrast to this on-demand setting, Lin and Zhang (2024) study scheduled services in a bidirectional corridor system with modular vehicles, where dispatch headways are decision variables and the objective balances operational costs, passenger waiting costs, and freight handling costs.

From the perspective of problem setting and modeling, most existing studies on integrated transport adopt the viewpoint of logistics service providers, where the primary objectives are operational cost minimization and freight demand satisfaction, thereby neglecting the implications for passenger services. In practice, however, the integration of freight with passengers raises important concerns: allocating vehicle capacity to freight without careful balance can lead to in-vehicle congestion, while detours and delays may further degrade passenger service quality (Pogna et al., 2025). Moreover, existing studies are largely unimodal, focusing exclusively on either road or rail system, thereby overlooking opportunities for road-rail integration, particularly in settings where railway stations act as natural hubs.

This study addresses these gaps by adopting a passenger-centric problem setting and by effectively integrating rail stations as transfer hubs. Specifically, we incorporate multiple service quality indicators, such as passenger pickup delay, detour delay, and in-vehicle congestion, to explicitly capture the trade-offs between passenger experience and overall system efficiency. In addition, we extend the modeling framework to include road-rail connections, thereby reflecting a key operational constraint often ignored in prior studies.

From the perspective of solution methodology, ALNS has been the predominant choice in studies formulated as variants of the PDP, including, but not limited to, integrated transport (e.g., Ghilas et al., 2016; Sampaio et al., 2020; Hatzenbühler et al., 2023; Özarık et al., 2023; Zamal et al., 2025). ALNS, first introduced by Ropke and Pisinger (2006), extends the Large Neighborhood Search (LNS) framework by adaptively selecting operators based on weights updated from historical performance, and has since become a widely used heuristic for VRP-type problems. Past studies have made methodological advancements in tailoring operators to the specific requirements of the problem.

Our solution approach builds upon the LNS framework but introduces a novel adaptive selection scheme that emphasizes improvements in the objective dimensions most critical to the current solution. The framework is further strengthened with a VNS intensification phase, yielding a new class of ALNS that adapts effectively to practical settings with complex, multidimensional objectives.

3. Problem description

We study a PDP in the context of integrated passenger–freight transportation, where the system comprises both road and rail modes. The planning task is to design static on-demand road services that support both passengers and freight in this road–rail intermodal system, while prioritizing passenger service quality. Here, “on-demand” refers to demand-responsive road services in which vehicles are routed to serve individual requests, in contrast to fixed-line public transport services. While many requests can be satisfied by a door-to-door service using the road system alone, a subset of requests requires multimodal transport that combines road and rail. In such cases, proper synchronization of road-based services with railway operations, particularly with respect to passenger transfer waiting times, is crucial in ensuring seamless mobility (Stokkink et al., 2025). Within this setting, we consider the design and optimization of the on-demand road services, where the rail component is treated as an integral but exogenous part of the system.

In this context, the train station is modeled as a key node, serving as both the origin and destination for a portion of passenger and freight requests. Road vehicles can access the train station to perform pickup and dropoff operations. Since train stations are typically urban hotspots where demand may peak in line with train schedules, curbside or parking capacity can become a scarce resource when vehicles serve train-related requests. Vehicle visits must therefore be coordinated to ensure smooth road–rail connections and mitigate passenger or freight congestion within the station.

Directly enforcing station parking capacity in continuous time would require checking simultaneous vehicle occupancy over all possible parking durations, leading to a computationally prohibitive number of checks. Enforcing capacity at every possible discrete timestamp would create a similarly excessive computational burden. An interval-based representation is therefore adopted in which the planning horizon is discretized into a set $\mathcal{T}$ of consecutive intervals of length τ^{p}, with a parking capacity limit κ^{p} enforced for each interval $t \in \mathcal{T}$. The interval length can be adjusted to balance modeling accuracy and computational complexity. Note that the vehicle routing and service schedule itself remains continuous-time, as is standard in PDPs. The discretized set $\mathcal{T}$ is used only for station-level parking evaluation: when a road vehicle visits the train station, its continuous arrival and departure times are associated with the corresponding parking intervals. This hybrid representation preserves continuous-time routing and scheduling while enabling the aggregate curbside capacity of the physical station to be enforced over discrete intervals.

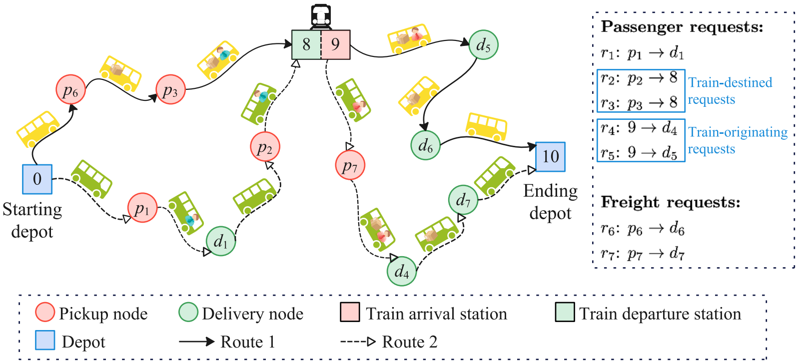


Fig. 1. Illustration of a toy network with two vehicles serving passengers and freight.

The railway services are assumed to operate according to a fixed and predetermined timetable with sufficient capacity to accommodate the demand. We further assume a generic headway of τ^{h} between successive train departures, which determines the transfer waiting times for train-destined passengers based on their station arrival times. These arrival and departure times at the station are determined by the road-based pickup-and-delivery plan.

Consistent with a static PDP setting, all request information is assumed to be known in advance of the optimization process. The set of requests is denoted by R, which consists of two disjoint subsets: passenger requests R^{p} and freight requests R^{f}. Passenger requests are treated as mandatory and must be served, whereas freight requests are modeled as soft requests that may be skipped at a penalty. This distinction reflects the passenger-priority setting considered in this study and is incorporated in the objective function through the skipped-freight penalty. Each request $r \in R$ is associated with a pickup node $p(r)$, a delivery node $d(r)$, and an earliest pickup time $\bar{\tau}_r$, which for passengers corresponds to their desired pickup time. In addition to purely road-based requests, a subset of requests either originates from or is destined to the train station. The set of requests destined for the train station is denoted by R^{+}, with $R^{\mathrm{p}+}$ and $R^{\mathrm{f}+}$ representing its passenger and freight subsets. Each request $r \in R$ has an associated demand volume ρ_r^{+}, measured either as the number of passengers or freight pallets. The pallets are assumed to be standardized, compact units, making them suitable for consolidation with passengers. For normalization, we assume that one pallet occupies the same vehicle capacity as one passenger. Alternative passenger-to-freight occupancy ratios can be represented by rescaling the freight demand volume or the effective freight capacity.

A fleet of homogeneous vehicles, denoted by K, can simultaneously transport passengers and freight. The usage of vehicle k incurs a fixed deployment cost of c_k^{v}. Each vehicle $k \in K$ has a certain capacity, denoted by κ^{v}, in terms of the total number of passengers and freight pallets that can be loaded. The vehicle capacity allocated to freight is limited to κ^{f} to respect the passenger comfort threshold. It is assumed that the demand volume of each request does not exceed κ^{v}. For any request whose demand exceeds this capacity, the corresponding pickup and delivery nodes are duplicated. Hence, the full demand at a node can be served by a single vehicle, eliminating the need for transshipment between vehicles.

The input network is represented by a directed graph $G = (N, A)$, where the node set N comprises regular customer nodes, train station nodes, and depots; A denotes the arcs. Each customer node corresponds to exactly one request, serving as either its pickup or delivery location, and neither the pickup nor the delivery is a station node. A station is represented by a set of duplicated nodes, denoted by N^{t}, to allow multiple visits by a vehicle. Each duplicate corresponds either to the pickup of a train-originating request or the delivery of a train-destined request. For each node $i \in N$, let ρ_i^{p} and ρ_i^{f} denote the net passenger and freight demand volumes, respectively, which are positive at the corresponding pickup nodes, negative at the corresponding delivery nodes, and zero otherwise. Since each request node is associated with either a passenger or a freight request, at most one of ρ_i^{p} and ρ_i^{f} is nonzero, and we define $\rho_i = \rho_i^{\mathrm{p}} + \rho_i^{\mathrm{f}}$ as the net demand volume at node i. Finally, we consider a single depot in the network, represented by two nodes: n^{ds} for vehicle departures and n^{de} for vehicle returns. Let $N^{\mathrm{d}} = \{n^{\mathrm{ds}}, n^{\mathrm{de}}\}$ denote the set of depot nodes. Considering the above setting, each vehicle is allowed to visit a node at most once.

A schematic representation of a network is provided in Fig. 1. The figure contains three regular road-based requests (r_1, r_6, and r_7), two train-destined requests (r_2 and r_3), and two train-originating requests (r_4 and r_5). Nodes 8 and 9 correspond to the same physical railway station, representing the train-departure and train-arrival station nodes, respectively. For train-destined requests r_2 and r_3, node 8 serves as the delivery node; for train-originating requests r_4 and r_5, node 9 serves as the pickup node. For simplicity, station copies for requests r_2–r_5 are not shown.

Each arc $(i, j) \in A$ is associated with a fixed travel time $\tau_{i,j}$ and a routing cost of $c_{i,j}^{\mathrm{r}}$ that is incurred by every vehicle traversing the arc. The travel times are assumed to satisfy the triangle inequality. For each pair of distinct duplicated station nodes $i, j \in N^{\mathrm{t}}$, arcs (i, j) and (j, i) are added with a small travel time $\tau_{i,j} = \varepsilon$, assumed to be negligible relative to the travel times between non-duplicated nodes. This discourages artificial cycles among duplicates while allowing for consecutive visits at the train station. At each node, the vehicle's dwell time is determined by the sum of boarding and alighting passengers and the amount of freight loaded and unloaded, scaled by the per-unit service time parameter τ^{d}.

Accordingly, the optimization balances vehicle deployment and routing costs with service-quality penalties for passengers and freight. Passenger service quality is captured through pickup delay, detour delay, transfer waiting time, and in-vehicle congestion.

Table 1
Sets and indices.

Notation	Description
K	Set of vehicles, indexed by k
R	Set of requests, indexed by r
$R^{\mathrm{p}}/R^{\mathrm{f}}$	Set of passenger/freight requests
$R^{\mathrm{p}+}$	Set of passenger requests destined to the train station
N	Set of all nodes, indexed by i or j
N^{t}	Set of train station nodes
N^{d}	Set of depot nodes, indexed by $n^{\mathrm{ds}}/n^{\mathrm{de}}$
A	Set of arcs, indexed by (i,j)
$\mathcal{T}$	Set of parking intervals, indexed by t

Freight service quality is captured through pickup and detour delays, while unserved freight demand incurs a skipped-request penalty. All penalties are evaluated in monetary cost units, allowing heterogeneous service-quality indicators to be combined in a single composite objective.

4. Optimization model

This section presents the mathematical formulation of the problem. Section 4.1 first introduces the notation. Subsequently, Section 4.2 describes the objective function, which balances vehicle utilization against the service quality provided to passengers and freight. Sections 4.3 to 4.9 present the mathematical formulation of seven groups of constraints. Finally, Section 4.10 introduces several valid inequalities that tighten the formulation.

4.1. Notation

A summary of the sets, indices, and parameters used in the model is provided in Tables 1 and 2. The variables are organized according to their modeling roles. Decision variables represent the routing and service plan or performance-related quantities used in the objective and constraints. Specifically, the model includes variables for vehicle routing, deployment, scheduling, and request-service decisions, which represent the core decisions of a PDP; load-tracking variables for enforcing vehicle capacity and evaluating in-vehicle congestion; time-related service-quality variables for measuring pickup delay, excessive delay, detour time, and transfer waiting time; and station-capacity variables for linking continuous station arrival/departure times to discretized parking intervals. Auxiliary variables are introduced only to linearize or simplify the formulation and can be eliminated without changing the underlying optimization problem. A complete description of the variables is given in Table 3.

The time dimension is captured by a hybrid continuous–discrete representation. The pickup-and-delivery schedule is modeled in continuous time, with $s_{i,k}$ denoting the arrival time of vehicle k at node i for scheduling and delay computation. Discrete variables are introduced only to map these continuous times to interval-based features: train-destined passenger arrivals are linked to the exogenous rail headway for transfer-waiting-time calculation, while station visits are linked to intervals in $\mathcal{T}$ for aggregate parking-capacity enforcement.

The big-M constants used in the formulation are constraint-specific and are distinguished by descriptive superscripts and, where applicable, subscripts. Their definitions and computations based on valid upper bounds of the corresponding variables are summarized in Appendix A.

4.2. Objective

The objective in Eq. (1) is to minimize the total cost, where the first two terms represent fixed vehicle and routing costs, and the remaining terms capture penalty costs associated with deteriorations in various dimensions of service quality. The penalty parameters convert service-quality indicators with different physical units into monetary cost units. They may be calibrated using operator preferences, service-level agreements, or value-of-time estimates, and may also be varied to reflect different passenger-priority policies.

The third and fourth terms correspond to passenger-related costs. In particular, the third term captures three types of delay time: pickup, detour, and transfer. The pickup delay represents the time a passenger spends waiting at the origin before being served, while an additional penalty is imposed when this delay exceeds a delay-time threshold, reflecting the increased discomfort experienced by passengers. The detour delay captures the additional in-vehicle travel time incurred due to indirect routing. The transfer delay applies to train-destined passengers, for whom the road-based service functions as a first-mile feeder to their rail journey, and corresponds to waiting time at the station. The parameters c^{wp}, $\hat{c}^{\mathrm{wp}}$, c^{dp}, and c^{t} denote the unit costs (per minute per passenger) of pickup delay, excessive pickup delay beyond the threshold, detour delay, and transfer delay, respectively.

The fourth term captures passenger discomfort resulting from traveling in a crowded vehicle. Both passengers and freight contribute to vehicle crowdedness, while only passengers incur a penalty for this effect. The parameter c^{c} denotes the penalty associated with this discomfort, measured per passenger per minute.

The last two terms correspond to freight-related costs. The fifth term minimizes the penalty cost of skipping freight requests where c^{s} denotes the penalty per pallet. The last term accounts for the penalty costs for pickup and detour delays, where the parameters c^{wf}

Table 2
Parameters.

Notation	Description
$c^{\mathrm{r}}_{i,j}$	Routing cost incurred for operating a vehicle on arc (i,j); proportional to the distance
c^{v}_{k}	Fixed cost for deploying vehicle k
$c^{\mathrm{wp}}/c^{\mathrm{wf}}$	Penalty cost per minute for late pickup of a passenger or freight pallet
$\hat{c}^{\mathrm{wp}}$	Additional penalty cost per minute for late pickup of a passenger beyond a threshold
$c^{\mathrm{dp}}/c^{\mathrm{df}}$	Penalty cost per minute of detour when delivering a passenger or freight pallet
c^{t}	Penalty cost per minute of transfer waiting time for a passenger
c^{c}	Discomfort penalty per minute for a passenger in a crowded vehicle
c^{s}	Penalty cost for skipping a freight pallet
$p(r)$	Pickup node index associated with request r
$d(r)$	Delivery node index associated with request r
ϱ^{+}_{r}	Positive integer representing the demand volume associated with request r
$\rho^{\mathrm{p}}_{i}/\rho^{\mathrm{f}}_{i}$	Net passenger/freight demand volume at node i: positive at the corresponding pickup nodes, negative at the corresponding delivery nodes, and 0 otherwise
ρ_{i}	Net demand volume at node i, defined as $\rho_i = \rho^{\mathrm{p}}_{i} + \rho^{\mathrm{f}}_{i}$
$\bar{\tau}_{r}$	Earliest pickup time associated with request r
θ^{c}	Threshold of in-vehicle congestion associated with passenger discomfort, defined as a fraction of vehicle capacity
θ^{d}	Threshold of delay time for picking up a passenger
$\tau_{i,j}$	Travel time on arc (i,j)
τ^{d}	Variable dwell time determined by boarding/alighting passengers and loading/unloading freight
τ^{h}	Train headway at the train station
τ^{p}	Duration of an interval during which parking capacity is enforced at the train station
κ^{v}	Carrying capacity of a vehicle
κ^{f}	Freight carrying capacity of a vehicle
κ^{p}	Parking capacity of the train station

Table 3
Variables.

Notation	Description
Decision variable	
Routing, deployment, scheduling, and request service	
$x_{i,j,k}$	Binary variable, equal to 1 if vehicle k traverses arc (i,j), 0 otherwise
v_{k}	Binary variable, equal to 1 if vehicle k is used, 0 otherwise
$s_{i,k}$	Continuous variable denoting the arrival time of vehicle k at node i
$y_{r,k}$	Binary variable, equal to 1 if request r is served by vehicle k, 0 otherwise
z_{r}	Binary variable, equal to 1 if freight request r is skipped, 0 otherwise
On-board load and congestion evaluation	
$g^{\mathrm{p}}_{i,k}/g^{\mathrm{f}}_{i,k}$	Integer variable denoting the number of on-board passengers/freight of vehicle k upon arrival at node i (before boarding/alighting)
$\tilde{g}^{\mathrm{p}}_{i,j,k}/\tilde{g}^{\mathrm{f}}_{i,j,k}$	Integer variable denoting the number of on-board passengers/freight when vehicle k traverses arc (i,j)
$w_{i,j,k}$	Binary variable, equal to 1 if the level of in-vehicle congestion on vehicle k exceeds the congestion-ratio threshold θ^{c} when traversing arc (i,j), 0 otherwise
$\varphi_{i,j,k}$	Integer variable denoting the number of on-board passengers on a congested arc (i,j) traversed by vehicle k
Time-related service indicators	
$l_{r,k}$	Continuous variable denoting the pickup delay time when vehicle k picks up request r
$\hat{l}_{r,k}$	Continuous variable denoting the pickup delay time exceeding the delay-time threshold θ^{d} when vehicle k picks up request r
$o_{r,k}$	Continuous variable denoting the additional detour time for request r picked up by vehicle k
$e_{r,k}$	Continuous variable denoting the transfer waiting time for train-destined passenger request r served by vehicle k
Parking capacity evaluation	
$\hat{q}^{\mathrm{a}}_{i,k}$	Integer variable denoting the index of the parking interval when vehicle k arrives at a train station node i
$\hat{q}^{\mathrm{d}}_{i,k}$	Integer variable denoting the index of the parking interval when vehicle k departs from a train station node i
$\hat{q}^{\mathrm{clear}}_{i,k}$	Integer variable denoting the index of the first-clear parking interval when vehicle k is no longer occupying a train station node i
$u^{\mathrm{a}}_{i,t,k}$	Binary variable, equal to 1 if vehicle k arrives at a train station node i in the t^{th} interval, 0 otherwise
$u^{\mathrm{clear}}_{i,t,k}$	Binary variable, equal to 1 if vehicle k departs from a train station node i in the $(t-1)^{\mathrm{th}}$ interval, 0 otherwise
$\sigma_{i,t,k}$	Binary variable, equal to 1 if vehicle k parks at a train station node i in the t^{th} interval, 0 otherwise
$\hat{\sigma}_{t,k}$	Binary variable, equal to 1 if vehicle k parks at the train station in the t^{th} interval, 0 otherwise
δ_{k}	Binary variable, equal to 1 if vehicle k visits the train station, 0 otherwise
Auxiliary variable	
$q_{r,k}$	Integer variable used to linearize the modulo operation with respect to the train headway interval for train-destined passenger request r served by vehicle k
$s^{\mathrm{d}}_{i,k}$	Continuous variable denoting the departure time of vehicle k from a train station node i, computed from its arrival time and service duration

and c^{df} denote the delay costs per pallet per minute.

$$\min Z = \underbrace{\sum_{k\in K} c^{\mathrm{v}}_k v_k}_{\text{Vehicle deployment}} + \underbrace{\sum_{k\in K}\sum_{i\in N}\sum_{j\in N} c^{\mathrm{r}}_{i,j}\, x_{i,j,k}}_{\text{Routing}}$$
$$+ \underbrace{\sum_{k\in K}\sum_{r\in R^{\mathrm{p}}} \varrho^{+}_r \left(c^{\mathrm{wp}}\, l_{r,k} + \hat{c}^{\mathrm{wp}}\, \hat{l}_{r,k} + c^{\mathrm{dp}}\, o_{r,k} + c^{\mathrm{t}}\, e_{r,k}\right)}_{\text{Passenger delay}} + \underbrace{\sum_{k\in K}\sum_{(i,j)\in A} c^{\mathrm{c}}\, \tau_{i,j}\, \varphi_{i,j,k}}_{\text{Passenger congestion}} \tag{1}$$
$$+ \underbrace{\sum_{r\in R^{\mathrm{f}}} c^{\mathrm{s}}\, \varrho^{+}_r\, z_r}_{\text{Skipped freight}} + \underbrace{\sum_{k\in K}\sum_{r\in R^{\mathrm{f}}} \varrho^{+}_r \left(c^{\mathrm{wf}}\, l_{r,k} + c^{\mathrm{df}}\, o_{r,k}\right)}_{\text{Freight delay}}.$$

4.3. Vehicle routing and scheduling constraints

Vehicle flow balance is enforced by Eqs. (2) to (4), which ensure that each vehicle, if used, departs from the starting depot n^{ds}, returns to the ending depot n^{de}, and visits each intermediate node exactly once along its route, respectively.

$$\sum_{j\in N\setminus N^{\mathrm{d}}} x_{i,j,k} = v_k \qquad \forall k\in K, i = n^{\mathrm{ds}}, \tag{2}$$
$$\sum_{j\in N\setminus N^{\mathrm{d}}} x_{j,i,k} = v_k \qquad \forall k\in K, i = n^{\mathrm{de}}, \tag{3}$$
$$\sum_{j\in N\setminus\{n^{\mathrm{ds}}\}} x_{i,j,k} = \sum_{j\in N\setminus\{n^{\mathrm{de}}\}} x_{j,i,k} \qquad \forall k\in K, \forall i\in N\setminus N^{\mathrm{d}}. \tag{4}$$

Three scheduling constraints are imposed to ensure temporal feasibility and request precedence. Eq. (5) guarantees that each request is picked up no earlier than its designated starting time. Eq. (6) enforces time propagation along each traversed arc. Finally, Eq. (7) explicitly enforces pickup-before-delivery precedence for each served request. When request r is assigned to vehicle k, its delivery time must be no earlier than its pickup time plus the pickup dwell time and the direct travel time between the corresponding pickup and delivery nodes.

$$s_{p(r),k} \geq \bar{\tau}_r \qquad \forall r\in R, \forall k\in K, \tag{5}$$
$$s_{j,k} \geq s_{i,k} + \tau_{i,j} + \tau^{\mathrm{d}}\,|\rho_i| - \mathrm{M}^{\mathrm{time}}_{i,j}\,(1 - x_{i,j,k}) \qquad \forall i\in N, \forall j\in N, \forall k\in K, \tag{6}$$
$$s_{d(r),k} \geq s_{p(r),k} + \tau_{p(r),d(r)} + \tau^{\mathrm{d}}\varrho^{+}_r - \mathrm{M}^{\mathrm{prec}}_r(1 - y_{r,k}) \qquad \forall r\in R,\ \forall k\in K. \tag{7}$$

4.4. Demand satisfaction constraints

From a passenger-centric perspective, Eqs. (8) and (9) ensure that each passenger request is satisfied by exactly one vehicle, whereas each freight request is either served or skipped.

$$\sum_{k\in K} y_{r,k} = 1 \qquad \forall r\in R^{\mathrm{p}}, \tag{8}$$
$$\sum_{k\in K} y_{r,k} + z_r = 1 \qquad \forall r\in R^{\mathrm{f}}. \tag{9}$$

Eqs. (10) and (11) couple the vehicle-routing and demand-assignment decisions, ensuring that the assigned vehicle visits both the corresponding pickup and delivery nodes. Depot nodes are excluded to prevent a vehicle from returning directly to the ending depot immediately after a pickup, or from visiting a delivery node directly after leaving the depot.

$$\sum_{j\in N\setminus N^{\mathrm{d}}} x_{p(r),j,k} = y_{r,k} \qquad \forall r\in R, \forall k\in K, \tag{10}$$
$$\sum_{i\in N\setminus N^{\mathrm{d}}} x_{i,d(r),k} = y_{r,k} \qquad \forall r\in R, \forall k\in K. \tag{11}$$

4.5. Vehicle capacity constraints

Eqs. (12) and (13) track the on-board load of each vehicle. Specifically, Eq. (12) maintains the number of passengers after visiting a node using the passenger-specific net demand ρ^{p}_i. Likewise, Eq. (13) accounts for the number of freight pallets on board using ρ^{f}_i.

$$g^{\mathrm{p}}_{j,k} \geq g^{\mathrm{p}}_{i,k} + \rho^{\mathrm{p}}_i - \mathrm{M}^{\mathrm{load,p}}_i(1 - x_{i,j,k}) \qquad \forall i\in N, \forall j\in N, \forall k\in K, \tag{12}$$
$$g^{\mathrm{f}}_{j,k} \geq g^{\mathrm{f}}_{i,k} + \rho^{\mathrm{f}}_i - \mathrm{M}^{\mathrm{load,f}}_i(1 - x_{i,j,k}) \qquad \forall i\in N, \forall j\in N, \forall k\in K. \tag{13}$$

To respect vehicle capacity, Eq. (14) ensures that the total on-board load never exceeds the vehicle's capacity. In addition, Eq. (15) restricts the maximum share of capacity that can be allocated to freight requests, thereby reserving space for passengers on board each of the vehicles.

$$g^{\mathrm{p}}_{i,k} + g^{\mathrm{f}}_{i,k} \le \kappa^{\mathrm{v}} \qquad \forall i \in N, \forall k \in K, \tag{14}$$

$$g^{\mathrm{f}}_{i,k} \le \kappa^{\mathrm{f}} \qquad \forall i \in N, \forall k \in K. \tag{15}$$

Eqs. (16) and (17) define an empty load state for each vehicle when departing from and returning to the depot.

$$g^{\mathrm{p}}_{n^{\mathrm{ds}},k} + g^{\mathrm{f}}_{n^{\mathrm{ds}},k} = 0 \qquad \forall k \in K, \tag{16}$$

$$g^{\mathrm{p}}_{n^{\mathrm{de}},k} + g^{\mathrm{f}}_{n^{\mathrm{de}},k} = 0 \qquad \forall k \in K. \tag{17}$$

4.6. In-vehicle congestion

In our setting, congestion penalties arise because passengers experience discomfort when vehicles become crowded, a condition driven by the combined presence of passengers and freight. This discomfort is captured by a congestion threshold θ^{c}, defined as a fraction of vehicle capacity. The threshold reflects the idea that passengers tolerate normal load levels but begin to perceive crowding once vehicle occupancy exceeds a critical utilization level. The threshold and the associated penalty parameter can be adjusted to capture varying perceptions of crowdedness in different operational contexts, allowing the model to flexibly represent different levels of passenger tolerance.

The in-vehicle congestion for passengers is captured through a set of constraints that link arc-based vehicle loads, congestion indicators, and the number of passengers exposed to congestion. Specifically, the following constraints define the number of on-board passengers and freight on each traversed arc, determine whether the total on-board load exceeds the congestion threshold, and use this condition to identify the number of passengers experiencing congestion discomfort.

Eqs. (18) and (19) transform the node-based on-board load into an arc-based representation. Specifically, the on-board passenger load on arc (i, j) for vehicle k is defined as the load after accounting for passenger boarding and alighting at node i, which corresponds to the load upon arrival at node j. Similarly, Eq. (19) defines the on-board load for freight.

$$\tilde{g}^{\mathrm{p}}_{i,j,k} \ge g^{\mathrm{p}}_{j,k} - \mathrm{M}^{\mathrm{arc,p}}(1 - x_{i,j,k}) \qquad \forall (i, j) \in A, \forall k \in K, \tag{18}$$

$$\tilde{g}^{\mathrm{f}}_{i,j,k} \ge g^{\mathrm{f}}_{j,k} - \mathrm{M}^{\mathrm{arc,f}}(1 - x_{i,j,k}) \qquad \forall (i, j) \in A, \forall k \in K. \tag{19}$$

After the on-board load on each arc is computed, Eq. (20) checks whether it exceeds the congestion threshold θ^{c}. Exceeding this threshold induces passenger discomfort, a condition captured through the binary indicator variable $w_{i,j,k}$. Alternatively, the formulation can be easily modified to capture congestion effects associated with the integration of freight, offering an additional perspective for scenarios in which passenger sensitivity is mainly triggered by the presence of freight in the same vehicle.

$$\mathrm{M}^{\mathrm{cong}}\, w_{i,j,k} \ge \tilde{g}^{\mathrm{p}}_{i,j,k} + \tilde{g}^{\mathrm{f}}_{i,j,k} - \theta^{\mathrm{c}}\, \kappa^{\mathrm{v}} \qquad \forall (i, j) \in A, \forall k \in K. \tag{20}$$

Finally, the number of passengers experiencing congestion on each congested arc, denoted by $\varphi_{i,j,k}$, is determined by Eqs. (21) and (22).

$$\varphi_{i,j,k} \le \mathrm{M}^{\mathrm{crowd}}\, w_{i,j,k} \qquad \forall (i, j) \in A, \forall k \in K, \tag{21}$$

$$\varphi_{i,j,k} \ge \tilde{g}^{\mathrm{p}}_{i,j,k} - \mathrm{M}^{\mathrm{crowd}}(1 - w_{i,j,k}) \qquad \forall (i, j) \in A, \forall k \in K. \tag{22}$$

4.7. Delay time calculation

This section describes the calculation of the three delay components: pickup, detour, and transfer.

Eq. (23) calculates the pickup delay for a passenger or freight request. For passenger requests, Eq. (24) computes the excess delay beyond the threshold θ^{d}. With nonnegativity of $\hat{l}_{r,k}$, it yields $\max\{l_{r,k} - \theta^{\mathrm{d}}, 0\}$.

$$l_{r,k} \ge s_{p(r),k} - \bar{\tau}_r - \mathrm{M}^{\mathrm{pick}}_r(1 - y_{r,k}) \qquad \forall r \in R, \forall k \in K, \tag{23}$$

$$\hat{l}_{r,k} \ge l_{r,k} - \theta^{\mathrm{d}} \qquad \forall r \in R^{\mathrm{p}}, \forall k \in K. \tag{24}$$

The detour time is measured as the difference between the actual arrival time at the delivery location and the theoretical earliest possible delivery time, which is determined by the actual pickup time. This is represented by Eq. (25).

$$o_{r,k} \ge s_{d(r),k} - \left(s_{p(r),k} + \tau_{p(r),d(r)} + \tau^{\mathrm{d}} \varrho^{+}_r\right) - \mathrm{M}^{\mathrm{det}}(1 - y_{r,k}) \qquad \forall r \in R, \forall k \in K. \tag{25}$$

The transfer waiting time is defined as the interval between a passenger's arrival at the train station and the departure of the next train, assuming trains depart with a constant headway τ^{h}. Eqs. (26) and (27) enforce this definition by linking the transfer waiting time $e_{r,k}$ with the discrete train departure schedule. In particular, $\tau^{\mathrm{h}} q_{r,k}$ denotes the departure time of the last train before the vehicle's arrival, whereas $\tau^{\mathrm{h}} q_{r,k} + \tau^{\mathrm{h}}$ corresponds to the departure time of the next train after the arrival. An example is shown in Fig. 2, where the last train departs at the end of interval 1 (i.e., $q_{r,k} = 1$), resulting in a transfer waiting time of 4.

$$e_{r,k} \ge \tau^{\mathrm{h}} q_{r,k} + \tau^{\mathrm{h}} - s_{d(r),k} - \mathrm{M}^{\mathrm{tra}}(1 - y_{r,k}) \qquad \forall r \in R^{\mathrm{p}+}, \forall k \in K, \tag{26}$$

$$e_{r,k} \le \tau^{\mathrm{h}} q_{r,k} + \tau^{\mathrm{h}} - s_{d(r),k} + \mathrm{M}^{\mathrm{tra}}(1 - y_{r,k}) \qquad \forall r \in R^{\mathrm{p}+}, \forall k \in K. \tag{27}$$

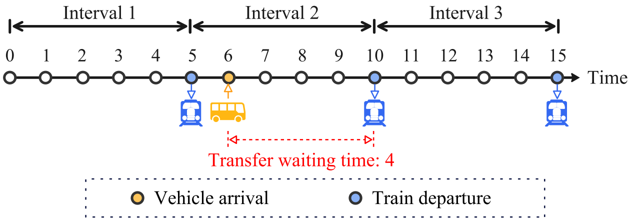


Fig. 2. Example of transfer waiting time calculation with a headway of $\tau^{\mathrm{h}} = 5$.

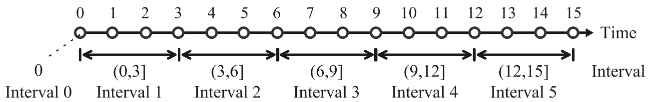


Fig. 3. Example of interval index calculation with an interval length of $\tau^{\mathrm{p}} = 3$.

4.8. *Train station parking capacity*

As described in Section 3, station parking capacity is enforced over discrete intervals rather than continuously or at every possible timestamp. Each station visit is mapped to the relevant intervals, where the aggregate capacity limit is enforced. We adopt a left-open, right-closed convention for defining the time range of each interval. Let τ^{p} denote the length of the intervals over which parking capacity is enforced. The t^{th} interval, $\forall t \geq 1$, is defined as $((t-1)\tau^{\mathrm{p}}, t\tau^{\mathrm{p}}]$. To distinguish vehicles that do not visit the train station from those that do, we introduce a dummy interval indexed by 0. Specifically, if a vehicle does not visit the train station, its arrival-time variable is set to zero, which in turn assigns it to parking interval 0. An example is provided in Fig. 3.

Parking capacity is defined at the aggregate level of the station, but to model routing decisions, we duplicate the station into request-specific copies. Each duplicated copy records the parking demand associated with a visit at the individual level. The following constraints link the arrival and departure times at duplicated station nodes, the corresponding occupied parking intervals, and the aggregate occupancy at the physical station. Capacity is then enforced at the aggregate station level by integrating the occupancy across all duplicated nodes. This construction also applies to settings in which multiple local node representations share a common facility-level capacity.

Define arrival and departure times. For each vehicle $k \in K$ and duplicated station node $i \in N^{\mathrm{t}}$, the arrival time is represented by the variable $s_{i,k}$. We further introduce an auxiliary variable $s^{\mathrm{d}}_{i,k}$, denoting the departure time from node i. This is calculated by Eq. (28), where $r(i)$ denotes the request index associated with node i.

$$s^{\mathrm{d}}_{i,k} = s_{i,k} + \tau^{\mathrm{d}} \left|\rho_i\right| y_{r(i),k} \qquad \forall i \in N^{\mathrm{t}}, \forall k \in K. \tag{28}$$

Calculate interval indices. The first and last intervals in which a vehicle parks are determined by its arrival and departure times. Accordingly, we introduce the integer variables $\hat{q}^{\mathrm{a}}_{i,k}$ and $\hat{q}^{\mathrm{d}}_{i,k}$, which denote the indices of the arrival and departure intervals at node i, respectively. They are defined by the following constraints:

$$\hat{q}^{\mathrm{a}}_{i,k} = \left\lceil \frac{s_{i,k}}{\tau^{\mathrm{p}}} \right\rceil \qquad \forall i \in N^{\mathrm{t}}, \forall k \in K,$$

$$\hat{q}^{\mathrm{d}}_{i,k} = \left\lceil \frac{s^{\mathrm{d}}_{i,k}}{\tau^{\mathrm{p}}} \right\rceil \qquad \forall i \in N^{\mathrm{t}}, \forall k \in K.$$

These two constraints can be linearized by:

$$\hat{q}^{\mathrm{a}}_{i,k}\tau^{\mathrm{p}} \geq s_{i,k} \qquad \forall i \in N^{\mathrm{t}}, \forall k \in K, \tag{29}$$

$$(\hat{q}^{\mathrm{a}}_{i,k} - 1)\tau^{\mathrm{p}} < s_{i,k} \qquad \forall i \in N^{\mathrm{t}}, \forall k \in K, \tag{30}$$

$$\hat{q}^{\mathrm{d}}_{i,k}\tau^{\mathrm{p}} \geq s^{\mathrm{d}}_{i,k} \qquad \forall i \in N^{\mathrm{t}}, \forall k \in K, \tag{31}$$

$$(\hat{q}^{\mathrm{d}}_{i,k} - 1)\tau^{\mathrm{p}} < s^{\mathrm{d}}_{i,k} \qquad \forall i \in N^{\mathrm{t}}, \forall k \in K. \tag{32}$$

Define arrival and departure interval indicators. We introduce a binary variable $u^{\mathrm{a}}_{i,t,k}$ to indicate whether vehicle k arrives and starts parking at node i during interval t. Eq. (33) links the integer-valued interval index to this binary indicator, while Eq. (34) ensures that exactly one interval is assigned.

$$\sum_{t \in \mathcal{T}} t\, u^{\mathrm{a}}_{i,t,k} = \hat{q}^{\mathrm{a}}_{i,k} \qquad \forall i \in N^{\mathrm{t}}, \forall k \in K, \tag{33}$$

$$\sum_{t \in \mathcal{T}} u^{\mathrm{a}}_{i,t,k} = 1 \qquad \forall i \in N^{\mathrm{t}}, \forall k \in K. \tag{34}$$

For the departure event, we introduce the concept of the *first-clear interval*, defined as the first interval in which the vehicle is no longer occupying the station. Its index is computed by Eq. (35), where the binary variable δ_k indicates whether vehicle k visits the train station, as defined in Eq. (36).

$$\hat{q}^{\text{clear}}_{i,k} = \hat{q}^{\text{d}}_{i,k} + \delta_k \qquad \forall i \in N^{\text{t}}, \forall k \in K, \tag{35}$$

$$\delta_k \geq x_{i,j,k} \qquad \forall i \in N^{\text{t}}, \forall j \in N, \forall k \in K. \tag{36}$$

Based on this computation, Eqs. (37) and (38) further define the binary indicator $u^{\text{clear}}_{i,t,k}$ for the first-clear interval.

$$\sum_{t \in \mathcal{T} \cup \{|\mathcal{T}|+1\}} t\, u^{\text{clear}}_{i,t,k} = \hat{q}^{\text{clear}}_{i,k} \qquad \forall i \in N^{\text{t}}, \forall k \in K, \tag{37}$$

$$\sum_{t \in \mathcal{T} \cup \{|\mathcal{T}|+1\}} u^{\text{clear}}_{i,t,k} = 1 \qquad \forall i \in N^{\text{t}}, \forall k \in K. \tag{38}$$

Recognize parking occupancy per interval. To identify all intervals during which a vehicle occupies a duplicated train station node, we apply a recursion-based calculation. A binary variable $\sigma_{i,t,k}$ is introduced to indicate whether vehicle k parks at station node i in interval t. Eq. (39) initializes the occupancy state, while Eq. (40) updates the occupancy recursively for subsequent intervals.

$$\sigma_{i,t,k} = 0 \qquad \forall i \in N^{\text{t}}, t = 0, \forall k \in K, \tag{39}$$

$$\sigma_{i,t,k} = \sigma_{i,t-1,k} + u^{\text{a}}_{i,t,k} - u^{\text{clear}}_{i,t,k} \qquad \forall i \in N^{\text{t}}, \forall t \in \mathcal{T} \setminus \{0\}, \forall k \in K. \tag{40}$$

To clarify the effect of the calculation, Fig. 4 illustrates how the constraints handle two occupancy cases: (i) when a vehicle enters and exits within the same interval, and (ii) when it enters in one interval and exits in another, resulting in occupancy across multiple intervals.

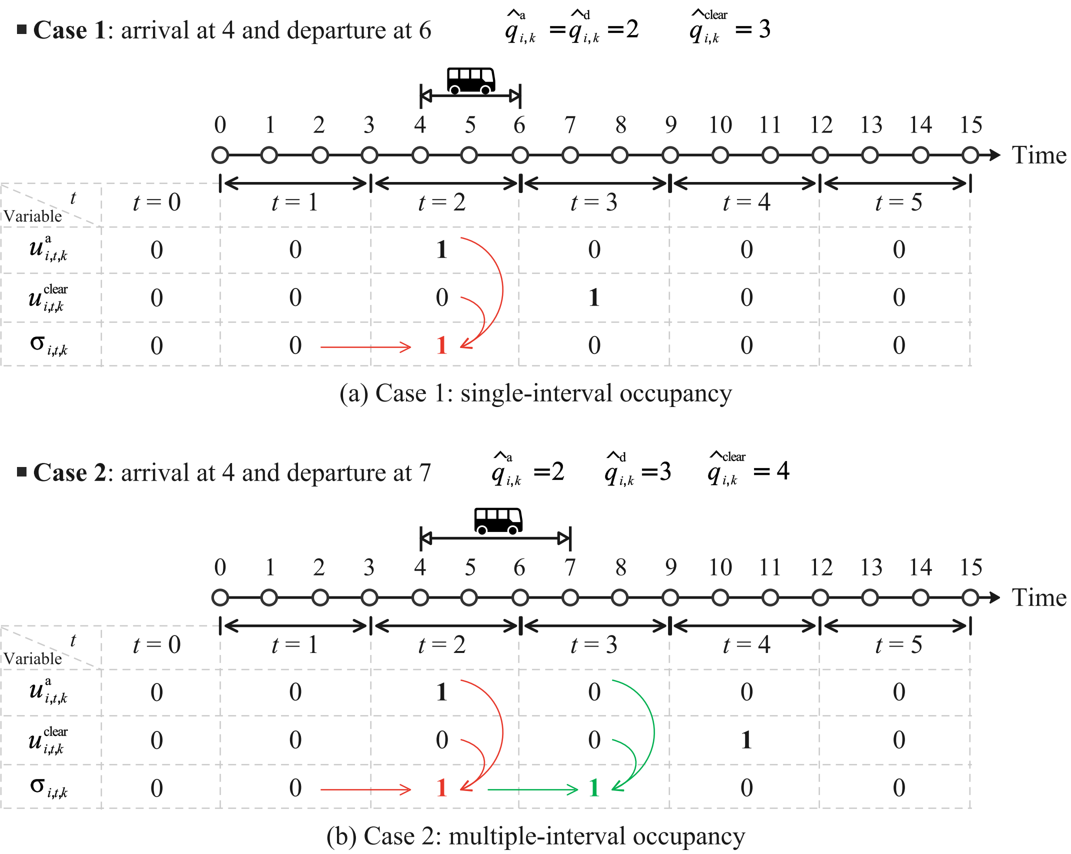


Fig. 4. Example of interval occupancy with $\tau^{\text{p}} = 3$.

Aggregate parking occupancy for duplicated nodes. A vehicle k is considered to be at the train station during interval t if and only if it occupies at least one of the duplicated nodes representing the station in that interval. This relationship is enforced by Eqs. (41) and (42), which determine the binary variable $\hat{\sigma}_{t,k}$ indicating whether vehicle k parks at the train station in interval t. Finally, Eq. (43) imposes the parking capacity at the train station, where the dummy interval with index 0 is excluded.

$$\hat{\sigma}_{t,k} \geq \sigma_{i,t,k} \qquad \forall i \in N^{\text{t}}, \forall t \in \mathcal{T}, \forall k \in K, \tag{41}$$

$$\hat{\sigma}_{t,k} \leq \sum_{i \in N^{\mathrm{t}}} \sigma_{i,t,k} \qquad \forall t \in \mathcal{T}, \forall k \in K, \tag{42}$$

$$\sum_{k \in K} \hat{\sigma}_{t,k} \leq \kappa^{\mathrm{p}} \qquad \forall t \in \mathcal{T} \setminus \{0\}. \tag{43}$$

4.9. Variable domains

Decision variables are defined by Eqs. (44)–(60). Eqs. (61) and (62) define the auxiliary variables.

$$x_{i,j,k} \in \{0,1\} \qquad \forall i \in N, \forall j \in N, \forall k \in K, \tag{44}$$

$$v_k \in \{0,1\} \qquad \forall k \in K, \tag{45}$$

$$s_{i,k} \geq 0 \qquad \forall i \in N, \forall k \in K, \tag{46}$$

$$y_{r,k} \in \{0,1\} \qquad \forall r \in R, \forall k \in K, \tag{47}$$

$$z_r \in \{0,1\} \qquad \forall r \in R^{\mathrm{f}}, \tag{48}$$

$$g^{\mathrm{p}}_{i,k}, g^{\mathrm{f}}_{i,k} \in \mathbb{N} \qquad \forall i \in N, \forall k \in K, \tag{49}$$

$$\tilde{g}^{\mathrm{p}}_{i,j,k}, \tilde{g}^{\mathrm{f}}_{i,j,k} \in \mathbb{N} \qquad \forall i \in N, \forall j \in N, \forall k \in K, \tag{50}$$

$$w_{i,j,k} \in \{0,1\} \qquad \forall i \in N, \forall j \in N, \forall k \in K, \tag{51}$$

$$\varphi_{i,j,k} \in \mathbb{N} \qquad \forall i \in N, \forall j \in N, \forall k \in K, \tag{52}$$

$$l_{r,k}, o_{r,k} \geq 0 \qquad \forall r \in R, \forall k \in K, \tag{53}$$

$$\hat{l}_{r,k} \geq 0 \qquad \forall r \in R^{\mathrm{p}}, \forall k \in K, \tag{54}$$

$$0 \leq e_{r,k} \leq \tau^{\mathrm{h}} \qquad \forall r \in R^{\mathrm{p}+}, \forall k \in K, \tag{55}$$

$$\hat{q}^{\mathrm{a}}_{i,k}, \hat{q}^{\mathrm{d}}_{i,k}, \hat{q}^{\mathrm{clear}}_{i,k} \in \mathbb{N} \qquad \forall i \in N^{\mathrm{t}}, \forall k \in K, \tag{56}$$

$$u^{\mathrm{a}}_{i,t,k}, \sigma_{i,t,k} \in \{0,1\} \qquad \forall i \in N^{\mathrm{t}}, \forall t \in \mathcal{T}, \forall k \in K, \tag{57}$$

$$u^{\mathrm{clear}}_{i,t,k} \in \{0,1\} \qquad \forall i \in N^{\mathrm{t}}, \forall t \in \mathcal{T} \cup \{|\mathcal{T}|+1\}, \forall k \in K, \tag{58}$$

$$\hat{\sigma}_{t,k} \in \{0,1\} \qquad \forall t \in \mathcal{T}, \forall k \in K, \tag{59}$$

$$\delta_k \in \{0,1\} \qquad \forall k \in K, \tag{60}$$

$$q_{r,k} \in \mathbb{N} \qquad \forall r \in R^{\mathrm{p}+}, \forall k \in K, \tag{61}$$

$$s^{\mathrm{d}}_{i,k} \geq 0 \qquad \forall i \in N^{\mathrm{t}}, \forall k \in K. \tag{62}$$

4.10. Valid inequalities

This section introduces several valid inequalities to tighten the LP relaxation of the model without changing the optimal objective value.

As passenger requests must all be served, at least one vehicle must be deployed; thus, Eq. (63) enforces the use of the first vehicle, indexed by k_0, by fixing its activation variable. A symmetry-breaking constraint represented by Eq. (64) is imposed to enforce an ordering among vehicles to eliminate symmetric solutions arising from permuting vehicle indices without affecting feasibility or cost.

$$v_{k_0} = 1, \tag{63}$$

$$v_k \geq v_{k+1} \qquad \forall k, k+1 \in K. \tag{64}$$

Eq. (65) links vehicle deployment to demand assignment, ensuring that a vehicle must be activated if a request is assigned to it. Eq. (66) stipulates that a vehicle remains idle if no request is assigned.

$$v_k \geq y_{r,k} \qquad \forall r \in R, \forall k \in K, \tag{65}$$

$$v_k \leq \sum_{r \in R} y_{r,k} \qquad \forall k \in K. \tag{66}$$

Eq. (67) prohibits vehicles from departing from the ending depot or returning to the starting depot, eliminating infeasible depot-related arcs. The direct reverse arc from the delivery node to its corresponding pickup node, which is infeasible under the pickup-before-delivery precedence constraint Eq. (7), is eliminated by Eq. (68).

$$\sum_{j \in N} x_{n^{\mathrm{de}},j,k} + \sum_{i \in N} x_{i,n^{\mathrm{ds}},k} = 0 \qquad \forall k \in K, \tag{67}$$

$$x_{d(r),p(r),k} = 0 \qquad \forall r \in R, \forall k \in K. \tag{68}$$

5. A decomposition-based matheuristic

This section presents the proposed decomposition-based matheuristic. Section 5.1 first analyzes model complexity and presents the framework. Subsequently, Section 5.2 introduces the master problem, which is solved using an eALNS heuristic. Section 5.3 describes a propagation-based approach to solve the subproblem. Section 5.4 details the simulated annealing (SA) acceptance criterion. Finally, Section 5.5 explains the feasibility and optimality checks where a schedule-shifting strategy is involved.

5.1. An overview of the decomposition framework

The proposed model introduces considerable solving complexity beyond the traditional PDPs. Specifically, the number of binary routing variables scales as $\mathcal{O}(|N|^2|K|)$. The inclusion of multiple service quality indicators for both passengers and freight further increases the problem size, particularly with respect to $|R|$ and $|K|$. Moreover, a global station parking capacity constraint contributes $\mathcal{O}(|N^{\mathrm{t}}||\mathcal{T}||K|)$ additional terms. This constraint couples all vehicle routes together, adding a high degree of interdependence and reducing the effectiveness of decomposition during optimization.

In addition to the increase in model size, the service quality constraints are defined based on threshold-triggered logical conditions, e.g., penalties applied only when certain indicators exceed predefined limits. Capturing these relationships requires additional binary variables and big-M constraints, which in turn lead to weaker LP relaxations. Moreover, since these penalties depend on aggregated values across entire routes, they disrupt the decomposability inherent in classical PDP formulations. As a result, the model not only becomes less compact but also exhibits poor lower-bound quality in a typical branch-and-cut-based solver, reducing its tractability for exact solution methods.

To address the computational challenges arising primarily from service quality constraints and route-coupling constraints, we develop a decomposition-based matheuristic, as outlined in Algorithm 1. In this approach, the master problem focuses on routing decisions, while the subproblem evaluates schedule-related components under fixed routes. This separation enables an exploratory search over routing plans, supports structured evaluation of the schedule-related costs induced by each fixed route, and facilitates parallel computation of the subproblems.

In particular, the master problem explores promising routing solutions using an eALNS, which integrates VNS with three ordered neighborhoods. Within this VNS, the search progressively shifts focus from broad improvements to nuanced schedule-targeted refinements, which reduces oscillations between competing cost dimensions while still allowing subsequent neighborhoods to enhance the solution. For each routing plan generated by the master problem, the subproblem determines a time-feasible schedule for each vehicle k, where the vehicle index also serves as the route index. These subproblems are independent and can be solved in parallel using a propagation-based approach. The resulting schedules are then aggregated and evaluated jointly: station parking capacity and vehicle fleet size are treated as feasibility conditions, with violations penalized through a soft penalty in the objective, while freight request skipping is captured through its associated penalty in the objective function, serving as part of the optimality evaluation. The solution is accepted based on an SA criterion.

In subsequent iterations, the schedule-related costs computed in the subproblems are fed back into the master problem, thereby guiding the search toward more promising regions of the solution space. Unlike traditional methods that explicitly compute the cost of each request's placement within the master problem, our decomposition bypasses this step by allowing the master problem to directly use these subproblem-derived costs, thereby reducing computational effort.

5.2. Master problem: An eALNS heuristic

In this section, we develop an eALNS heuristic to obtain a routing plan, where the main decision variables are the routing variable $x_{i,j,k}$ and the request assignment variable $y_{r,k}$. The master problem is handled heuristically because, even after separating the schedule-evaluation component, the routing layer remains a large combinatorial problem with route-coupling effects induced by shared fleet usage and station parking capacity. Solving a restricted master problem repeatedly as an MILP would require repeated branch-and-bound searches over routing variables and would not exploit the continuity between neighboring routing solutions. By contrast, eALNS modifies an incumbent routing plan incrementally, preserves useful route structures across iterations, and uses cost information returned by the subproblems to guide subsequent destroy and repair decisions.

Section 5.2.1 defines a set of destroy operators, each of which removes requests from the current solution according to a specific criterion. Section 5.2.2 presents the repair operators to reinsert the removed requests. Section 5.2.3 details the adaptive selection mechanism tailored to our problem. Finally, Section 5.2.4 proposes an intensification technique based on VNS.

5.2.1. Destroy operators

We implement 10 destroy operators to form the destroy operator set Ω^{d}. We categorize these operators into two families: *Shaw-removal-based operators* and *worst-removal-based operators*. The individual operators differ in both the number of requests they remove and the criteria by which requests or routes are selected.

Family 1: Shaw-removal-based operators. Proposed by Shaw (1998), the rationale of Shaw-type operators is to remove requests that exhibit a high degree of similarity, thereby increasing the potential for improved solutions when these requests are reinserted. Three operators are designed based on this idea, denoted by ϕ_1^{d}–ϕ_3^{d}.

Operator ϕ_1^{d} begins by selecting a served request uniformly at random to be removed as the seed for the removal process. All remaining requests are then evaluated using an overall relatedness measure, weighted by the three-element vector $\boldsymbol{\omega}^{\mathrm{shaw}}$, which

Algorithm 1 A decomposition-based matheuristic.

Initialize a routing solution $\boldsymbol{x}$
Set iteration index $i \leftarrow 0$
while termination criterion not met **do**
    $i \leftarrow i + 1$
    $\boldsymbol{\pi}_i \leftarrow$ UPDATEPROBABILITIES(cost shares) ▷ Section 5.2.3
    $\phi^{\mathrm{d}} \leftarrow$ ADAPTIVESELECT($\Omega^{\mathrm{d}}, \boldsymbol{\pi}_i$) ▷ Section 5.2.1
    $\phi^{\mathrm{r}} \leftarrow$ RANDOMSELECT(Ω^{r}) ▷ Section 5.2.2
    $\boldsymbol{x} \leftarrow \phi^{\mathrm{r}}\left(\phi^{\mathrm{d}}(\boldsymbol{x})\right)$
    for all routes $k \in K$ defined by $\boldsymbol{x}$ **in parallel do** ▷ Section 5.3
        Compute vehicle deployment, routing, and in-vehicle congestion costs
        Apply PROPAGATIONSCHEDULING($\boldsymbol{x}_k$) to generate a locally optimal schedule $\boldsymbol{s}_k$
        Compute schedule-related costs based on the schedule
    end for
    Evaluate candidate solution by feasibility and optimality checks ▷ Section 5.5
    if $i \bmod \mathcal{I}^{\mathrm{shift}} = 0$ **then**
        Apply SCHEDULESHIFTING($\boldsymbol{s}$)
    end if
    Apply SA acceptance criterion and update incumbent accordingly ▷ Section 5.4
    Update global best solution if an improvement is found
    if $\mathrm{i} \bmod \mathcal{I}^{\mathrm{polish}} = 0$ **then** ▷ Section 5.2.4
        Apply VNS to the global best solution
    else if Stagnates **and** Cooldown $\geq \mathcal{I}^{\mathrm{cool}}$ **then**
        Apply VNS to the incumbent solution
    end if
end while
Return best routing solution $\boldsymbol{x}$ with its schedule $\boldsymbol{s}$

captures spatial proximity, temporal alignment, and demand volume. Given a removal budget of $n^{\mathrm{remove}}_{\mathrm{shaw}}$, the operator selects the $n^{\mathrm{remove}}_{\mathrm{shaw}} - 1$ most related requests using a biased random mechanism (Ropke and Pisinger, 2006). This bias is controlled by the parameter β, which favors highly related requests to intensify the removal process, while still allowing occasional selection of less related requests to preserve diversification. Operators ϕ^{d}_2 and ϕ^{d}_3 adopt the same mechanism, but restrict the relatedness evaluation to spatial proximity and temporal alignment only, respectively.

Family 2: Worst-removal-based operators. Two categories of worst removal operators are defined. The first targets the removal of individual requests (operators ϕ^{d}_4–ϕ^{d}_7), where $n^{\mathrm{remove}}_{\mathrm{worst}}$ requests are removed in each iteration, provided that at least this number of requests remains. The second targets the removal of an entire route (operators ϕ^{d}_8–ϕ^{d}_{10}), thereby imposing a higher level of destruction on the current solution.

These worst-removal-based operators are tailored to the proposed decomposition-based algorithm. Instead of re-evaluating the objective function with and without each request or route, they exploit service-quality indicators that are computed and stored within the subproblem. Specifically, the marginal contributions of each request to the total, routing, and delay costs are assessed in the subproblem and passed to the master problem. These marginal costs serve as criteria to identify and remove poorly performing requests. Accordingly, operators ϕ^{d}_4–ϕ^{d}_6 remove requests based on marginal total, routing, and delay costs across all routes, respectively, while ϕ^{d}_7 selects a route at random and removes the requests with the highest marginal total cost. Similarly, operators ϕ^{d}_8–ϕ^{d}_{10} identify and remove the route with the worst total, routing, and delay costs, respectively.

5.2.2. Repair operators

We implement 4 repair operators to reinsert removed requests, forming the repair operator set Ω^{r}. We categorize these operators into two families: *regret-insertion-based operators* and *best-insertion-based operators*. For each unserved request, feasibility and insertion costs are evaluated over λ_2 randomly selected candidate routes, comprising all existing routes plus one new route consisting solely of this request. Within each candidate route, costs are further evaluated across λ_1 feasible pickup–delivery insertion positions, also selected at random. Each candidate position pair is ordered such that the pickup node is inserted before its corresponding delivery node, thereby preserving pickup-before-delivery precedence by construction.

Family 1: Regret-insertion-based operators. Operators ϕ^{r}_1 and ϕ^{r}_2 follow the general principle of regret insertion, where the reinsertion decision is based on comparing the insertion costs of a request across λ_1 locations. Operator ϕ^{r}_1 fully accounts for insertion costs across λ_2 candidate routes in each iteration, whereas operator ϕ^{r}_2 employs a more computationally efficient strategy by only maintaining the global best and second-best insertion costs.

Family 2: Best-insertion-based operators. Operators ϕ^{r}_3 and ϕ^{r}_4 follow the principle of selecting the insertion position that minimizes the incremental cost of reinserting a request. Operator ϕ^{r}_3 evaluates λ_1 insertion positions for each unserved request across existing routes, as well as the option of creating a new route. Operator ϕ_4^{r} extends the previous variant by also considering the option of skipping a request.

5.2.3. Tailored adaptive mechanism

When destroy operators are selected uniformly at random, as in a standard LNS approach, each operator is assigned the same probability regardless of the cost structure of the current solution. While such random selection promotes exploration, it may be inefficient in situations where costs associated with a particular dimension of solution quality dominate the objective value. In these cases, repeatedly selecting operators that have little impact on the prevailing cost structure can result in wasted iterations. To address this limitation, we design an adaptive selection mechanism. Unlike conventional score-based adaptive schemes driven by historical operator performance, our mechanism is tailored to the problem context and explicitly accounts for the inclusion of service-quality indicators defined across multiple performance dimensions. Since the destroy operators determine which parts of the incumbent solution are opened for modification and explicitly target different cost components, the proposed cost-structure-driven selection mechanism is applied to the destroy operators, while the repair operator is selected uniformly at random.

To guide the search process more effectively, the objective function is decomposed into cost groups so that the search can prioritize improving the dimension of solution quality that dominates the current objective. Specifically, we distinguish between (i) routing cost (the second term in the objective), (ii) schedule-related cost (penalties associated with passenger and freight delays), and (iii) all remaining cost components, which are aggregated into a third group.

An adaptive destroy-operator selection mechanism is then designed based on the Exponential Moving Average (EMA) of cost shares. Let C_i^{route}, C_i^{sched}, and C_i^{total} denote the routing cost, schedule-related cost, and total objective value in iteration i, respectively. The shares for the three cost groups in each iteration are computed as

$$s_i^{\mathrm{route}} = \frac{C_i^{\mathrm{route}}}{C_i^{\mathrm{total}}}, \quad s_i^{\mathrm{sched}} = \frac{C_i^{\mathrm{sched}}}{C_i^{\mathrm{total}}}, \quad s_i^{\mathrm{other}} = 1 - s_i^{\mathrm{route}} - s_i^{\mathrm{sched}}.$$

Let $\mathcal{G}$ represent the set of the three cost groups, and let $g \in \mathcal{G}$ denote a group index. The EMA value for cost group g in iteration i, denoted as $\hat{s}_i^g$, is initialized at $\frac{1}{3}$, providing a neutral starting point before adaptation. To avoid overreacting to fluctuations in a single iteration, the shares are updated over iterations using

$$\hat{s}_i^g = (1 - \alpha^{\mathrm{smooth}})\hat{s}_{i-1}^g + \alpha^{\mathrm{smooth}} s_i^g,$$

where $\alpha^{\mathrm{smooth}} \in (0, 1)$ is a smoothing factor.

Accordingly, the destroy operators are categorized into three groups: $\phi_2^{\mathrm{d}}, \phi_5^{\mathrm{d}}$, and ϕ_9^{d} are classified as route-target operators; $\phi_3^{\mathrm{d}}, \phi_6^{\mathrm{d}}$, and ϕ_{10}^{d} as schedule-target operators; while $\phi_1^{\mathrm{d}}, \phi_4^{\mathrm{d}}, \phi_7^{\mathrm{d}}$, and ϕ_8^{d} fall into the general-target group.

The EMA values $\hat{s}_i^g$ for all cost groups are converted into selection probabilities for the corresponding operator groups. To maintain diversity in the search, each probability is bounded below by a minimum threshold $\pi_{\min}$. The normalized probability for group g in iteration i, denoted as π_i^g, is thus given by

$$\pi_i^g = \frac{\max(\hat{s}_i^g, \pi_{\min})}{\sum_{g \in \mathcal{G}} \max(\hat{s}_i^g, \pi_{\min})},$$

where the minimum threshold is imposed inside the fraction to ensure a valid probability distribution. Thus, the adaptive mechanism prioritizes operators that are most relevant to the current structure of the objective function, while still ensuring that all operator groups remain available for exploration.

5.2.4. VNS-based intensification

While the ALNS framework explores promising solutions effectively, it relies on a fixed removal size, which consistently modifies the incumbent solution to a relatively large extent. When the incumbent already exhibits a good structure, such aggressive destruction may break it unnecessarily, whereas a better solution might be reached by only slightly perturbing the incumbent. To address this, we integrate a VNS procedure that applies targeted shaking within a smaller range, thereby complementing ALNS with a more balanced trade-off between diversification and intensification.

Targeting the three cost groups defined for the destroy operators and adaptive mechanism, three destroy operators are selected to guide the search toward the intended cost component: $\{\phi_4^{\mathrm{d}}, \phi_5^{\mathrm{d}}, \phi_6^{\mathrm{d}}\}$, targeting the total, routing, and delay costs, respectively. In each case, the number of removed requests is restricted to $n_{\mathrm{worst}}^{\mathrm{remove}} = 1$ to ensure only slight perturbations. A neighborhood is denoted by $\mathcal{N}(\phi_i^{\mathrm{d}}, \{\phi_1^{\mathrm{r}}, \phi_2^{\mathrm{r}}, \phi_3^{\mathrm{r}}, \phi_4^{\mathrm{r}}\})$, where a destroy operator ϕ_i^{d} is paired with the full set of repair operators. For each neighborhood, all four repairs are applied individually to the same destroyed solution, and the best reconstructed solution is retained.

The *total-cost-oriented neighborhood*, defined with destroy operator ϕ_4^{d}, is applied first and systematically seeks improvements at an overall level by jointly considering all cost components in the objective. In doing so, it explores improvements that arise from global trade-offs, serving as a coarse yet comprehensive screening step before any fine-grained adjustments. Next, the *route-oriented neighborhood*, using ϕ_5^{d}, refines the routing plan by prioritizing travel distance, which represents a noticeable portion of the objective and can otherwise undermine schedule quality if left inefficient. Finally, the *schedule-oriented neighborhood*, based on ϕ_6^{d}, performs a precise improvement to enhance time feasibility and reduce delays, which are central to passenger service quality. Placing this schedule-oriented step last is crucial because if it were applied first, its gains would likely be disturbed by subsequent route- or total-cost-focused adjustments.

This ordered progression moves from broad to increasingly focused neighborhoods, adapting the idea of VNS from expanding neighborhood size to a cost-component perspective. The three neighborhoods are applied sequentially so that broad cost efficiency is established first, routing plans are improved next, and schedule refinements are applied last, where they have the greatest impact on delay reduction. The VNS procedure is invoked under two conditions, each occurring after the SA acceptance step has updated the incumbent and global best solutions.

Condition 1: periodic polishing. Targeting the current global best solution, this periodic polishing step is performed every $\mathcal{I}^{\text{polish}}$ iterations, passing the global best solution to the VNS to identify potential improvements.

Condition 2: stagnation perturbation. When targeting the incumbent solution, a stagnation perturbation is invoked if two criteria are met simultaneously: (i) the global best solution has not been improved for $\mathcal{I}^{\text{stag}}$ consecutive iterations, and (ii) a cooldown period $\mathcal{I}^{\text{cool}}$ has elapsed since the last invocation of the VNS. The cooldown counter $\mathcal{I}^{\text{cool}}$ is measured as the difference between the current iteration and the iteration of the most recent VNS application, thereby preventing excessive use of the perturbation step.

5.3. Subproblem: propagation-based scheduling

The route set obtained from the master problem determines the values of the decision variables $x_{i,j,k}$ and $y_{r,k}$ for each route indexed by its assigned vehicle k. Based on these variables, we define $\bar{A}_k := \{(i,j) \in A \mid x_{i,j,k} = 1\}$ and $\bar{R}_k := \{r \in R \mid y_{r,k} = 1\}$ as the set of arcs used and the set of requests served by vehicle k, respectively. Using these inputs, all route-specific cost components can be systematically evaluated.

First, the *vehicle deployment and routing costs* for vehicle k are computed as $c^{\text{v}}_k + \sum_{(i,j)\in\bar{A}_k} c^{\text{r}}_{i,j}\, x_{i,j,k}$.

Second, the *vehicle congestion penalty* is derived from the load calculations defined in Eqs. (18) to (22). In particular, for each arc $(i,j) \in \bar{A}_k$, we update the on-board loads as

$$\tilde{g}^{\text{p}}_{i,j,k} = g^{\text{p}}_{j,k}, \quad \tilde{g}^{\text{f}}_{i,j,k} = g^{\text{f}}_{j,k},$$

and compute

$$\varphi_{i,j,k} = \begin{cases} \tilde{g}^{\text{p}}_{i,j,k}, & \text{if } \tilde{g}^{\text{p}}_{i,j,k} + \tilde{g}^{\text{f}}_{i,j,k} - \theta^{\text{c}}\kappa^{\text{v}} > 0, \\ 0, & \text{otherwise.} \end{cases}$$

The discomfort penalty for route k is then given by $\sum_{(i,j)\in\bar{A}_k} c^{\text{c}}\, \tau_{i,j}\, \varphi_{i,j,k}$.

Finally, the *schedule-related costs* involve the time dimension. For each route, we define a subproblem that can be decomposed over routes and further optimizes the schedule locally. The primary decision variable of this subproblem is $s_{i,k}$, with an output of a locally optimal schedule together with the associated delay penalties. The objective of the subproblem is to minimize the delay penalties for both passengers and freight:

$$\mathcal{Z}_{SP} = \min \sum_{k\in K} \sum_{r\in R^{\text{p}}} \varrho^{+}_r \left(c^{\text{wp}}\, l_{r,k} + \hat{c}^{\text{wp}}\, \hat{l}_{r,k} + c^{\text{dp}}\, o_{r,k} + c^{\text{t}}\, e_{r,k} \right) + \sum_{k\in K} \sum_{r\in R^{\text{f}}} \varrho^{+}_r \left(c^{\text{wf}}\, l_{r,k} + c^{\text{df}}\, o_{r,k} \right)$$

s.t. Eqs. (46), (53) to (61) and

$$s_{p(r),k} \geq \bar{\tau}_r \qquad \forall r \in \bar{R}_k, \forall k \in K, \tag{69}$$

$$s_{j,k} \geq s_{i,k} + \tau_{i,j} + \tau^{\text{d}}\, |\rho_i| \qquad \forall (i,j) \in \bar{A}_k, \forall k \in K, \tag{70}$$

$$s_{d(r),k} \geq s_{p(r),k} + \tau_{p(r),d(r)} + \tau^{\text{d}} \varrho^{+}_r \qquad \forall r \in \bar{R}_k, \forall k \in K, \tag{71}$$

$$l_{r,k} \geq s_{p(r),k} - \bar{\tau}_r \qquad \forall r \in \bar{R}_k, \forall k \in K, \tag{72}$$

$$\hat{l}_{r,k} \geq l_{r,k} - \theta^{\text{d}} \qquad \forall r \in \bar{R}_k \cap R^{\text{p}}, \forall k \in K, \tag{73}$$

$$o_{r,k} \geq s_{d(r),k} - \left(s_{p(r),k} + \tau_{p(r),d(r)} + \tau^{\text{d}} \varrho^{+}_r \right) \qquad \forall r \in \bar{R}_k, \forall k \in K, \tag{74}$$

$$q_{r,k} = \left\lceil \frac{s_{d(r),k}}{\tau^{\text{h}}} \right\rceil \qquad \forall r \in \bar{R}_k \cap R^{\text{p}+}, \forall k \in K, \tag{75}$$

$$e_{r,k} = \tau^{\text{h}}\, q_{r,k} - s_{d(r),k} \qquad \forall r \in \bar{R}_k \cap R^{\text{p}+}, \forall k \in K. \tag{76}$$

Instead of solving the MILP formulation directly, we propose a propagation-based approach, in which Eqs. (69) and (70) are replaced by the following propagation rule, while Eq. (71) is automatically satisfied through the accumulated predecessor-based propagation along each pickup-before-delivery feasible route:

$$s_{j,k} = \max \left\{ \bar{\tau}_{r(j)},\ s_{i,k} + \tau_{i,j} + \tau^{\text{d}} |\rho_i| \right\}, \quad \forall (i,j) \in \bar{A}_k,\ \forall k \in K. \tag{77}$$

For notational convenience, in Eq. (77), we set $\bar{\tau}_{r(j)} = 0$ when j is not a pickup node, corresponding to the beginning of the planning horizon. In this way, the arrival time at each node along a route is propagated by comparing its service time window with the departure time from its predecessor node. Once the vector of arrival times s is determined, the values of the remaining variables in the subproblem can be computed directly through Eqs. (72) to (76). Thus, the propagation approach solves the scheduling subproblem in linear time, requiring only a single forward pass over the nodes of a route.

Proposition 1 states that the proposed propagation procedure yields the locally optimal schedule for a given route, for a given set of parameters as defined in Lemma 1. We emphasize that the conditions indicated are sufficient for Proposition 1 to hold, but not necessary. At the same time, it also implies that global optimality cannot be guaranteed, as achieving this would require a joint optimization over all routes. Since the performance indicators are computed independently for each route, the resulting solution is locally optimal at the route level.

Lemma 1. *If $c^{\mathrm{t}} \le c^{\mathrm{dp}} \le c^{\mathrm{wp}}$ and $c^{\mathrm{df}} \le c^{\mathrm{wf}}$, then $\mathcal{Z}_{SP}(\mathbf{s}') \ge \mathcal{Z}_{SP}(\mathbf{s})$ for all feasible schedules $\mathbf{s}' \ge \mathbf{s}$.*

Proof. Given $\mathbf{s}' \ge \mathbf{s}$, we rewrite $\mathbf{s}' = \mathbf{s} + \boldsymbol{\epsilon}$ for which $\boldsymbol{\epsilon} \ge \mathbf{0}$.
For any fixed feasible schedule $\mathbf{s}$, Eq. (69) ensures that the right-hand side of Eq. (72) is nonnegative, while Eq. (71) ensures that the right-hand side of Eq. (74) is nonnegative. Since the corresponding delay variables enter the objective with nonnegative coefficients, their objective-minimizing values satisfy

$$l_{r,k} = s_{p(r),k} - \bar{\tau}_r \qquad \forall r \in \bar{R}_k, \forall k \in K, \tag{78}$$

$$o_{r,k} = s_{d(r),k} - \left(s_{p(r),k} + \tau_{p(r),d(r)} + \tau^{\mathrm{d}}\varrho_r^{+}\right) \qquad \forall r \in \bar{R}_k, \forall k \in K. \tag{79}$$

A piecewise comparison of every element in $\mathcal{Z}_{SP}(\mathbf{s})$ indicates:

$$l'_{r,k} - l_{r,k} = \epsilon_{p(r)}, \tag{80}$$

$$\hat{l}'_{r,k} - \hat{l}_{r,k} \ge 0, \tag{81}$$

$$o'_{r,k} - o_{r,k} = \epsilon_{d(r)} - \epsilon_{p(r)}, \tag{82}$$

$$e'_{r,k} - e_{r,k} \ge -\epsilon_{d(r)}. \tag{83}$$

Under $c^{\mathrm{t}} \le c^{\mathrm{dp}} \le c^{\mathrm{wp}}$ and $c^{\mathrm{df}} \le c^{\mathrm{wf}}$, substituting these (in)equalities into the passenger- and freight-related terms of $\mathcal{Z}_{SP}$ yields $\mathcal{Z}_{SP}(\mathbf{s}) - \mathcal{Z}_{SP}(\mathbf{s}') \le 0$. □

Proposition 1. *If the conditions in Lemma 1 are satisfied, the locally optimal schedule of a given route can be obtained by a single forward propagation procedure using Eq. (77).*

Proof. (a) Feasibility. By construction, Eq. (77) ensures that Eqs. (69) and (70) are satisfied. Moreover, since each fixed route preserves pickup-before-delivery order by construction and the travel times satisfy the triangle inequality, repeated application of Eq. (70) from $p(r)$ to $d(r)$ guarantees Eq. (71): the accumulated travel time is no smaller than $\tau_{p(r),d(r)}$, while the pickup dwell time $\tau^{\mathrm{d}}\varrho_r^{+}$ is included in the propagation. The remaining constraints are computations that do not influence the feasibility of the solution, but rather compute the objective value. This implies that any solution $\mathbf{s}$ obtained by iteratively applying Eq. (77) along the fixed route is feasible.
(b) Optimality. Suppose that there exists a feasible schedule $\mathbf{s}'$ with $s'_{j,k} < s_{j,k}$ for some node j, and let j be the first such node along the fixed route. Then either Eq. (69) or Eq. (70) must be violated.
Case I: Let $s_{j,k} = \bar{\tau}_{r(j)}$. This case applies only when j is a pickup node; for a non-pickup node, the first term is zero and the predecessor-based bound applies. Hence, if $s'_{j,k} < s_{j,k}$, Eq. (69) is violated.
Case II: Let $s_{j,k} = s_{i,k} + \tau_{i,j} + \tau^{\mathrm{d}}|\rho_i|$. Since j is the first node along the fixed route for which $s'_{j,k} < s_{j,k}$, all preceding nodes occur at the same or a later time in $\mathbf{s}'$ than in $\mathbf{s}$; in particular, $s'_{i,k} \ge s_{i,k}$. Hence, feasibility requires $s'_{j,k} \ge s'_{i,k} + \tau_{i,j} + \tau^{\mathrm{d}}|\rho_i| \ge s_{j,k}$, yielding a contradiction.
Hence, for any feasible schedule it must hold that $\mathbf{s}' \ge \mathbf{s}$. Then, according to Lemma 1, $\mathcal{Z}_{SP}(\mathbf{s}') \ge \mathcal{Z}_{SP}(\mathbf{s})$, suggesting that the cost associated with any feasible solution $\mathbf{s}'$ is at least as high as the cost of solution $\mathbf{s}$.
□

This result also explains why the subproblem is not solved by a generic MILP solver within the proposed matheuristic. For a fixed route, the propagation procedure exploits the sequential structure of the scheduling problem and obtains the locally optimal route-level schedule under the conditions of Lemma 1. Replacing this specialized procedure with a generic MILP solver would therefore introduce additional computational overhead without improving the route-level schedule obtained under these conditions.

5.4. Solution acceptance

The overall algorithm starts with an unserved request list containing all requests and applies operator ϕ_3^{r} to construct an initial route solution. This initial solution is then passed to the subproblem for evaluation of the objective. The solution quality in each iteration is evaluated based on its overall objective value, which consists of the total cost of all routes together with soft penalties introduced by the checks detailed in Section 5.5.

An SA criterion is employed to determine whether a new solution is accepted. Let Z_i^{new}, Z_i^{old}, and $\mathcal{T}_i$ denote the objective value of the new solution, the objective value of the incumbent solution, and the temperature at iteration i, respectively. A worse solution is accepted with probability $\exp\left(-\frac{Z_i^{\mathrm{new}} - Z_i^{\mathrm{old}}}{\mathcal{T}_i}\right)$.

Two mechanisms regulate the temperature: reheating and holding. The temperature starts at $\mathcal{T}^{\mathrm{start}}$ and decreases geometrically with cooling factor α_1^{cool} until the reheating threshold $\mathcal{I}^{\mathrm{reheat}}$ is reached. Once this threshold is reached, the temperature is reheated to a higher level $\mathcal{T}^{\mathrm{reheat}}$ (Barrena et al., 2014; Santini et al., 2018). For the subsequent $\mathcal{I}^{\mathrm{hold}}$ iterations, the temperature is maintained at this level before decreasing again with a different cooling factor α_2^{cool} until reaching the terminal temperature $\mathcal{T}^{\mathrm{end}}$. This reheating–holding mechanism increases the likelihood of accepting diverse solutions, thereby enhancing exploration and reducing the risk of premature convergence. The algorithm terminates when reaching the maximum number of iterations $\mathcal{I}^{\mathrm{max}}$.

5.5. Solution evaluation and schedule-shifting strategy

In each iteration, the eALNS is invoked to solve the master problem, with the operators guided by information obtained from the subproblem in the preceding iteration. The resulting routes are passed to the subproblem for evaluation at the route level. At an aggregate level, the overall solution quality is assessed through two checks: optimality and feasibility.

Optimality check. The optimality check addresses unserved freight requests. If any requests remain unserved, a penalty term $\sum_{r\in R^{\mathrm{f}}} c^{\mathrm{s}}\,\rho_r^{+}\,z_r$ is added to the objective.

Feasibility check 1: Fleet size. The first feasibility check concerns the fleet size. Specifically, if the number of routes, denoted by $|\mathcal{R}|$, exceeds the designated fleet size, $|K|$, a soft penalty $\eta^{\mathrm{fleet}}(|\mathcal{R}|-|K|)$ is imposed, where η^{fleet} denotes the unit penalty for each vehicle exceeding the available fleet size.

Feasibility check 2: Parking capacity. The second feasibility check concerns parking capacity. For each route, the parking constraints Eqs. (41) to (43) are verified based on the arrival times $\boldsymbol{s}$ obtained from the subproblem, and any violation incurs a soft penalty of $\eta^{\mathrm{park}}\big(\sum_{k\in K}\hat{\sigma}_{t,k}-\kappa^{\mathrm{p}}\big)$, where η^{park} denotes the unit penalty for exceeding capacity by one vehicle. This check is enabled by the information provided by the subproblem, which determines the arrival and departure times at train stations for each visit and generates all parking occupancy intervals according to Eqs. (29)–(40).

Feasibility improvement: Schedule-shifting strategy. A drawback of solely relying on soft penalties for parking capacity violations is that it becomes difficult to obtain feasible solutions under strict capacity limits. This is because the subproblem generates only locally optimal schedules for individual routes, without accounting for the aggregate parking constraint across all routes. As a result, simply penalizing violations may trap the search in infeasible regions. To address this issue, we introduce a schedule-shifting strategy to partially restore feasibility.

This strategy is applied every $\mathcal{I}^{\mathrm{shift}}$ iterations, at which point the schedules of a selected set of routes are delayed in order to reduce the maximum violation interval. Two main decisions are involved: (i) identifying the routes whose schedules will be shifted, and (ii) determining the shifted value for each affected node along the selected routes. The detailed procedure is described in the following steps.

Step 1: Identification of the most overloaded interval. Let $Q_t := \sum_{k\in K}\hat{\sigma}_{t,k}$ denote the number of vehicles occupying parking slots in interval t. The most overloaded interval is identified as $t^{\mathrm{worst}} := \arg\max_{t\in\mathcal{T}} Q_t$. In each implementation of the shifting procedure, we focus on this interval and its nearest subsequent available interval, which is given by

$$t^{\mathrm{new}} := \min\left\{\, t\in\mathcal{T} \;\middle|\; t > t^{\mathrm{worst}},\ Q_t < \kappa^{\mathrm{p}} \,\right\}.$$

Step 2: Shifting selected routes. To restore feasibility at t^{worst}, $Q_{t^{\mathrm{worst}}}-\kappa^{\mathrm{p}}$ routes are selected and their schedules are shifted. For each route parking at a duplicated station node during t^{worst}, the minimum delay required to move it into t^{new} is given by the difference between the starting time of the nearest next available interval, $(t^{\mathrm{new}}-1)\tau^{\mathrm{p}}+\epsilon$, and the starting time of the corresponding parking operation of vehicle k, where ϵ is a small positive value introduced to ensure that the shifted operation falls strictly outside of interval t^{worst}.

The schedule for nodes prior to this station node remains unchanged, while from this station node onward, it is re-propagated using Eq. (77), which defines two bounding conditions: the arrival time at a node may be bounded by the departure time of its predecessor or by its own time window. In the first case, a uniform shift Δs_k can be applied to all subsequent nodes. In the second case, the shift is node-specific and denoted by $\Delta s_{i,k}$, where i indexes a node whose arrival time is affected. Fig. 5 illustrates these two situations. After evaluating the shift for each route contributing to the most overloaded interval t^{worst}, the cost of shifting route k is measured by its incremental schedule-related cost $\Delta\mathcal{Z}_{SP,k}$. The $Q_{t^{\mathrm{worst}}}-\kappa^{\mathrm{p}}$ routes with the smallest increments are then selected, and the corresponding shifts are applied.

Pseudocode of the schedule-shifting procedure is presented in Algorithm 2. After shifting a partial schedule out of t^{worst}, we do not verify whether the updated schedule enters another overloaded interval, as checking and resolving all violations would be computationally expensive. Consequently, the shifted solution may remain infeasible due to residual violations in other intervals. The proposed shifting strategy therefore serves as a targeted repair mechanism that improves feasibility at the aggregate level, after which any remaining violations are penalized through soft penalties.

6. Model application and results

This section presents the results of numerical experiments. Section 6.1 introduces the case study and evaluates the algorithm in terms of performance and scalability. Section 6.2 examines the effect of two algorithmic enhancement techniques. Section 6.3 reports the sensitivity analyses on fleet size, parking capacity, vehicle capacity, and vehicle deployment cost. Finally, Section 6.4 analyzes the benefits and applicability of an integrated transport service relative to separated transport solutions.

6.1. Baseline setup and performance evaluation

We evaluate the performance of our approach on a case study based in the city center of Stockholm, Sweden. The dataset introduced by Hatzenbühler et al. (2023) is adopted, where passenger requests reflect expected daily passenger movements in Stockholm, freight requests are evenly distributed, and time windows follow a peak-hour distribution. The instance considered consists of 80 requests, evenly split between passengers and freight. For both passenger and freight requests, the same distribution is applied across regular, train-destined, and train-originating requests, following a ratio of 18:11:11, yielding 40 requests in total for each group. The

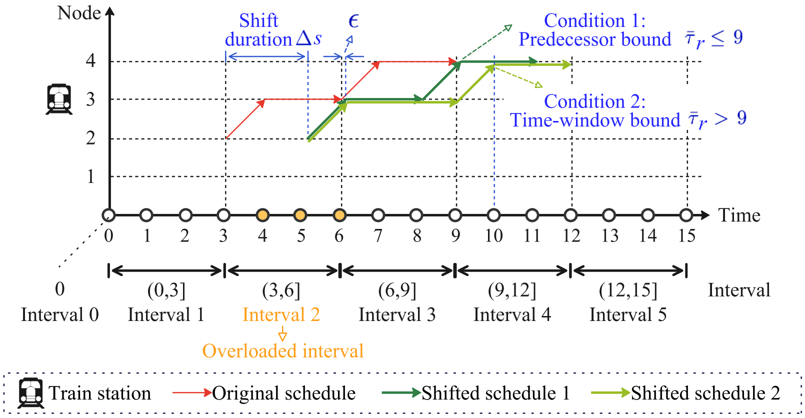


Fig. 5. Shift duration at train station for a partial route $2 \rightarrow 3 \rightarrow 4$ with $\tau^{\mathrm{p}} = 3$. Shifted schedule 1 is used under condition 1, where the earliest pickup time at node 4 is no greater than 9 and the predecessor constraint is binding; otherwise, shifted schedule 2 is used under condition 2, where the time-window constraint is binding.

Algorithm 2 Schedule-shifting procedure.

Identify the most overloaded interval $t^{\mathrm{worst}} := \arg\max_{t \in \mathcal{T}} Q_t$
Identify the first non-violating interval after t^{worst}, denoted by t^{new}
Identify all violating routes parking in t^{worst}
Calculate the minimum number of routes to shift $\lambda \leftarrow Q_{t^{\mathrm{worst}}} - \kappa^{\mathrm{p}}$
for each violating route k **do**
  Identify duplicated station nodes where route k parks in t^{worst}
  Obtain the arrival times at these duplicated station nodes
  Find the earliest arrival time s_k^{early} and the corresponding duplicated station node i_k^{early}
  Compute the shift duration at node i_k^{early} by $\Delta s_{i,k} \leftarrow (t^{\mathrm{new}} - 1)\,\tau^{\mathrm{p}} + \epsilon - s_k^{\mathrm{early}}$
  Apply $\Delta s_{i,k}$ to the station node i_k^{early}
  Re-propagate arrival and departure times for subsequent nodes by Eq. (77)
  Update the parking intervals for affected station nodes
  Evaluate incremental schedule-related cost of shifting route k: $\Delta \mathcal{Z}_{SP,k}$
end for
Select λ violating routes with the smallest $\Delta \mathcal{Z}_{SP,k}$
Replace the old schedules of the selected routes with their shifted schedules
Apply soft penalties $\eta^{\mathrm{park}}\left(\sum_{k \in K} \hat{\sigma}_{t,k} - \kappa^{\mathrm{p}}\right)$ if violations exist

underlying network comprises 120 nodes, including a train station and a depot, as shown in Fig. 6. Pairwise distances between nodes are derived from road network data provided by OpenStreetMap contributors (2017), while travel times are obtained by assuming a vehicle speed of 10 km/h. The variable dwell time per request unit τ^{d} is set to 0.2 min. The train headway τ^{h} is set to 15 min, and the parking interval duration τ^{p} to 5 min, with a parking capacity of $\kappa^{\mathrm{p}} = 5$ vehicles per interval. The fleet consists of 14 vehicles, each with capacity $\kappa^{\mathrm{v}} = 15$, of which the freight share is $\kappa^{\mathrm{f}} = 0.5\kappa^{\mathrm{v}}$.

The routing cost per kilometer on an arc $c_{i,j}^{\mathrm{r}}$ is set to 1, and the vehicle deployment cost c_k^{v} to 10. Passenger-related penalty costs are defined as follows: late pickup $c^{\mathrm{wp}} = 1.5$ per minute, late pickup beyond the delay threshold $\hat{c}^{\mathrm{wp}} = 2.25$ per minute, detour $c^{\mathrm{dp}} = 1$ per minute, and transfer $c^{\mathrm{t}} = 0.5$. The in-vehicle congestion penalty is set to $c^{\mathrm{c}} = 2$ per minute. Two threshold parameters are included: the pickup delay threshold $\theta^{\mathrm{d}} = 5$ minutes and the congestion threshold $\theta^{\mathrm{c}} = 0.4$, adopted from Liu et al. (2023). Freight-related penalty costs are as follows: skipped request $c^{\mathrm{s}} = 10$, late pickup $c^{\mathrm{wf}} = 0.2$ per minute, and detour $c^{\mathrm{df}} = 0.05$ per minute. All cost coefficients are tuned to balance the objectives; they do not correspond to actual monetary units.

Algorithmic parameters and hyperparameters are summarized in Appendix B. Unless otherwise specified, the proposed decomposition-based matheuristic is executed five times for each scenario to account for stochasticity. Throughout the results section, the proposed algorithm is denoted by “eALNS”, highlighting the core heuristic component around which the method is built.

To evaluate the performance of the proposed algorithm against Gurobi and its scalability across different problem sizes, small- and medium-scale instances are extracted from the large instance. The fleet size is set to 4 for both the small and medium instances, with a vehicle carrying capacity of 12 and a vehicle deployment cost of 5, while the remaining parameters are identical to those of the large instance. To provide a more comprehensive comparison, we consider three base cases corresponding to distributed, centralized, and clustered spatial demand distributions, which are further analyzed in Section 6.4. The distributed case corresponds to the baseline

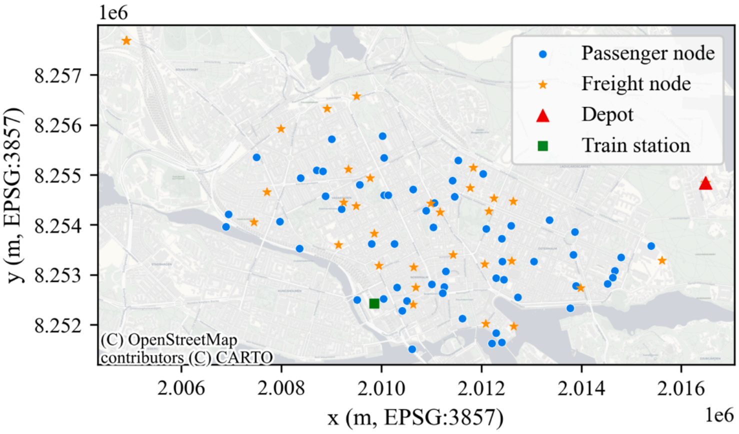


Fig. 6. Spatial distribution of the study area in Stockholm.

setting, where requests are widely spread over the service area with substantial overlap between passenger and freight demand; the centralized case concentrates requests around the city center; and the clustered case groups requests into several local clusters.

Starting from each large-scale base case, 10 variant cases of the same size are generated by re-pairing pickup and delivery nodes separately within the passenger and freight pools, while preserving the request-type composition of the base case. For each large case, a corresponding medium case is sampled as a subset of complete requests, and a corresponding small case is subsequently sampled from the medium case. This procedure yields 30 cases for each instance size. Because the re-pairing disrupts the structured origin–destination relationships of the base layouts, the variant cases are used only for the computational benchmark reported here, whereas the remaining experiments in Section 6 use the base cases, which preserve the original origin–destination structure.

Gurobi is given a time limit of 30 min for the small instances, while a more permissive limit of 60 min is used for the medium and large instances. For the large instances, memory-saving parameters are additionally applied. The proposed eALNS is run five times for each case, and the reported results are averaged over these independent runs. As the case-level results exhibit the same qualitative performance pattern across the tested instances, we present the grouped comparison in Table 4 to highlight the overall differences in solution quality, runtime, and scalability between the two approaches. The complete case-level results are provided in Appendix C.

The results indicate a clear scalability threshold for solving the full MILP directly. For small instances, Gurobi proves optimality for all cases, while eALNS obtains comparable objective values within 5 s. For medium instances, Gurobi finds feasible incumbents for all cases but retains large optimality gaps after the 60-min limit. In contrast, eALNS obtains lower objective values within substantially shorter computation times, typically around one minute. For large instances, Gurobi fails to produce a feasible incumbent because of memory limitations or expiration of the time limit, whereas eALNS consistently returns feasible solutions in approximately 14–18 min.

These results show that directly solving the complete MILP is effective only for small instances. For medium instances, Gurobi can still provide feasible incumbents but is no longer competitive within the tested time limit. For large instances, the full MILP becomes computationally prohibitive, making the proposed eALNS necessary for obtaining feasible solutions at practical scales. The comparison therefore identifies the transition from the optimality guarantees of the exact solver on small instances to the scalability advantage of eALNS on medium and large instances.

6.2. *Effect of algorithm enhancements*

This section uses the basic LNS algorithm as a baseline and examines the effect of two enhancements: an adaptive selection mechanism and VNS-based intensification. These enhancements yield three variants: LNS with adaptive selection (ALNS), LNS with VNS intensification (eLNS), and LNS with both enhancements (eALNS). The baseline scenario of the large instance is solved ten times with each of the four approaches, and their performance in terms of solution quality and computation time is summarized in Fig. 7.

(1) Solution quality. The eALNS delivers robust solutions, with a low median objective value and a narrow spread across the ten runs, demonstrating robustness and reliability. Although eLNS attains a slightly lower median, it exhibits greater variability. ALNS shows a higher median and the largest variability, though it has a greater likelihood of achieving better solutions than the baseline LNS. By contrast, LNS has the highest median and a wide spread, indicating weaker overall performance. This observation is further supported by Fig. 8, which reports the median performance over 10 independent runs (with different random seeds), together with the interquartile range (IQR, 25^{th}-75^{th} percentiles) as a measure of variability.

(2) Efficiency. Compared with LNS, all three enhanced variants require slightly more computation time to converge. Among them, eALNS has a marginally lower median runtime than ALNS and eLNS, with one outlier run finishing much faster.

In summary, the results indicate that eALNS achieves the most robust performance by consistently providing high-quality solutions. When computation speed is critical, LNS may be considered, albeit at the expense of solution quality.

Table 4
Comparison between Gurobi and the proposed eALNS.

Size	Distribution	$\lvert R\rvert$	$\lvert N\rvert$	Gurobi				eALNS	
				Outcome	Opt. gap	Time (s)	Obj.	Time (s)	Obj.
Small	Distributed	4	10	10 solved	0.0%	30	133	4	136
Small	Centralized	4	10	10 solved	0.0%	73	110	3	116
Small	Clustered	4	10	10 solved	0.0%	89	90	3	95
Medium	Distributed	14	24	10 solved	91.7%	3600	709	74	450
Medium	Centralized	14	24	10 solved	91.1%	3600	698	61	450
Medium	Clustered	14	24	10 solved	91.7%	3600	610	68	479
Large	Distributed	80	120	3 TL / 7 OOM	–	–	–	1109	3493
Large	Centralized	80	120	2 TL / 8 OOM	–	–	–	843	3567
Large	Clustered	80	120	6 TL / 4 OOM	–	–	–	953	3849

Notes: $\lvert R\rvert$: number of requests; $\lvert N\rvert$: number of nodes; Opt. gap: final optimality gap; Obj.: objective value. Each row summarizes 10 instances. For eALNS, CPU time and objective are first averaged over five independent runs for each case and then averaged over the 10 cases in the corresponding group. Gurobi optimality gaps, CPU times, and objective values are averaged only over cases for which a feasible incumbent is available. "Outcome" reports either "10 solved" when Gurobi returned a feasible incumbent for all 10 cases, or the numbers of runs terminating at the time limit without an incumbent (TL) and because of memory limitations (OOM).

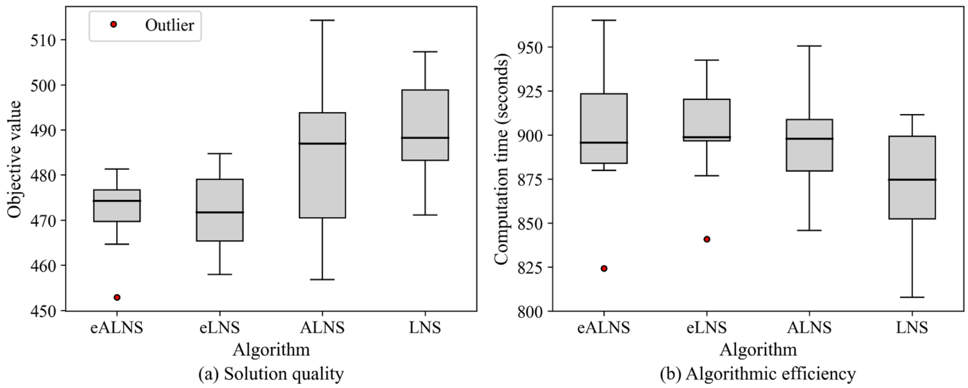


Fig. 7. Performance of LNS and its three enhanced variants.

6.3. Sensitivity analysis

This section examines the sensitivity of the proposed approach to key system parameters. For each analysis, only the parameter of interest is varied, whereas all other parameters are fixed at their baseline values. Section 6.3.1 analyzes the impact of fleet size and parking capacity, while Section 6.3.2 examines the effect of vehicle capacity and deployment cost.

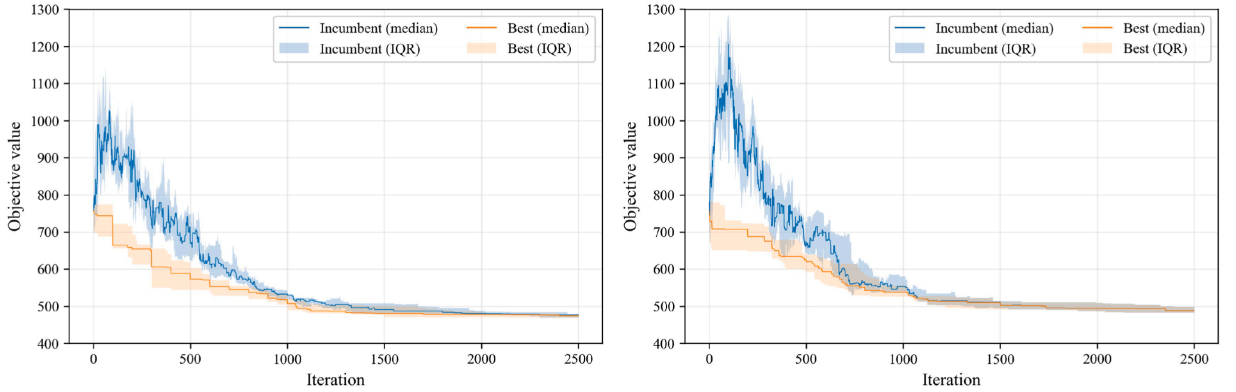
6.3.1. Impact of fleet size and parking capacity

Fleet size and station parking capacity are two critical factors that jointly determine system performance. Fleet size directly constrains the number of requests that can be served within the available resources, while the train station, as a hotspot of vehicle activity, can become a bottleneck when parking is limited. This section evaluates the effect of total fleet size on overall performance and capacity utilization, and further examines the interaction between parking capacity and fleet size by varying parking availability.

(1) Fleet size is varied from 8 to 18 in increments of 2. The resulting objective values and their components are reported in Fig. 9a, which shows that total cost decreases and eventually levels off as fleet size increases. The reduction is mainly driven by the delay penalty, indicating that scheduling flexibility improves with additional vehicles but saturates once sufficient capacity is available. The marginal gain is most pronounced when the fleet is small.

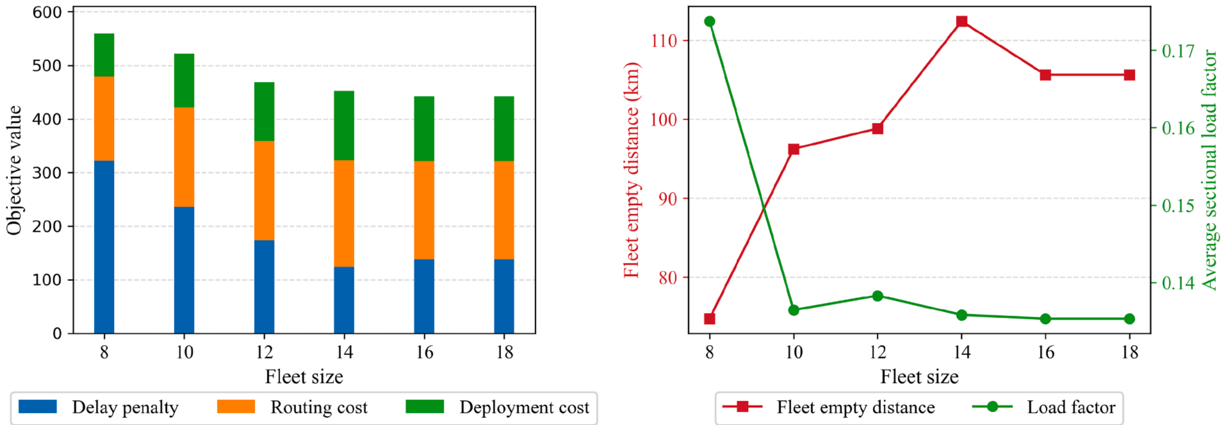
The total empty distance and average load factor are shown in Fig. 9(b). Here, the load factor is defined as the ratio of the mean carried load on non-empty route segments to vehicle capacity, reflecting how efficiently capacity is used when vehicles are loaded. Despite minor fluctuations arising from the trade-off between schedule and routing efficiency, the overall trend shows that increasing fleet size leads to greater empty distance and a lower load factor, which tend to stabilize as the fleet expands. This suggests that although additional vehicles enhance scheduling flexibility, they also lead to poorer capacity utilization. Therefore, it is important to select a moderate fleet size that balances service quality with resource efficiency.

(2) Parking capacity is varied from 1 to 6. For each parking capacity, experiments are conducted under three fleet sizes (14, 16, and 18) to assess the interaction between parking capacity and fleet size.



(a) eALNS (b) LNS

Fig. 8. Evolution of the objective value over iterations for eALNS and LNS.



(a) Cost components (b) Empty distance and load factor

Fig. 9. Impact of fleet size on system performance and capacity utilization.

The resulting objective values and their components are reported in Fig. 10. As parking capacity decreases from 6 to 2, the total cost shows minimal variation across the three fleet sizes, indicating that parking is not a binding constraint above a certain threshold. This threshold, which equals 1 in our case, is case-specific and cannot be generalized. When capacity reaches this threshold, the model becomes infeasible, leading to a sharp rise in total cost.

Comparing the same parking capacity across fleet sizes further reveals that the larger fleet incurs higher costs, most notably at capacity 2, which is just above the infeasibility threshold. As can be expected, the parking capacity constraint becomes more binding as fleet size increases, resulting in additional detours and delays.

Overall, the results suggest that system performance remains stable under moderate reductions in parking capacity, but deteriorates sharply once capacity drops below a critical availability threshold. Moreover, limited parking has a stronger adverse impact when larger fleets are deployed, highlighting the importance of jointly considering fleet size and parking resources in system design.

6.3.2. Impact of vehicle capacity and deployment cost

This section evaluates the impact of vehicle carrying capacity and deployment cost on resource utilization.

(1) Vehicle capacity is varied from 4 to 14 in increments of 2, with the capacity allocated to freight fixed at $\kappa^{\mathrm{f}} = 0.5\kappa^{\mathrm{v}}$. The resulting objective values and their components are reported in Fig. 11. At very small capacities, penalties for passenger in-vehicle congestion and skipped freight requests appear, and the delay penalty rises sharply. This indicates that insufficient vehicle capacity leads to degraded service quality for both passengers and freight. From a capacity of 6 onward, congestion and skip penalties are negligible until they vanish, and the objective stabilizes. This demonstrates the presence of a critical capacity threshold, above which the system operates feasibly and efficiently, with only marginal improvements from further increases in capacity.

In terms of utilization, Fig. 12 reports the total travel distance and empty distance across capacities. Results indicate that both measures exhibit the most pronounced improvement at the low-capacity end. Total travel distance decreases gradually, with a reduction of up to approximately 29% across the tested capacities, whereas empty travel distance shows a comparable decline of up to 27%. At the largest vehicle capacity ($\kappa^{\mathrm{v}} = 14$), both total and empty distances decrease slightly compared with the preceding case

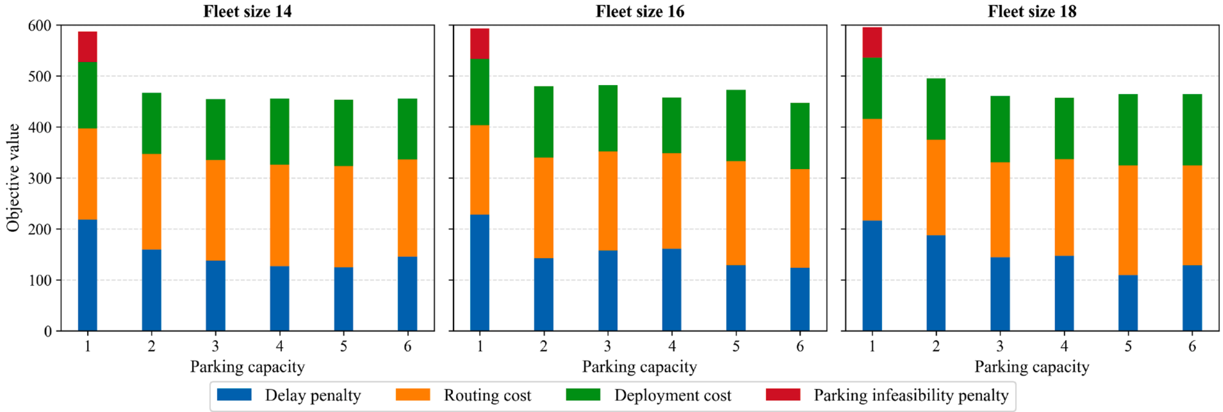


Fig. 10. Objective value under varying parking capacities.

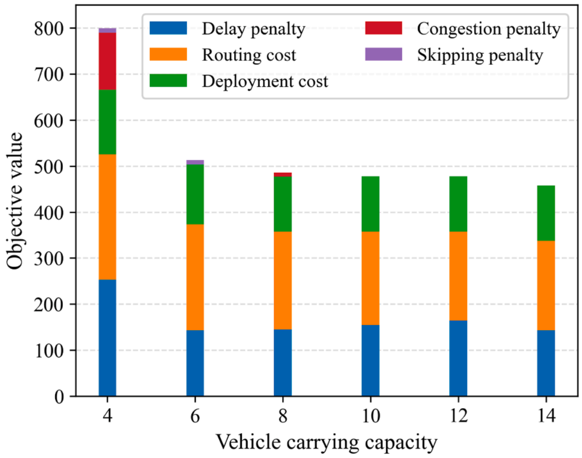
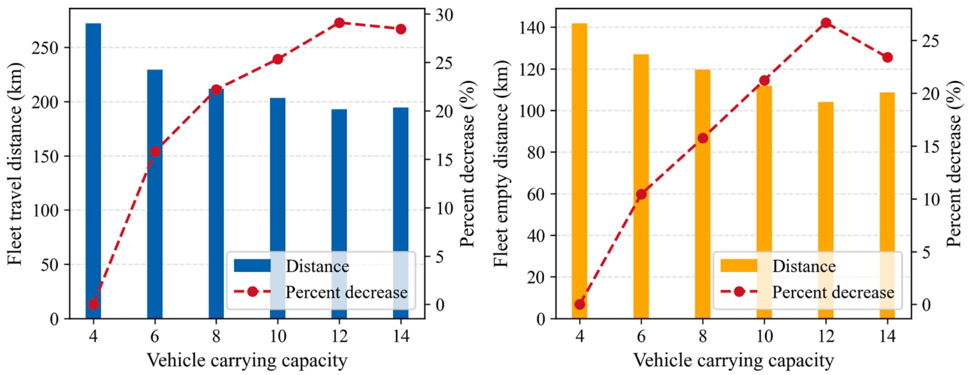


Fig. 11. Objective value under varying vehicle capacities.



(a) Total travel distance

(b) Empty travel distance

Fig. 12. Total and empty vehicle distance under varying capacities.

($\kappa^{v} = 12$). This pattern reflects the trade-off between schedule and routing efficiency, as evidenced by Fig. 11, where a smaller delay penalty is observed under $\kappa^{v} = 14$.

(2) Vehicle deployment cost is varied from 10 to 60 units in increments of 10. The resulting routing cost and delay penalty are shown in Fig. 13. As vehicle deployment cost increases, routing cost decreases, while delay penalty rises more sharply. In addition, the increase in delay penalty outweighs the savings in travel distance, which is consistent with our parameter setting where delay

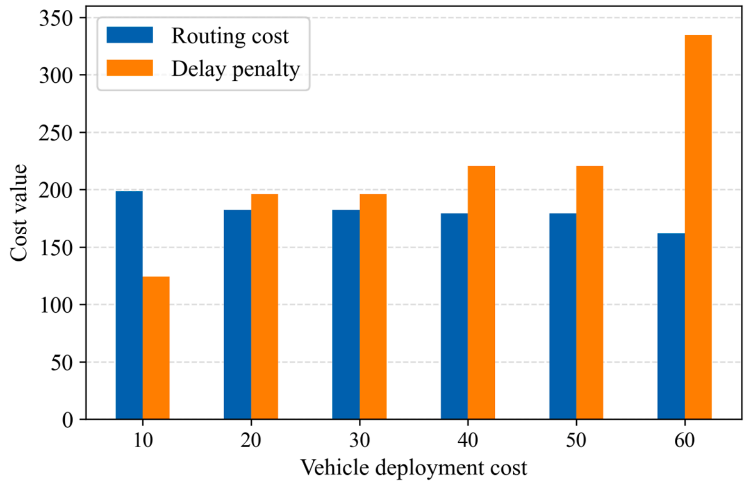


Fig. 13. Routing cost and delay penalty under varying deployment costs.

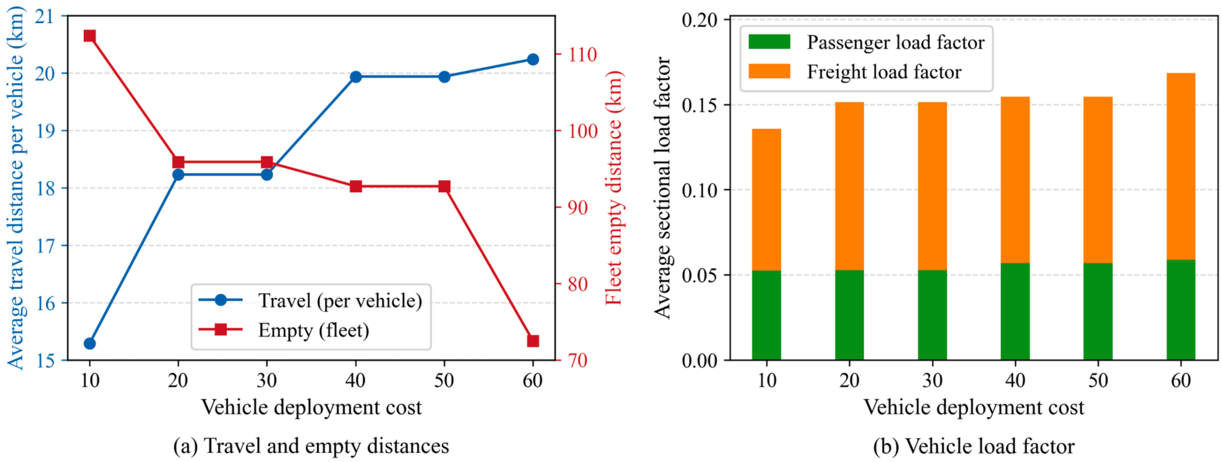


Fig. 14. Travel distance, empty distance, and load factor under varying deployment costs.

penalties, particularly for passenger delays, are relatively high. The findings indicate that when vehicles become a more valuable resource, service quality deteriorates disproportionately, even if operational distance is reduced.

A more detailed view of the routing and scheduling outcomes is given in Fig. 14, which reports travel distance at the vehicle level, together with empty distance and the average load factor, where the load factor follows the same definition as in Section 6.3.1. It is observed that with higher vehicle deployment cost, the average travel distance per vehicle increases, while empty distance decreases and the load factor improves. This demonstrates that increasing vehicle deployment cost enhances resource utilization, as vehicles operate with higher loads and reduced repositioning.

6.4. Benchmark analysis

This section compares integrated transport with a separated-transport benchmark, in which passengers and freight are served by dedicated fleets. The analysis includes three parts: Section 6.4.1 examines spatial demand distributions to identify conditions under which integration is most effective; Section 6.4.2 evaluates performance under varying total fleet sizes; and Section 6.4.3 studies different fleet reallocation plans under varying passenger–freight demand ratios.

6.4.1. Spatial demand distributions

This section considers three spatial demand distributions: distributed, centralized, and clustered, as described in Section 6.1. The spatial demand patterns for the centralized and clustered scenarios are shown in Fig. 15. The fleet size for integrated transport is fixed at 14, while in the separated case, the passenger-only and freight-only fleets are each set to 7, half of the integrated fleet.

The objective values of separated and integrated transport under the three demand distribution scenarios are reported in Fig. 16. Results show that integration achieves a lower total cost than separated services across all scenarios. In the clustered scenario, the

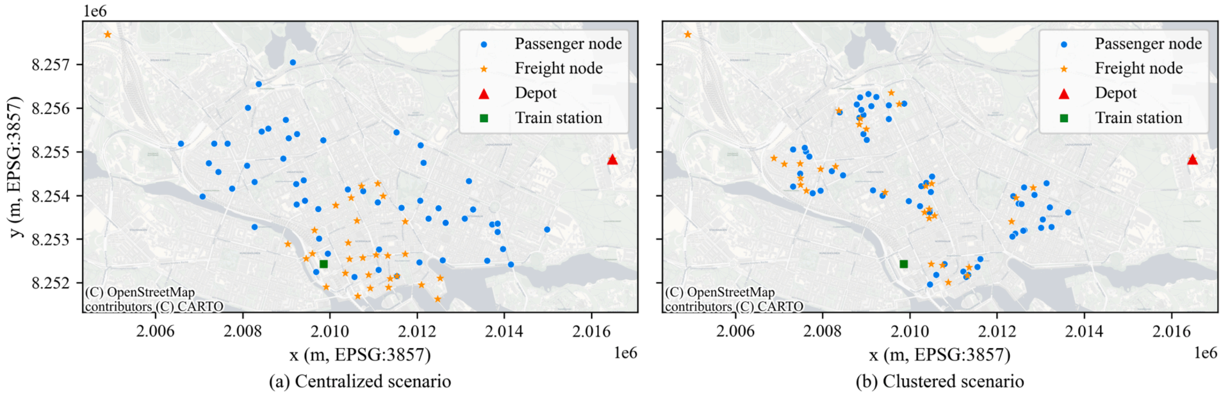


Fig. 15. Spatial distributions of demand in the centralized and clustered scenarios.

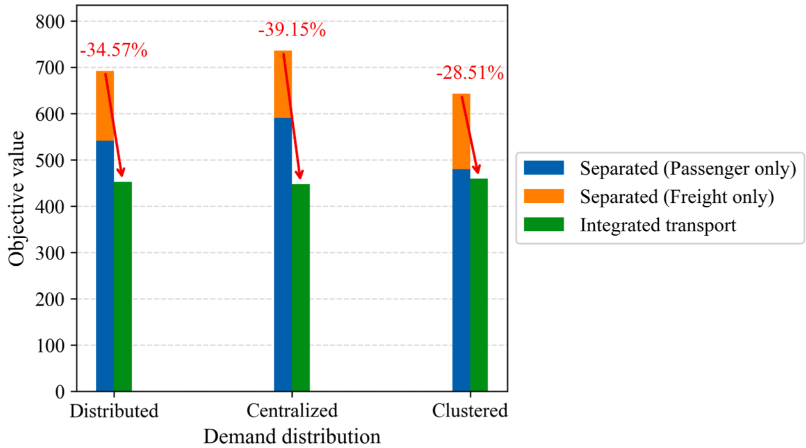


Fig. 16. Objective comparison of separated and integrated transport under different demand distributions.

benefit of integration is moderate, which may be attributed to demand being more fragmented. By comparison, the distributed and centralized scenarios yield more pronounced gains, indicating that integration is especially effective when demand is more spatially connected.

Among the three scenarios, we focus on the distributed scenario since the integrated case serves as the baseline, and compare it with the corresponding separated benchmark to analyze the performance indicators in detail. Results are reported in Table 5, covering per-passenger, per-freight, and per-route perspectives. Integrated transport achieves a 35% improvement in the overall objective, primarily by reducing delay penalty by 69%, as indicated in the 'Cost' columns, where the separate delay component and overall objective are given. This gain stems largely from the passenger side: average passenger delay is reduced by 65%, with substantial reductions in pickup and detour delays, while the reduction in transfer delay is comparatively smaller. In contrast, freight performance is mixed: while pickup delay decreases, detour delay increases, leading to a 59% rise in total delay. From a network perspective, integration reduces average route delay by 12% at the cost of a 25% increase in travel distance. The observed improvements in passenger service quality and overall system efficiency highlight the benefit of integration, where resource sharing across passengers and freight enables a balance between schedule coordination and route minimization.

6.4.2. *Total fleet size*

To examine the benefit of integration under different capacity levels, the total fleet size is varied following the settings in Section 6.3.1. In the separated case, the passenger-only and freight-only fleets are each set to half of the total fleet.

The objective values of separated and integrated transport under varying fleet sizes are reported in Fig. 17, where integrated transport achieves a lower total cost compared with separated transport across all fleet scenarios.

In addition, as fleet size increases, the objectives of both approaches decrease. For separated transport, the reduction is mainly driven by lower passenger-related costs, which reflects the model's prioritization of passengers over freight; consequently, enlarging the fleet in a separated system primarily benefits passengers, while freight shows limited variability. By contrast, integrated transport achieves efficiency by consolidating resources across both passenger and freight services. However, as fleets grow larger, the marginal benefit of such integration declines, and the objective gap between integrated and separated transport narrows.

Table 5
Benchmark comparison across passenger, freight, and route metrics.

Case	Delay per passenger				Delay per freight			Per-route		Cost	
	Pickup	Detour	Transfer	*Sum*	Pickup	Detour	*Sum*	Delay	Dist	DP	Obj
Int	0.6	0.8	1.7	3.1	0.5	9.9	10.4	41.6	15.3	124.2	452.9
Sep	2.5	3.7	2.7	8.9	2.1	4.4	6.5	47.4	12.3	402.7	692.1
Red	-78%	-78%	-36%	-65%	-78%	126%	59%	-12%	25%	-69%	-35%

Notes: Dist = distance; DP = delay penalty; Obj = objective value; Int = integrated transport; Sep = separated transport; Red = reduced percent, denoting the relative change of the integrated case compared with the separated case, with negative values indicating a reduction and positive values indicating an increase. All delay times are measured in minutes. Distance is measured in kilometers.

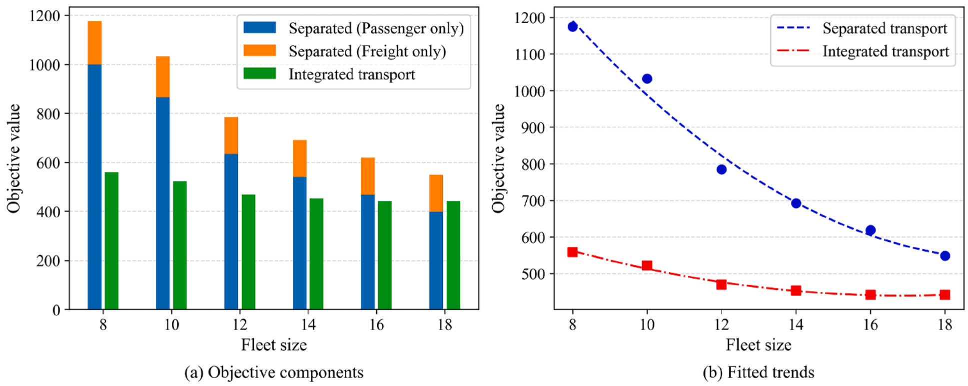


Fig. 17. Objective comparison of separated and integrated transport under different fleet sizes.

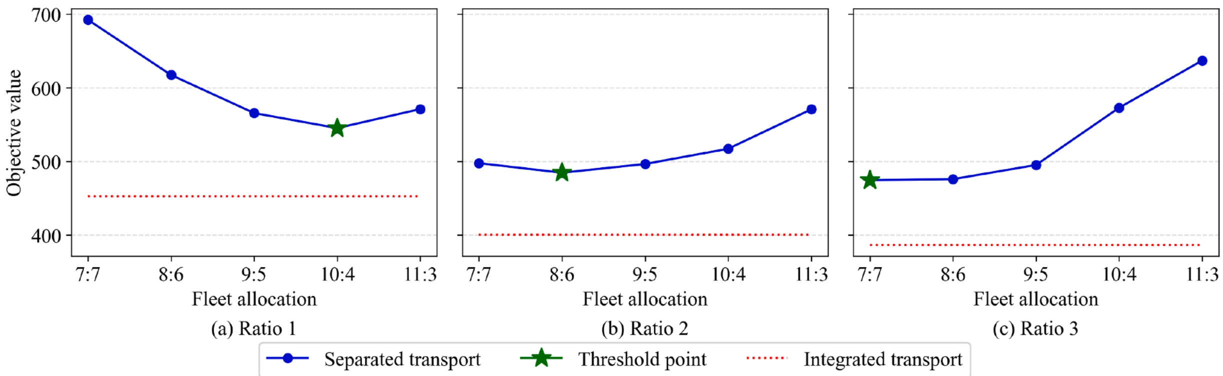


Fig. 18. Objective comparison of separated and integrated transport under different demand ratios.

Overall, these results highlight that integration is most valuable when resources are limited, enabling pronounced improvements for both passenger and freight services, whereas with larger fleets, the separated approach already provides sufficient flexibility to passengers and diminishes the relative advantage of integration.

6.4.3. Passenger–freight demand ratios and fleet reallocation

Sections 6.4.1 and 6.4.2 evaluate the benefit of integration under the setting where each dedicated fleet in separated transport equals half of the total fleet. This section relaxes that assumption by allowing the total fleet to be reallocated between passenger-only and freight-only services. Three demand-ratio cases are considered, each representing a fixed passenger–freight composition while keeping total demand unchanged. Under the one-to-one passenger–pallet capacity normalization, these cases also provide insight into how the relative composition of passenger and freight loads affects system performance. In the baseline case (ratio 1), passenger requests account for 50% of total demand. Two additional cases are generated with passenger shares of 37.5% (ratio 2) and 25% (ratio 3). The resulting objectives of separated and integrated transport under these ratios are reported in Fig. 18, where the x-axis represents the ratio of passenger fleets to freight fleets.

Results show that integration reduces total cost compared with separated transport across all tested passenger–freight demand compositions. Across these cases, the objective value decreases from 452.89 to 386.65, corresponding to a 14.6% reduction, as the passenger share is reduced from 50% to 25%. This pattern indicates that passenger demand imposes stronger service-quality pressure in the tested setting. At the same time, the consistent cost reduction achieved by integration across all tested compositions shows that integrated transport remains cost-effective under different fixed demand compositions, as jointly managed vehicles can flexibly serve both passenger and freight requests without requiring a pre-specified fleet split between the two services.

In addition, under each demand ratio, separated transport exhibits a threshold effect in fleet reallocation: shifting vehicles from freight to passenger initially reduces cost, but beyond a certain point total cost increases. As the passenger share declines (from ratio 1 to ratio 3), this threshold shifts to a smaller passenger-to-freight fleet ratio. By contrast, integrated transport eliminates this threshold effect, showing that jointly managed fleets are less sensitive to allocation decisions. These findings highlight that integration not only reduces total cost but also mitigates inefficiencies caused by imbalanced fleet assignments across different passenger–freight demand compositions.

7. Conclusion

In this study, we propose a passenger-centric PDP with integrated road-rail connections and formulate it as an MILP that incorporates multiple service quality indicators. A matheuristic is developed to decompose the original problem into a master problem and a subproblem, which are efficiently solved using eALNS and a propagation-based approach, respectively.

Numerical experiments based on a Stockholm case study demonstrate that the proposed algorithm achieves better solutions in substantially shorter time than Gurobi, with the two enhancement techniques ensuring robust performance. The sensitivity analysis highlights the importance of jointly considering fleet size and parking capacity due to the presence of availability thresholds. In addition, larger vehicle capacity and higher deployment cost reduce empty distance, although excessive deployment cost increases delay penalty. Finally, the benchmark comparison shows that integration through resource sharing enhances schedule coordination and service quality, thereby achieving balanced efficiency. The value of integration is particularly evident when fleet resources are limited and persists across different passenger–freight demand compositions, underscoring its applicability in resource-constrained settings with heterogeneous demand structures.

Our study demonstrates that integrated transport generates additional benefits compared with separated transport, mainly from the perspective of the public transport provider that prioritizes passenger service quality. However, as multiple stakeholders are involved, their individual benefits depend on how the gains are distributed. Future work could therefore investigate methods for allocating gains in a fair and effective manner to encourage broader stakeholder participation. Another important research direction is to capture the strategic interactions among stakeholders, including incentive alignment, cooperative versus competitive behaviors, and mechanisms that facilitate deeper collaboration. Addressing these aspects would provide a more complete understanding of how integrated transport can be made attractive to all parties involved.

CRediT authorship contribution statement

Siqiao Li: Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Conceptualization; **Patrick Stokkink:** Writing – original draft, Validation, Supervision, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization; **Oded Cats:** Writing – original draft, Validation, Conceptualization; **Mahnam Saeednia:** Writing – original draft, Validation, Supervision, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization.

Declaration of competing interest

Given his role as Editor, Oded Cats had no involvement in the peer review of this article and had no access to information regarding its peer review. Full responsibility for the editorial process for this article was delegated to another journal editor. All other authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This study is part of the project *Balancing Priorities: A Stakeholder-Centric Approach to Integrated Passenger and Freight Transport* with file number NWA.1418.24.047 of the research programme Innovation and Networks, funded by the Dutch Research Council (NWO) under the grant https://doi.org/10.61686/DSQLH88357.

Data availability

Data will be made available on request.

Appendix A. Big-M values

Sufficient lower bounds for the big-M parameters in the MILP formulation can be derived from valid bounds on the corresponding variables. Let H denote the length of the planning horizon, so that $0 \leq s_{i,k} \leq H$ for all $i \in N$ and $k \in K$. The on-board and arc-based load variables are bounded by the vehicle capacity, and the train-headway quotient satisfies $0 \leq q_{r,k} \leq Q^{\max}$, where $Q^{\max} = \lceil H/\tau^{\mathrm{h}} \rceil$. Table A1 summarizes the sufficient lower bounds for the Big-M parameters in the formulation.

Table A1
Sufficient lower bounds for the Big-M parameters in the MILP formulation.

Constraint(s)	Sufficient lower bound	Bound computation
Eq. (6)	$\mathrm{M}^{\mathrm{time}}_{i,j} \geq H + \tau_{i,j} + \tau^{\mathrm{d}}\lvert\rho_i\rvert$	Arrival times satisfy $0 \leq s_{i,k} \leq H$; when arc (i,j) is not traversed, the largest possible right-hand side is obtained from the maximum arrival time at node i plus travel and dwell times.
Eq. (7)	$\mathrm{M}^{\mathrm{prec}}_{r} \geq H + \tau_{p(r),d(r)} + \tau^{\mathrm{d}}\varrho^{+}_{r}$	When request r is not assigned to vehicle k, the largest possible right-hand side is obtained from the maximum pickup time H plus the direct travel and pickup dwell times.
Eq. (12)	$\mathrm{M}^{\mathrm{load,p}}_{i} \geq \kappa^{\mathrm{v}} + \max\{0, \rho^{\mathrm{p}}_{i}\}$	The passenger load is bounded by the vehicle capacity κ^{v}; the largest passenger-load increase after visiting node i is given by the positive part of ρ^{p}_{i}.
Eq. (13)	$\mathrm{M}^{\mathrm{load,f}}_{i} \geq \kappa^{\mathrm{f}} + \max\{0, \rho^{\mathrm{f}}_{i}\}$	The freight load is bounded by the freight capacity κ^{f}; the largest freight-load increase after visiting node i is given by the positive part of ρ^{f}_{i}.
Eq. (18)	$\mathrm{M}^{\mathrm{arc,p}} \geq \kappa^{\mathrm{v}}$	By the vehicle-capacity constraint (14), $g^{\mathrm{p}}_{j,k}$ is bounded above by κ^{v}.
Eq. (19)	$\mathrm{M}^{\mathrm{arc,f}} \geq \kappa^{\mathrm{f}}$	By the freight-capacity constraint (15), $g^{\mathrm{f}}_{j,k}$ is bounded above by κ^{f}.
Eq. (20)	$\mathrm{M}^{\mathrm{cong}} \geq (1-\theta^{\mathrm{c}})\kappa^{\mathrm{v}}$	The total on-board load is bounded by κ^{v}; hence, the maximum excess above the congestion threshold $\theta^{\mathrm{c}}\kappa^{\mathrm{v}}$ is $(1-\theta^{\mathrm{c}})\kappa^{\mathrm{v}}$.
Eqs. (21) and (22)	$\mathrm{M}^{\mathrm{crowd}} \geq \kappa^{\mathrm{v}}$	By the vehicle-capacity constraint (14), $\varphi_{i,j,k}$ is bounded above by κ^{v}.
Eq. (23)	$\mathrm{M}^{\mathrm{pick}}_{r} \geq H - \bar{\tau}_r$	The pickup time is bounded above by H, while the earliest pickup time is $\bar{\tau}_r$; therefore, the maximum pickup delay is $H - \bar{\tau}_r$.
Eq. (25)	$\mathrm{M}^{\mathrm{det}} \geq H$	The detour-delay expression is bounded by the planning horizon because arrival times lie within $[0, H]$ and direct travel and dwell terms are nonnegative.
Eqs. (26) and (27)	$\mathrm{M}^{\mathrm{tra}} \geq \tau^{\mathrm{h}}(Q^{\max}+1)$	The train-headway quotient satisfies $0 \leq q_{r,k} \leq Q^{\max}$, where $Q^{\max} = \lceil H/\tau^{\mathrm{h}} \rceil$; thus, the next train departure time is bounded by $\tau^{\mathrm{h}}(Q^{\max}+1)$.

The corresponding constraints are listed below for reference:
Eq. (6): $s_{j,k} \geq s_{i,k} + \tau_{i,j} + \tau^{\mathrm{d}}\lvert\rho_i\rvert - \mathrm{M}^{\mathrm{time}}_{i,j}(1 - x_{i,j,k}),\ \forall i \in N, \forall j \in N, \forall k \in K.$
Eq. (7): $s_{d(r),k} \geq s_{p(r),k} + \tau_{p(r),d(r)} + \tau^{\mathrm{d}}\varrho^{+}_{r} - \mathrm{M}^{\mathrm{prec}}_{r}(1 - y_{r,k}),\ \forall r \in R, \forall k \in K.$
Eq. (12): $g^{\mathrm{p}}_{j,k} \geq g^{\mathrm{p}}_{i,k} + \rho^{\mathrm{p}}_{i} - \mathrm{M}^{\mathrm{load,p}}_{i}(1 - x_{i,j,k}),\ \forall i \in N, \forall j \in N, \forall k \in K.$
Eq. (13): $g^{\mathrm{f}}_{j,k} \geq g^{\mathrm{f}}_{i,k} + \rho^{\mathrm{f}}_{i} - \mathrm{M}^{\mathrm{load,f}}_{i}(1 - x_{i,j,k}),\ \forall i \in N, \forall j \in N, \forall k \in K.$
Eq. (18): $\tilde{g}^{\mathrm{p}}_{i,j,k} \geq g^{\mathrm{p}}_{j,k} - \mathrm{M}^{\mathrm{arc,p}}(1 - x_{i,j,k}),\ \forall (i,j) \in A, \forall k \in K.$
Eq. (19): $\tilde{g}^{\mathrm{f}}_{i,j,k} \geq g^{\mathrm{f}}_{j,k} - \mathrm{M}^{\mathrm{arc,f}}(1 - x_{i,j,k}),\ \forall (i,j) \in A, \forall k \in K.$
Eq. (20): $\mathrm{M}^{\mathrm{cong}} w_{i,j,k} \geq \tilde{g}^{\mathrm{p}}_{i,j,k} + \tilde{g}^{\mathrm{f}}_{i,j,k} - \theta^{\mathrm{c}}\kappa^{\mathrm{v}},\ \forall (i,j) \in A, \forall k \in K.$
Eq. (21): $\varphi_{i,j,k} \leq \mathrm{M}^{\mathrm{crowd}} w_{i,j,k},\ \forall (i,j) \in A, \forall k \in K.$
Eq. (22): $\varphi_{i,j,k} \geq \tilde{g}^{\mathrm{p}}_{i,j,k} - \mathrm{M}^{\mathrm{crowd}}(1 - w_{i,j,k}),\ \forall (i,j) \in A, \forall k \in K.$
Eq. (23): $l_{r,k} \geq s_{p(r),k} - \bar{\tau}_r - \mathrm{M}^{\mathrm{pick}}_{r}(1 - y_{r,k}),\ \forall r \in R, \forall k \in K.$
Eq. (25): $o_{r,k} \geq s_{d(r),k} - \left(s_{p(r),k} + \tau_{p(r),d(r)} + \tau^{\mathrm{d}}\varrho^{+}_{r}\right) - \mathrm{M}^{\mathrm{det}}(1 - y_{r,k}),\ \forall r \in R, \forall k \in K.$
Eq. (26): $e_{r,k} \geq \tau^{\mathrm{h}} q_{r,k} + \tau^{\mathrm{h}} - s_{d(r),k} - \mathrm{M}^{\mathrm{tra}}(1 - y_{r,k}),\ \forall r \in R^{\mathrm{p}+}, \forall k \in K.$
Eq. (27): $e_{r,k} \leq \tau^{\mathrm{h}} q_{r,k} + \tau^{\mathrm{h}} - s_{d(r),k} + \mathrm{M}^{\mathrm{tra}}(1 - y_{r,k}),\ \forall r \in R^{\mathrm{p}+}, \forall k \in K.$

Appendix B. Experimental parameter settings

The algorithmic parameters and hyperparameter settings used in the computational experiments are summarized in Table B1.

Table B1
Algorithmic parameters and hyperparameters.

Notation	Description	Value
$\mathcal{T}^{\text{start}}$	Initial temperature in SA	100
$\mathcal{T}^{\text{end}}$	Final temperature in SA	0.1
$\mathcal{T}^{\text{reheat}}$	Reset temperature after reheating	120
α_1^{cool}	Cooling factor before reheating	0.9972
α_2^{cool}	Cooling factor after reheating	0.9922
$\mathcal{I}^{\max}$	Maximum number of iterations	2500
$\mathcal{I}^{\text{shift}}$	Iteration interval for invoking schedule shifting	100
$\mathcal{I}^{\text{reheat}}$	Iteration at which reheating is triggered	1500
$\mathcal{I}^{\text{hold}}$	Holding iterations after reheating	100
$\mathcal{I}^{\text{polish}}$	Period for VNS on polishing the global best solution	100
$\mathcal{I}^{\text{stag}}$	Iteration threshold for stagnation perturbation	300
$\mathcal{I}^{\text{cool}}$	Cooldown period before next VNS perturbation	150
α^{smooth}	Smoothing factor in EMA update of operator weights	0.1
$\pi_{\min}$	Minimum probability threshold in EMA	0.05
$n_{\text{shaw}}^{\text{remove}}$	Number of requests removed by Shaw removal	$0.2 \times \lvert R\rvert$
$n_{\text{worst}}^{\text{remove}}$	Number of requests removed by Worst removal	$0.1 \times \lvert R\rvert$
β	Bias exponent for Shaw removal selection	2
$\boldsymbol{\omega}^{\text{shaw}}$	Weights for relatedness criteria in Shaw removal operator	[0.99, 0.66, 0.33]
λ_1	Number of feasible pickup–delivery insertion positions evaluated in each candidate route	4
λ_2	Number of candidate routes considered for reinsertion of each request	4
η^{park}	Penalty per vehicle for violating parking capacity	30
η^{fleet}	Penalty per vehicle exceeding available fleet size	200

Note: $\lvert R\rvert$ denotes the number of requests.

Appendix C. Complete Gurobi–eALNS benchmark results

The complete case-level results underlying the aggregated comparison between Gurobi and eALNS are reported in Table C1.

Table C1
Complete case-level comparison of Gurobi and the proposed eALNS.

Dist.	Case ID	Small					Medium					Large				
		Gurobi			eALNS		Gurobi			eALNS		Gurobi			eALNS	
		Gap	Time	Obj.	Time	Obj.	Gap	Time	Obj.	Time	Obj.	Gap	Time	Obj.	Time	Obj.
Dis	1	0.0%	14	159	4	160	94.2%	3600	591	53	472	–	OOM	–	1323	3277
Dis	2	0.0%	44	94	4	94	89.1%	3600	697	76	353	–	OOM	–	1080	3707
Dis	3	0.0%	15	61	4	69	94.9%	3600	713	52	320	–	TL	–	1561	3139
Dis	4	0.0%	44	96	4	96	95.1%	3600	819	87	347	–	OOM	–	1442	3281
Dis	5	0.0%	20	65	4	73	91.0%	3600	510	67	369	–	OOM	–	1224	3187
Dis	6	0.0%	57	352	4	352	90.0%	3600	894	93	737	–	OOM	–	821	3583
Dis	7	0.0%	25	97	4	103	84.0%	3600	424	106	361	–	TL	–	925	2692
Dis	8	0.0%	12	31	4	31	97.0%	3600	1261	40	624	–	TL	–	763	4322
Dis	9	0.0%	18	75	3	83	91.0%	3600	529	105	416	–	OOM	–	867	4236
Dis	10	0.0%	47	296	3	296	90.9%	3600	652	60	500	–	OOM	–	1082	3510
Cen	1	0.0%	103	218	4	244	89.6%	3600	465	80	416	–	OOM	–	783	3299
Cen	2	0.0%	88	99	3	99	92.0%	3600	631	60	419	–	OOM	–	1160	3744
Cen	3	0.0%	39	91	3	98	94.0%	3600	741	51	515	–	OOM	–	946	3676
Cen	4	0.0%	76	94	2	94	95.1%	3600	800	53	409	–	OOM	–	940	3025
Cen	5	0.0%	20	41	3	42	96.2%	3600	1158	64	435	–	OOM	–	669	3705
Cen	6	0.0%	45	78	5	78	89.2%	3600	597	67	476	–	OOM	–	1015	3733
Cen	7	0.0%	36	50	3	64	88.7%	3600	435	80	283	–	TL	–	772	3153
Cen	8	0.0%	196	153	3	153	93.7%	3600	624	51	467	–	OOM	–	636	3727
Cen	9	0.0%	45	91	3	101	79.2%	3600	335	48	327	–	OOM	–	672	3625
Cen	10	0.0%	85	188	3	188	93.7%	3600	1189	58	757	–	TL	–	833	3985
Clu	1	0.0%	100	192	3	213	93.8%	3600	776	71	639	–	TL	–	839	3935
Clu	2	0.0%	112	91	3	91	91.5%	3600	410	62	303	–	TL	–	965	4075
Clu	3	0.0%	31	81	3	82	92.4%	3600	552	62	419	–	TL	–	1037	3396
Clu	4	0.0%	183	86	3	86	91.4%	3600	487	85	508	–	TL	–	1102	3097
Clu	5	0.0%	30	61	5	65	86.0%	3600	550	54	501	–	OOM	–	797	3628
Clu	6	0.0%	212	76	3	76	95.1%	3600	762	49	468	–	OOM	–	954	3989
Clu	7	0.0%	56	67	3	81	84.5%	3600	563	77	570	–	TL	–	664	3585
Clu	8	0.0%	22	66	4	66	93.5%	3600	592	75	357	–	OOM	–	1069	3793
Clu	9	0.0%	20	40	4	45	94.3%	3600	686	75	530	–	TL	–	1278	4630
Clu	10	0.0%	124	141	3	141	94.5%	3600	726	73	495	–	OOM	–	824	4361

Notes: Dis, Cen, and Clu denote distributed, centralized, and clustered spatial distributions, respectively. Time is reported in seconds. For eALNS, CPU time and objective are averaged over five independent runs for each case. "TL" indicates that Gurobi reached the time limit without returning an incumbent solution. "OOM" indicates that Gurobi reached the memory limit before producing a feasible incumbent.

References

Barrena, E., Canca, D., Coelho, L.C., Laporte, G., 2014. Single-line rail rapid transit timetabling under dynamic passenger demand. Transp. Res. B Methodol. 70, 134–150.

Behiri, W., Belmokhtar-Berraf, S., Chu, C., 2018. Urban freight transport using passenger rail network: scientific issues and quantitative analysis. Transp. Res. E Logist. Transp. Rev. 115, 227–245.

Bruzzone, F., Cavallaro, F., Nocera, S., 2021. The integration of passenger and freight transport for first-last mile operations. Transp. Policy 100, 31–48. https://doi.org/10.1016/j.tranpol.2020.10.009

Cleophas, C., Cottrill, C., Ehmke, J.F., Tierney, K., 2019. Collaborative urban transportation: recent advances in theory and practice. Eur. J. Oper. Res. 273 (3), 801–816.

Delle Donne, D., Alfandari, L., Archetti, C., Ljubić, I., 2023. Freight-on-transit for urban last-mile deliveries: a strategic planning approach. Transp. Res. B Methodol. 169, 53–81.

Department for Transport, 2025. Rail passenger numbers and crowding on weekdays in major cities in England and Wales. Technical Report. UK Government. https://www.gov.uk/government/statistics/rail-passenger-numbers-and-crowding-on-weekdays-in-major-cities-in-england-and-wales-2024/rail-passenger-numbers-and-crowding-on-weekdays-in-major-cities-in-england-and-wales.

Di, Z., Yang, L., Shi, J., Zhou, H., Yang, K., Gao, Z., 2022. Joint optimization of carriage arrangement and flow control in a metro-based underground logistics system. Transp. Res. B Methodol. 159, 1–23.

EIT Urban Mobility, 2022. Urban Vehicle Access Regulations: From Design to Implementation. Report. EIT Urban Mobility. https://www.eiturbanmobility.eu/wp-content/uploads/2022/10/EIT-UrbanMobilityNext6_HD2.pdf.

European Commission, 2007. Green Paper: Towards a new culture for urban mobility (COM/2007/551 final). Green Paper / Technical Report COM(2007) 551 final. European Commission. Brussels, 25 September 2007. https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:52007DC0551.

Fatnassi, E., Chaouachi, J., Klibi, W., 2015. Planning and operating a shared goods and passengers on-demand rapid transit system for sustainable city-logistics. Transp. Res. B Methodol. 81, 440–460.

Ghilas, V., Demir, E., Van Woensel, T., 2016. A scenario-based planning for the pickup and delivery problem with time windows, scheduled lines and stochastic demands. Transp. Res. B Methodol. 91, 34–51.

Hatzenbühler, J., Jenelius, E., Gidófalvi, G., Cats, O., 2023. Modular vehicle routing for combined passenger and freight transport. Transp. Res. A Policy Pract. 173, 103688.

Hörsting, L., Cleophas, C., 2023. Scheduling shared passenger and freight transport on a fixed infrastructure. Eur. J. Oper. Res. 306 (3), 1158–1169.

Huang, L., Wang, W., Su, Y., Li, F., Liang, Z., 2024. Integrated aircraft routing and cargo routing problem for combination airlines. Transp. Res. B Methodol. 188, 103063.

International Union of Railways (UIC), 2019. Railway Stations: Boosting the City. UIC Passenger Department, Paris, France. https://uic.org/IMG/pdf/2019_nextstation_railway_stations_boosting_the_city.pdf.

Li, S., Zhu, X., Shang, P., Li, T., Liu, W., 2023. Optimizing a shared freight and passenger high-speed railway system: a multi-commodity flow formulation with Benders decomposition solution approach. Transp. Res. B Methodol. 172, 1–31.

Li, S., Zhu, X., Shang, P., Wang, L., Li, T., 2024. Scheduling shared passenger and freight transport for an underground logistics system. Transp. Res. B Methodol. 183, 102907.

Lin, J., Zhang, F., 2024. Modular vehicle-based transit system for passenger and freight co-modal transportation. Transp. Res. C Emerg. Technol. 160, 104545.

Liu, S., Liu, L., Pei, D., Wang, J., 2023. Bi-objective bus scheduling optimization with passenger perception in mind. Sci. Rep. 13 (1), 6102.

Lu, Y., Peng, Y., Xia, D., Stokkink, P., Sharif Azadeh, S., 2026. Redesigning urban intermodal transit for passenger and freight co-modal mobility: metro line planning with integrated truck routing. Transp. Res. B Methodol. 211, 103538.

Machado, B., Pimentel, C., de Sousa, A., 2023. Integration planning of freight deliveries into passenger bus networks: exact and heuristic algorithms. Transp. Res. A Policy Pract. 171, 103645.

Mandal, M.P., Archetti, C., 2025. Decomposition matheuristics for last mile delivery using public transportation systems. Soft Comput. 29 (3), 1511–1539.

OpenStreetMap contributors, 2017. Planet dump. https://planet.osm.org, https://www.openstreetmap.org. Retrieved from OpenStreetMap.

Ozturk, O., Patrick, J., 2018. An optimization model for freight transport using urban rail transit. Eur. J. Oper. Res. 267 (3), 1110–1121.

Pogna, A., Zhang, K., Zuiderwijk, A., Stokkink, P., 2025. Assessing the Integration of Freight Transportation into Public Transport: A Stakeholder Analysis of Barriers and Opportunities. Technical Report. Center for Open Science.

Ropke, S., Pisinger, D., 2006. An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. Transp. Sci. 40 (4), 455–472.

Sampaio, A., Savelsbergh, M., Veelenturf, L.P., Van Woensel, T., 2020. Delivery systems with crowd-sourced drivers: a pickup and delivery problem with transfers. Networks 76 (2), 232–255.

Santini, A., Ropke, S., Hvattum, L.M., 2018. A comparison of acceptance criteria for the adaptive large neighbourhood search metaheuristic. J. Heuristics 24 (5), 783–815.

Savelsbergh, M., Van Woensel, T., 2016. 50th anniversary invited article-city logistics: challenges and opportunities. Transp. Sci. 50 (2), 579–590.

Shaw, P., 1998. Using constraint programming and local search methods to solve vehicle routing problems. In: Maher, M., Puget, J.-F. (Eds.), Principles and Practice of Constraint Programming — CP98. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 417–431.

Stokkink, P., De Palma, A., Geroliminis, N., 2025. Optimizing multi-modal ride-matching with transfers. Transp. Res. C Emerg. Technol. 179, 105274.

Stokkink, P., Geroliminis, N., 2025. On the optimal micro-hub locations in a multi-modal last-mile delivery system. Transp. Res. E Logist. Transp. Rev. 203, 104344.

Tang, C.-H., Yan, S., Chen, Y.-H., 2008. An integrated model and solution algorithms for passenger, cargo, and combi flight scheduling. Transp. Res. E Logist. Transp. Rev. 44 (6), 1004–1024.

Ursavas, E., Zhu, S.X., 2018. Integrated passenger and freight train planning on shared-use corridors. Transp. Sci. 52 (6), 1376–1390.

Watkins, E., Antonelli, C., Bergeling, E., 2024. Low Emission Zones: Navigating the Social Challenges of Clean Air Policies in EU Cities – Key aspects for policymakers. Technical Report. Institute for European Environmental Policy (IEEP). Brussels, Belgium. https://ieep.eu/wp-content/uploads/2024/09/Low-Emission-Zones-Navigating-the-Social-Challenges-IEEP-2024-1.pdf.

Zamal, M.A., Schrotenboer, A.H., Van Woensel, T., 2025. The two-echelon vehicle routing problem with pickups, deliveries, and deadlines. Comput. Oper. Res. 179, 107016.

Zhang, A., Van Hui, Y., Leung, L., 2004. Air cargo alliances and competition in passenger markets. Transp. Res. E Logist. Transp. Rev. 40 (2), 83–100.

Zhen, L., Zhang, N., Yang, Z., 2023. Integrated optimization for high-speed railway express system with multiple modes. Transp. Res. E Logist. Transp. Rev. 180, 103336.

Özarık, S.S., Lurkin, V., Veelenturf, L.P., Van Woensel, T., Laporte, G., 2023. An adaptive large neighborhood search heuristic for last-mile deliveries under stochastic customer availability and multiple visits. Transp. Res. B Methodol. 170, 194–220.