Transportation Research Part E 206 (2026) 104589

Contents lists available at ScienceDirect

Transportation Research Part E

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

Semi-flexible transit service optimization considering scenario-based demand fluctuations

Yating Liu [a], Ziyulong Wang [b], Oded Cats [b], Xin Pei [a], Pan Shang [c,d,*]

[a] Department of Automation, BNRist, Tsinghua University, 100084, Beijing, China
[b] Department of Transport and Planning, Delft University of Technology, 2600 GA city, the Netherlands
[c] School of Traffic and Transportation, Key Laboratory of Transport Industry of Comprehensive Transportation Theory, Beijing Jiaotong University, 100044, Beijing, China
[d] Key Laboratory of Transport Industry of Comprehensive Transportation Theory, Beijing Jiaotong University, 100044, Beijing, China

ARTICLE INFO

Keywords:
Demand-responsive transit
Semi-flexible transit
Vehicle routing problem with time windows
Space-time-state network
Alternating direction method of multipliers

ABSTRACT

Semi-flexible transit, integrating fixed-route and on-demand services, offers a demand-adaptive and cost-effective alternative for public transit users, particularly in low-demand conditions. Despite the growing interest in this system, existing approaches have failed to develop comprehensive optimization methods for managing demand fluctuations across distinct scenarios, thereby significantly constraining operational adaptability in semi-flexible transit services. To address this research gap, we propose a scenario-based optimization model that jointly determines the fleet size and master routes at the tactical level as well as sub-routes at the operational level. The objective is to minimize travel costs while ensuring service feasibility under varying passenger demand scenarios, accounting for constraints such as travel time, state changes, time windows, and route consistency. Then, an Augmented Lagrangian Relaxation under Alternating Direction Method of Multipliers (ALR-ADMM) decomposition solution framework is introduced to decouple the proposed integrated problem into three sub-problems, namely master route, sub-route and service planning problems. Numerical experiments on the Sioux-Falls network validate the proposed model and solution approach, achieving a 94.93 % reduction in computation time while maintaining an average optimality difference of 0.57 % compared to the Gurobi optimizer. Sensitivity analysis further examines the effects of vehicle capacity limits, penalty parameters, and demand stop selection, revealing their impact on computational efficiency and operational costs. The applicability of our approach is further assessed through a real-world case study on the West Jordan network, which provides evidence of the ALR-ADMM-based algorithm in terms of both solution quality and computational efficiency. Our findings illustrate the feasibility and potential of the proposed model and algorithm in navigating both the tactical and operational scheme of semi-flexible transit within modern urban transit systems.

1. Introduction

Efficient and sustainable transit solutions are essential in urban areas as they alleviate congestion, reduce environmental impact, enhance accessibility, and support the evolving mobility needs of growing populations. Traditional fixed-route transit–with predetermined stops, routes, and schedules–struggles to accommodate spatially dispersed and temporally variable travel demand, particularly when bridging the critical first- and last-mile gap. To address this pressing issue, Demand-Responsive Transit (DRT) has emerged as

* Corresponding author.
E-mail addresses: liuyt24@mails.tsinghua.edu.cn (Y. Liu), z.wang-19@tudelft.nl (Z. Wang), O.Cats@tudelft.nl (O. Cats), peixin@mail.tsinghua.edu.cn (X. Pei), shangpan@bjtu.edu.cn (P. Shang).

https://doi.org/10.1016/j.tre.2025.104589
Received 4 April 2025; Received in revised form 1 December 2025; Accepted 1 December 2025
Available online 12 December 2025

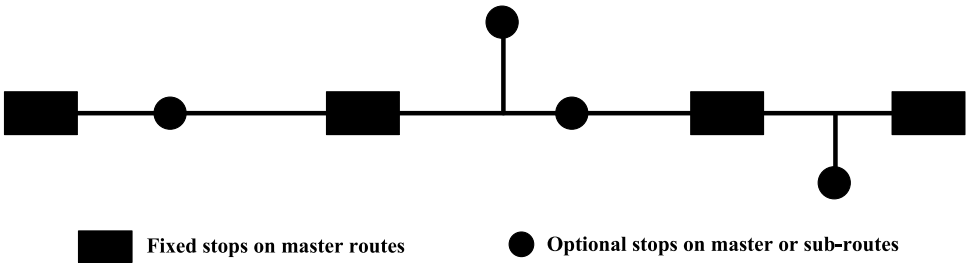


Fig. 1. Schematic drawing of semi-flexible transit service, adapted from Errico et al. (2013).

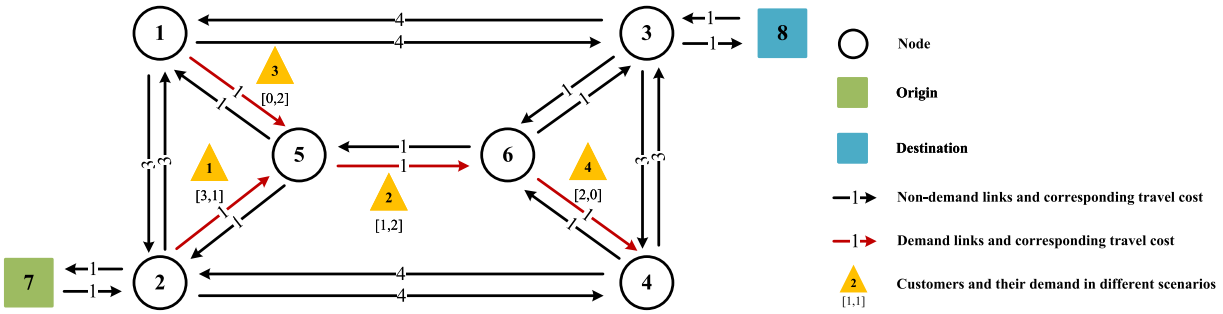


Fig. 2. Example network used for illustrating the semi-flexible transit service route design problem.

an alternative, offering flexible routing and personalized service in response to user reservations. Conversely, purely DRT often incurs circuitous detours and extended vehicle tours, which degrade service quality and cost efficiency under variable, complex, and dispersed demand patterns (Nourbakhsh and Ouyang, 2012; Chen and Nie, 2017; Frei et al., 2017; Li et al., 2023).

Semi-Flexible Transit (SFT) system, a hybrid DRT mode that blends characteristics of both traditional and flexible transit systems (Errico et al., 2013), has emerged as a promising alternative that balances these trade-offs. Specifically, fixed stops along the master route–with published timetables–provide baseline coverage and reliability, while optional stops, located either on master route or short sub-routes, are dynamically scheduled in response to passenger reservations. These optional stops on sub-routes are typically situated near the master corridor. By co-locating meeting points for fixed and flexible components (Li et al., 2023), SFT combines the efficiency and legibility of fixed-route service with the responsiveness of on-demand operations, offering a cost-effective, demand-adaptive solution, particularly suitable for "curb-to-curb" access in areas with low or fluctuating demand (Koffman, 2004; Mishra and Mehran, 2024). Fig. 1 schematically illustrates the SFT scheme (adapted from Errico et al. (2013)).

Despite the potential of SFT systems, a key challenge, however, is that the fixed service component must be tactically planned to remain effective across a range of demand patterns that simultaneously determine the operational flexible component. Much of the existing literature focuses either on purely DRT systems (Liu and Ceder, 2015; Tong et al., 2017; Chen et al., 2021) or treats fixed and flexible services separately or in a decomposed fashion (Archetti et al., 2018; Narayan et al., 2020; Huang et al., 2020; Berrada and Poulhès, 2021; Yang et al., 2021a; Shang et al., 2022; Zhao et al., 2023). Such approaches understate the operational interdependence intrinsic to SFT, where fixed and flexible elements must be jointly optimized to enhance reliability and adaptability under low-demand and high-variability conditions (Crainic et al., 2012; Frei et al., 2017; Sörensen et al., 2021).

To address the challenges identified above, we propose a novel SFT optimization model that provides a comprehensive planning method across tactical and operational levels under varying demand scenarios. Instead of enumerating all possible demand realizations, we adopt a scenario-based perspective, where a "demand scenario" represents a time-based aggregation of confirmed reservations within the upcoming planning horizon. For instance, the set of reserved passenger requests for a specific period on the following day can be treated as one demand scenario within this day-ahead horizon. By jointly planning fleet size, master routes, and sub-routes against these demand scenarios, our objective is to develop routing plans that remain reliable and efficient across the entire considered horizon. This novel variant of the Vehicle Routing Problem with Time Windows (VRPTW) is represented on a Space-Time-State (STS) network, which simultaneously encodes spatial topology, temporal constraints, and vehicle occupancy states. The problem is then decomposed and solved using a customized and improved Augmented Lagrangian Relaxation under the Alternating Direction Method of Multipliers (ALR-ADMM) algorithm. The proposed model is applied to the Sioux-Falls benchmark network and further validated on a real-world case study in West Jordan, Utah. Across both settings, the approach demonstrates computational efficiency and yields high-quality solutions, supporting its feasibility and promise for effectively managing the dynamics of SFT systems.

The remainder of this article is organized as follows: Section 2 presents a comprehensive review of state-of-the-art research on DRT services, covering flexible transit, mixed service of fixed-route and flexible transit, and their VRPTW optimization approaches. Then, Section 3 describes the SFT VRPTW using an illustrative example and introduces the optimization model. We then illustrate the

Table 1
Summary of related studies on DRT service planning.

Topic	Publication	Decision level	Fixed-line service	Problem	Demand	Solution algorithm
Flexible transit	Petit and Ouyang (2022)	Strategic	–	ODP	Pre-ordered	CA
	Tong et al. (2017)	Tactical	–	VRPTW	Pre-ordered	LR
	Bakas et al. (2016)	Operational	–	DARPTW	Pre-ordered	Modified ADARTW
	Wang et al. (2020)	Operational	–	VRP	Pre-ordered + Dynamic	Two-stage method using NSGA-II
Semi- flexible transit	Li et al. (2023)	Strategic	Pre-defined	ODP	Pre-ordered	ABC
	Chen and Nie (2017)	Strategic	Pre-defined	ODP	Pre-ordered	GA
	Zhao et al. (2023)	Strategic + Tactical	Pre-defined	ODP	Pre-ordered	Boundary-start-based two-step heuristic algorithm
	Qiu et al. (2014)	Tactical	Pre-defined	VRPTW	Pre-ordered	Dynamic station strategy + Insertion heuristic algorithm
	Narayan et al. (2020)	Operational	Pre-defined	RCS	Dynamic	Agent-based simulation
	Leffler et al. (2024)	Operational	Pre-defined	RCS	Dynamic	Agent-based simulation
	This study	**Tactical + Operational**	**Jointly optimized**	**VRPTW**	**Demand scenarios**	**ALR-ADMM**

ODP: Optimal Design Problem; DARPTW: Dial-A-Ride Problem with Time Windows; RCS: Route Choice Simulation;
CA: Continuum Approximation; LR: Lagrangian Relaxation; ADARTW: Advance request Dial-A-Ride problem with Time Windows;
ABC: Artificial Bee Colony algorithm; GA: Genetic Algorithm;
ALR-ADMM: Augmented Lagrangian Relaxation under Alternating Direction Method of Multipliers.

ALR-ADMM-based solution approach in Section 4, followed by numerical experiments on the Sioux-Falls network and a sensitivity analysis of relevant parameters in Section 5. In Section 6, a case study application for the real-world West Jordan network is conducted to further validate the applicability of the proposed algorithm. Lastly, Section 7 concludes the paper.

2. Literature review

Given that SFT can be regarded as a subset of the broader notion of DRT systems, we expand our literature review to include relevant studies from these domains. For a comprehensive survey of DRT systems, we direct readers to the work by Vansteenwegen et al. (2022), which offers a framework of their definitions and classification. In the following, we provide a comprehensive overview of the state-of-the-art research on DRT from the perspectives of modeling flexible transit in isolation, joint modeling of fixed-line and flexible transits, and optimization approaches for VRPTW, respectively.

2.1. Evolution of flexible transit services

Flexible transit is mostly studied within the scope of DRT, including Dial-A-Ride (DAR), demand-responsive service and on-demand service. DAR is considered the earliest DRT system, initially designed for individuals with limited mobility, such as the elderly and disabled (Madsen et al., 1995; Cordeau and Laporte, 2003). Since its proposal, DAR has evolved through extensive research, leading to developments such as checkpoint DAR (Daganzo, 1984) and bimodal DAR (Liaw et al., 1996).

The main difference between demand-responsive service and on-demand service lies in the timing of travel requests (Vansteenwegen et al., 2022). Demand-responsive service requires users to pre-order their travel demands or depends on demand prediction by operators, functioning in a static manner. For instance, Bakas et al. (2016) addressed the static DAR Problem with Time Windows (DARPTW) and a fixed fleet of vehicles, enabling passengers to specify their origin times from fixed bus stops. Conversely, on-demand service allows real-time travel requests, operating both statically and dynamically. Customized bus is such an emerging mode to address these real-time demands. Wang et al. (2020) proposed a two-stage method to solve a multi-objective model that minimizes total passenger travel time and operator costs, handling both static and dynamic passenger requests. Tong et al. (2017) investigated the design of customized bus services using a spatiotemporal framework, optimizing passenger-to-vehicle allocation, bus routes, and timetables. This concept was further advanced by Shen et al. (2021) and Guo et al. (2023). Chen et al. (2021) considered a customized bus route design problem, which features multi-trip, multi-pickup and delivery with time windows. In the same line of research, Petit and Ouyang (2022) introduced stochastic user demand to achieve real-time customized bus route optimization.

Table 2
Notation of sets.

Symbols	Descriptions
G	Network graph
N	Set of nodes in the network
L	Set of links in the network
L_d	Set of links with demand stops in the network
L_n	Set of links without demand stops in the network
L_v	Set of links that vehicle v passes through in the network
A_v	Set of STS arcs that vehicle v passes through in the network
A_d	Set of STS arcs that correspond to links with demand stops that are served by vehicles
S	Set of demand scenarios, representing different reserved demand combinations
T	Set of time
V	Set of originally scheduled vehicles
V^*	Set of re-employed vehicles

Table 3
Notation of indices.

Symbols	Descriptions
i, j, j'	Indices of nodes in the network
o_v	Index of origin of vehicle v, with departure time t_v^{start} and state w_o
d_v	Index of destination of vehicle v, with arrival time t_v^{end} and state w_d
(i, j)	Index of links in the network
(i, j, t, t', w, w')	Index of STS arcs that vehicle v passes through in the network
$t, t', t'', t_v^{\text{start}}, t_v^{\text{end}}$	Indices of time
w, w', w'', w_o, w_d	Indices of loading state
s	Index of demand scenarios
v	Index of vehicles, including originally planned vehicles and re-employed vehicles
p	Index of passenger

Compared to static demand forecasts, optimization approaches that consider dynamic and real-time demands are more aligned with real-world applications. However, these approaches entail a more complex operational environment, potentially resulting in route or schedule variations as well as increased computational costs.

2.2. Integration of fixed-line and flexible transit services

It has been increasingly recognized that a single mode of transit is often inadequate for meeting the diverse travel demands of travelers. While fixed-line transit is effective in densely populated areas, DRT is more suitable for areas with low-to-moderate demands (Qiu et al., 2014; Sörensen et al., 2021; Berrada and Poulhès, 2021). However, completely replacing traditional fixed-line transit with DRT would result in significant operational costs (Calabrò et al., 2023; Shahin et al., 2024) and scalability issues (Narayan et al., 2022). To address this challenge, transit systems that combine the characteristics of both traditional transit and DRT have emerged, such as Demand-Adaptive Systems (DAS) and SFT systems. Flusberg (1976) described an implementation of such a combined transit system in Merrill, Wisconsin. Since then, several studies have explored integrating traditional fixed-route service with a demand-adaptive service, incorporating both fixed and flexible service into a single transit system. For example, Qiu et al. (2014) proposed a dynamic stop strategy to improve the performance of flex-route transit, where the passenger requests are uniformly distributed in both flexible zone and fixed checkpoints. Chen and Nie (2017) analyzed a new transit system that integrates traditional fixed-line service with a flexible service, in which the flexible service operates with a stable headway. Narayan et al. (2020) proposed a multimodal route choice and assignment model that allows users to combine fixed and flexible transit or use them as individual modes with demand being endogenously defined. A stochastic user equilibrium model for combined fixed and flexible transit services was formulated and solved by Liu et al. (2025) Additionally, Zhao et al. (2023) proposed an optimization method to jointly design regular and DRT services, where the terminal stops of regular bus lines, the service area of the DRT, and the fleet size of both regular and DRT are optimized simultaneously. Leffler et al. (2021) developed a simulation model to evaluate the replacement of traditional public transit services with DRT services along branch routes. Building on this work, Leffler et al. (2024) proposed a utility-based transit route-choice model, which allows users to combine walking, fixed-line transit and flexible transit into a single trip. Furthermore, an enhanced SFT system where one flexible stop can serve multiple users was proposed by Li et al. (2023). These combined transit systems offer both reliability and flexibility, providing passengers with multiple travel options. The related studies on DRT service planning are summarized in Table 1.

2.3. Optimization of routing and scheduling in demand-responsive transit services

Most optimization studies for DRT systems have focused on the Vehicle Routing Problem (VRP) (Bruni et al., 2014; Guo et al., 2018) or the location problem for demand stops (Qiu et al., 2014). However, there is growing interest in the joint optimization of

Table 4
Notation of parameters.

Symbols	Descriptions
$q^s_{i,j}$	Number of passengers with boarding requests at demand stop on link (i,j) under scenario s
$c_{i,j,t,t',w,w'}$	Travel cost of vehicles from node i at time t to node j at time t', with loading state changing from w to w'
$TT_{i,j}$	Travel time of link (i,j)
t^{start}_v	Departure time of vehicle v from origin
t^{end}_v	Arrival time of vehicle v at destination
$[e^s_p, l^s_p]$	Designated boarding time window of passenger p, where e^s_p and l^s_p represent the earliest and latest boarding times for passenger p in scenario s, respectively
Q_v	Maximum passenger capacity of vehicle v
M	A large number
ξ	Penalty cost incurred for deploying re-employed vehicles, = 0 if no re-employed vehicle is deployed
π	Linear coefficient of travel cost to travel time
k	Number of iterations
$\theta^{(k)}$	Step size at iteration k, given by $\theta^{(k)} = \frac{1}{k+1}$
$LB^{(k)}, UB^{(k)}$	Lower and upper bounds obtained after the kth iteration of ALR-ADMM algorithm
LB, UB	Final lower and upper bounds at convergence of ALR-ADMM algorithm

Table 5
Notation of variables.

Symbols	Descriptions
$y_{i,j,v}$	= 1 if the master route of vehicle v passes through arc (i,j); = 0 otherwise
$x^s_{i,j,t,t',w,w',v}$	= 1 if the sub-route of vehicle v passes through arc (i,j) in time window (t,t') with state (w,w') in scenario s; = 0 otherwise
$z^s_{i,j}$	= 1 if the demand stop on link (i,j) is determined to be served in scenario s; = 0 otherwise
$\lambda^s_{i,j,v}, \lambda^s_{i,j}$	Lagrangian multipliers
ρ_1, ρ_2	Quadratic penalty parameters of ALR-ADMM algorithm ($\rho_1, \rho_2 > 0$)

both routing and scheduling, leading to a focus on the VRPTW. Crainic et al. (2012) proposed a mathematical description and a solution method to determine the master schedule for selected compulsory stops based on their time windows in a DAS system. Bakas et al. (2016) modified the advanced request DAR problem with time windows (ADARTW) algorithm developed by Jaw et al. (1986), formulating an objective function that considered the negative effects from customers' perspective while also capturing the operator's cost. Quadrifoglio et al. (2006) introduced a method to design and evaluate the performance of mobility allowance shuttle transit services, where vehicles may deviate from a fixed route with some compulsory checkpoints to serve demand distributed within a certain service area. This method was further developed by the same team in subsequent works (Quadrifoglio et al., 2008; Lu et al., 2011).

As for the solution methods, heuristic methods have been widely adopted for DRT routing and scheduling problems. For instance, Quadrifoglio et al. (2007) designed an insertion heuristic method to solve the VRPTW more efficiently. Besides, other heuristic methods, such as NSGA-II (Wang et al., 2020; Ma et al., 2021), adaptive memory programming (Malucelli et al., 2001), ant colony algorithm (Li et al., 2021), and genetic algorithm (Shrivastava and O'Mahony, 2006; Jorgensen et al., 2007; Guo et al., 2018; Huang et al., 2020) also perform well in jointly optimizing routing and scheduling for DRT systems. Apart from these heuristic algorithms, Yao et al. (2019) introduced the ADMM algorithm to solve the multi-VRP, originally used in the field of convex optimization as an integration of the ALR and Block Coordinate Descent (BCD) methods. This approach was then applied to VRP optimization by Yang et al. (2020, 2021b, 2022a,b), Guo et al. (2023) and Wang et al. (2024), demonstrating that the ALR-ADMM-based approach can efficiently obtain high-quality solutions with relatively tight lower bounds.

2.4. Study contributions

Prior work has largely treated the fixed-route and demand-responsive components of DRT separately; a few studies considered joint optimization of consistency between fixed and flexible services within SFT systems. Meanwhile, variability in passenger requests continues to hinder wider adoption of DRT by introducing uncertainty into route and schedule planning. To address this gap, we propose a scenario-based Semi-Flexible Transit Vehicle Routing Problem with Time Windows (SFT-VRPTW) method that explicitly accounts for demand fluctuations. The specific contributions of this study are as follows:

- Problem formulation: We introduce a scenario-based SFT optimization framework that enhances service efficiency and reduces operational costs. "Demand scenarios" are defined as time-based aggregations of confirmed passenger reservations within a day-ahead planning horizon. Within this framework, fleet size and fixed master routes are optimized as robust tactical decisions across all scenarios, while flexible sub-routes are determined as scenario-specific operational adjustments to ensure consistency.
- Mathematical modeling: We develop a comprehensive mathematical formulation on a three-dimensional STS network. This unified representation simultaneously captures vehicle routes, schedules, and onboard occupancy while accommodating controlled route deviations and time-window constraints.

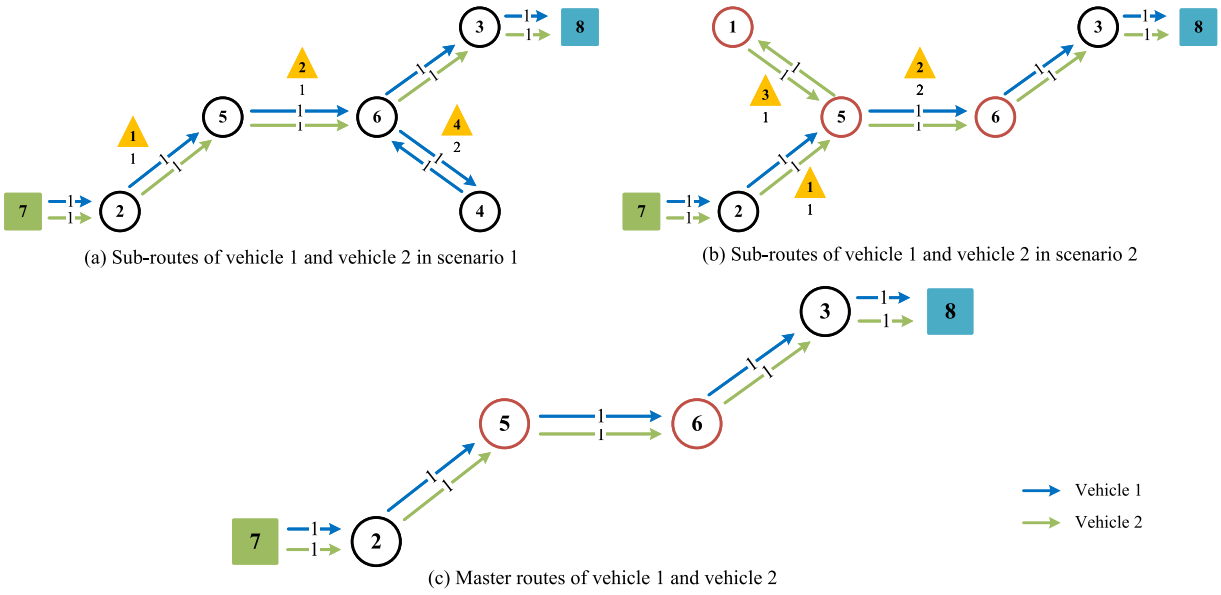


Fig. 3. Master routes and scenario-based sub-routes for vehicles 1 and 2.

- Solution algorithm: We design a scalable and computationally efficient ALR-ADMM method that decomposes the integrated problem into three manageable, interrelated sub-problems. For realistic transit network instances, the approach converges rapidly to high-quality solutions and demonstrates clear efficiency advantages over the commercial solver Gurobi.

3. Methodology

This section addresses the routing problem for SFT under fluctuating demands. In SFT systems, vehicles operate along pre-planned routes but can deviate under scenario-dependent demand variations, subject to capacity and detour feasibility constraints. The master route is defined as the basic route which vehicles must follow during each trip, while the sub-routes represent deviations or detours made in response to passenger demand at optional stops. To optimize these routing decisions, we develop an STS network-based optimization model that integrates vehicle routing constraints, service time windows, and passenger demand fluctuations. This model simultaneously designs the master route and scenario-based sub-routes for the vehicles by considering multiple demand scenarios. The objective is to minimize the total travel cost for all vehicles while ensuring feasible service operations across demand variations.

3.1. Network construction

The problem is modeled on an urban road network represented as a graph $G = (N, L)$, where N represents the set of nodes and L is the set of arcs in the network. Specifically, node set N indicates intersections in the real-world road network, while link set L represents road segments. Among these links, L_d denotes the set of links with passenger demands, which means that there is a demand stop on link (i, j) with customers' boarding requests. Each state arc $(i, j, t, t', w, w') \in A_v$ is associated with a vehicle routing cost $c_{i,j,t,t',w,w'}$, where i and j represent the nodes, t and t' indicate the time that vehicle v arrives at i and j respectively, while w and w' denote the number of on-board passengers when v arrives at the corresponding node. To capture passenger demand fluctuations, we introduce S as the set of demand scenarios, where each scenario represents a distinct combination of advanced reservations aggregated over a time period within the planning horizon. For each scenario $s \in S$, $q^s_{i,j}$ indicates the number of boarding requests of the demand stop at link (i, j). Set V represents the originally scheduled vehicles operating in the SFT system, which follow master routes but can deviate to serve passenger requests along branch lines. Each vehicle v starts its journey from an origin o_v, picks up passengers at a sequence of links, and reaches its destination d_v, which corresponds to passengers' destination.

3.2. Assumptions and notations

Assumptions 1–4 are made to construct and simplify the model, thereby improving computational efficiency without losing realism. Complete definitions of all sets, indices, parameters, and decision variables are provided in Tables 2–5.

Assumption 1 [Time window adherence]: Passengers must submit their reservation requests throughout the lead time, including destination, expected arrival time, boarding stop, and the number of boarding passengers.

Assumption 2 [Static reservations]: Once a passenger's reservation is confirmed, the passenger cannot cancel or change the

reservation request.

Assumption 3 [Limited capacity]: Vehicle capacity is fixed under a "one person, one seat" policy, and the bundling of travel requests is not considered.

Assumption 4 [Re-employed vehicles]: An additional set of re-employed vehicles is available for deployment at a penalty cost if the originally scheduled fleet is insufficient.

3.3. Illustrative example

To illustrate the service design problem and the proposed optimization approach, we consider an 8-node network example. As shown in Fig. 2, this road network consists of 8 nodes and 22 arcs, with links (2,5), (5,6), (1,5) and (6,4) defined as links with demand stops. Two vehicles, each with a capacity of 3 passengers, provide services from node 7 to node 8. The goal here is to plan the master route for each vehicle and determine the corresponding sub-routes under different scenarios. The resulting master and sub-routes of the vehicles generated by the optimization model are shown in Fig. 3(a) and (b).

In scenario 1, vehicle 2 follows the master route, while vehicle 1 deviates at node 6 to serve the passenger request along link (6,4). In scenario 2, vehicle 1 adheres to the master route, and vehicle 2 deviates at node 5 to serve the passenger requests along link (1,5). Fig. 3 (c) illustrates the master routes for both vehicles 1 and 2. It can be observed that the master routes for both vehicles are identical, being 7-2-5-6-3-8. Across all scenarios, sub-routes cover the determined master route, which serves as the backbone of the service.

3.4. Model formulation

This section presents the core optimization model for designing SFT operations within the STS network, explicitly accounting for demand variability through multiple representative demand scenarios. Based on the Assumptions stated in Section 3.2, Rules 1–3 are set to construct the STS network and remove redundant arcs in advance, which can reduce search space and improve computational efficiency.

Rule 1 [Service time window]: If vehicle v serves passenger p in scenario s, then the departure time index t of vehicle v serving passenger p must fall within the required boarding time window $[e_p^s, l_p^s]$, that is, $w' > w$ if and only if $e_p^s \le t \le l_p^s$.

Rule 2 [Vehicle loading capacity]: For any vehicle v, the passenger loading state w and w' should not be larger than the vehicle capacity Q_v at any time, that is, $0 \le w \le Q_v$ and $0 \le w' \le Q_v$.

Rule 3 [Link travel time]: For each STS arc (i, j, t, t', w, w'), the relationship between time t and t' satisfies $t' = t + TT_{i,j}$, where $TT_{i,j}$ is the travel time of link (i, j).

Regarding **Rule 3**, the link travel time $TT_{i,j}$ is flexible to incorporate time-dependent or congestion-sensitive traffic effects. In such cases, the state arc $(i, j, t, t', w, w') \in A_v$ can be generalized to $(i, j, t, t + \tau_t^s, w, w') \in A_v$, where the time index τ_t^s refers to the link travel time if vehicle v enters link (i, j) at time t in scenario s (Lu et al., 2022).

The mathematical formulation is built directly upon the foundational Assumptions and Rules. The complete optimization model for the SFT vehicle routing problem, with the primary objective of minimizing total travel costs, is presented in Eq. (1):

$$\min Z = \sum_{s \in S} \sum_{v \in V \cup V^*} \sum_{(i,j,t,t',w,w') \in A_v} c_{i,j,t,t',w,w'} x^s_{i,j,t,t',w,w',v} \tag{1}$$

In Eq. (1), an additional set of re-employed vehicles V^* is introduced to minimize the fleet size while maintaining service feasibility through penalty-based outsourcing. Specifically, when the originally scheduled fleet is insufficient, re-employed vehicles V^* are activated to accommodate the excess demand. Conversely, any originally scheduled vehicles that are redundant stay in the depot via dummy arcs $(o_v, o_v, t, t+1, w_o, w_o)$. The routing cost $c_{i,j,t,t',w,w'}$ is defined to capture both operational and penalty costs. As shown in Eq. (2), π represents the linear coefficient relating travel cost to travel time, and ξ denotes the penalty cost incurred for deploying re-employed vehicles. Further details on re-employed vehicles can be found in Yang et al. (2022b)

$$c_{i,j,t,t',w,w'} = \begin{cases} \pi \cdot TT_{i,j} + \xi, & i = o_v, j \neq o_v \\ 0, & i = j = o_v \\ \pi \cdot TT_{i,j}, & \text{otherwise} \end{cases} \tag{2}$$

The constraints are presented in Eqs. (3) to (11).

To ensure feasible routing, flow balance constraints enforce that each vehicle enters and exits the network correctly while maintaining a consistent flow at intermediate nodes. Specifically, each vehicle v departs from its origin o_v via precisely one arc and arrives at its destination d_v via one arc only. At intermediate nodes, the number of incoming arcs must match the number of outgoing arcs, maintaining flow equilibrium throughout the network:

$$\sum_{(i,j) \in A_v} y_{i,j,v} = 1, \forall v \in V \cup V^*, i = o_v \tag{3}$$

$$\sum_{(i,j) \in A_v} y_{i,j,v} = 1, \forall v \in V \cup V^*, j = d_v \tag{4}$$

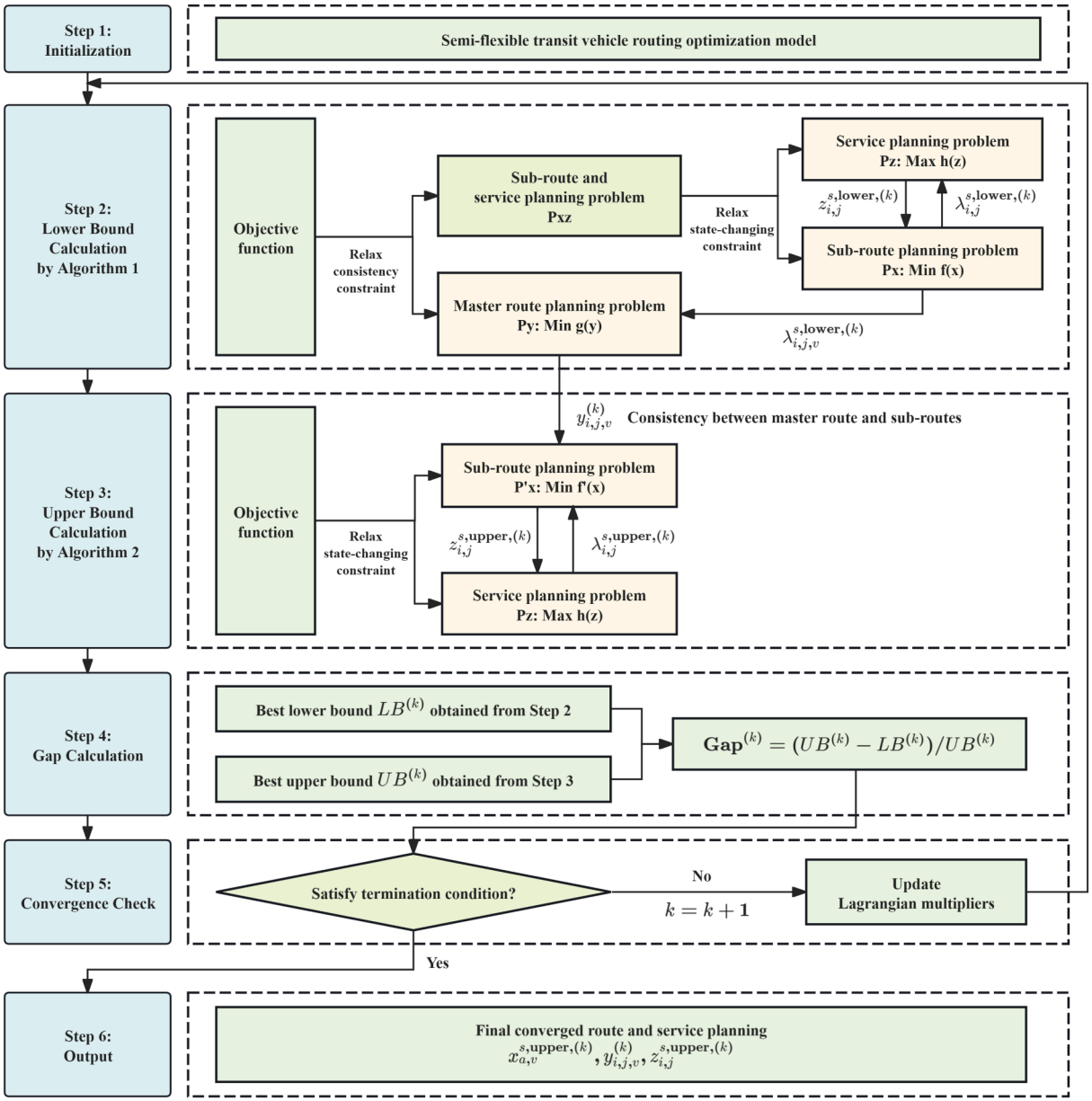


Fig. 4. ALR-ADMM decomposition framework.

$$\sum_{(i,j)\in A_v} y_{i,j,v} - \sum_{(j,j')\in A_v} y_{j,j',v} = 0, \forall v \in V \cup V^*, i \neq o_v, j' \neq d_v, j \notin \{o_v, d_v\} \tag{5}$$

For sub-routes, flow balance constraints remain consistent across all scenarios. Each vehicle v departs from its origin o_v via exactly one arc and arrives at the destination d_v via precisely one arc. Similarly, the number of departure arcs must match the number of arrival arcs for intermediate nodes, ensuring flow equilibrium within the network under all scenarios:

$$\sum_{(i,j,t,t',w,w')\in A_v} x^s_{i,j,t,t',w,w',v} = 1, \forall s \in S, v \in V \cup V^*, i = o_v, t = t^{\text{start}}_v, w = w_o \tag{6}$$

$$\sum_{(i,j,t,t',w,w')\in A_v} x^s_{i,j,t,t',w,w',v} = 1, \forall s \in S, v \in V \cup V^*, j = d_v, t' = t^{\text{end}}_v, w' = w_d \tag{7}$$

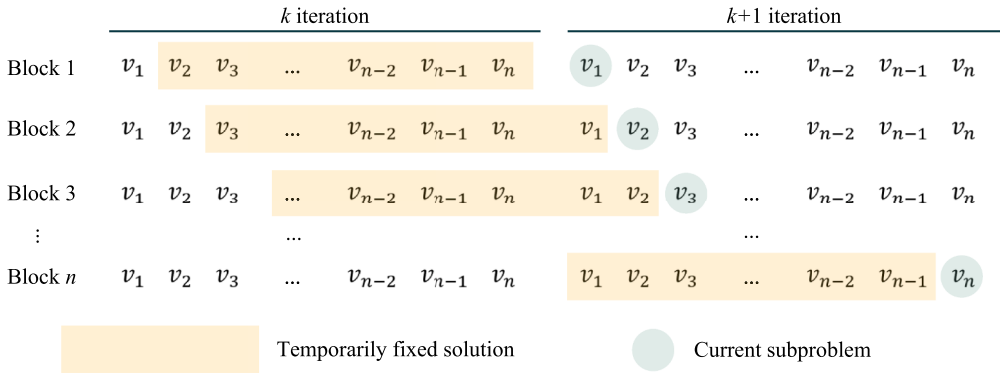


Fig. 5. ADMM procedure for iteratively solving the sub-problems, inspired by Yao et al. (2019).

$$\sum_{(j,t',w')\in A_v} x^s_{i,j,t,t',w,w',v} - \sum_{(j',t'',w'')\in A_v} x^s_{j',i,t'',t,w'',w,v} = 0,$$
$$\forall s \in S, v \in V \cup V^*, (i,t,w) \notin \{(o_v, t_v^{\text{start}}, w_o), (d_v, t_v^{\text{end}}, w_d)\} \tag{8}$$

At links with demand stops, a state-changing constraint ensures that the load state difference between the outgoing and incoming nodes of the link matches the passenger requests at that demand stop. This condition guarantees that if a demand stop is determined to be served in a scenario, vehicles serving the stop should fully accommodate all passenger demand. In cases where the capacity of originally scheduled vehicles is insufficient, re-employed vehicles are deployed to meet the excess demand:

$$\sum_{v\in V\cup V^*} \sum_{(i,j,t,t',w,w')\in A_d} x^s_{i,j,t,t',w,w',v} \times (w' - w) = q^s_{i,j} z^s_{i,j}, \forall s \in S, (i,j) \in L_d \tag{9}$$

At links without demand stops, another state-changing constraint is imposed to ensure the loading state difference between the outgoing and incoming nodes of the link is zero. This maintains balance within the transit system and keeps consistency in vehicle load across the network:

$$\sum_{v\in V\cup V^*} \sum_{(i,j,t,t',w,w')\in A_v} x^s_{i,j,t,t',w,w',v} \times (w' - w) = 0, \forall s \in S, (i,j) \in L_n \tag{10}$$

For the consistency between master routes and sub-routes, we define that the sub-routes of vehicles must contain their master routes. In other words, if a link is included in the master route of vehicle v, this link is bound to exist in its sub-routes. Sub-routes may deviate due to demand variations, but they must always include the master route as a base structure.

$$\sum_{(i,j,t,t',w,w')\in A_v} x^s_{i,j,t,t',w,w',v} \geq y_{i,j,v}, \forall s \in S, v \in V \cup V^*, (i,j) \in L_v \tag{11}$$

Given the complexity of the SFT routing problem, a direct solution approach is computationally intractable. In the following section, we describe the proposed solution method.

4. Solution method

We propose a decomposition-based approach, using the Lagrangian Relaxation (LR) method to break down the entire problem into manageable sub-problems. Initially introduced by Kohl and Madsen (1997) to solve a VRPTW, LR demonstrated its capability to handle constraints where each customer must be served when requested. However, the direct application of LR results in sub-problems with identical structures across vehicles, leading to symmetry issues (Niu et al., 2018; Yao et al., 2019; Yang et al., 2020). To differentiate sub-problems and improve convergence, Yang et al. (2020) proposed an Augmented Lagrangian Relaxation (ALR) method, which introduces quadratic penalty terms into the objective function. The introduction of quadratic penalties in ALR, however, results in nonlinearity and coupled decision variables, making direct decomposition infeasible. To address this, we integrate ALR with the Block Coordinate Descent (BCD) method to introduce ADMM, extending the framework proposed by Yang et al. (2022b). This combined ALR-ADMM method can iteratively coordinate sub-problems in a linear format while preserving decomposition. Meanwhile, it effectively manages the dependencies introduced by ALR with computational efficiency. Thus, it has been successfully applied in various domains, including VRPTW for logistics (Wang et al., 2024).

Building on the established feasibility of the ALR-ADMM framework, we extend it to the SFT-VRPTW. The proposed solution approach decomposes the coupled problem into three sub-problems: (i) *master route planning* (fixed backbone), (ii) *sub-route planning* (scenario-dependent local deviations), and (iii) *service planning* (vehicle timing/assignment under time windows). Unlike prevailing two-block approaches that couple only master routing and service planning, introducing a dedicated sub-route layer is essential to explicitly manage service-consistency between fixed master routes and scenario-dependent flexible operations, thereby isolating the

most entangling dependencies and improving computational efficiency under fluctuating demand. Fig. 4 illustrates the information flow, while Algorithm 1 outlines the iterative procedure. Each iteration k proceeds as follows:

Step 1 – Initialization: Set multipliers $\lambda^{(0)}$, penalty ρ, and any variables; warm-start the master and scenario decisions.

Step 2 – Lower-bound (LB) evaluation: We dualize the route-consistency constraint (13) into the augmented Lagrangian, which splits the problem into a *master route planning* sub-problem **Py** and a combined *sub-route & service* sub-problem **Pxz**. We then further relax the state-changing constraint (17) within **Pxz**, yielding separable sub-problems **Px** (sub-routes) and **Pz** (service). Solving these produces a valid dual value that we record as the iteration's lower bound $LB^{(k)}$ (see Section 4.1 and Algorithm 2).

Step 3 – Upper-bound (UB) construction: We fix the master route obtained in Step 2 to directly enforce route consistency, and relax only constraint (17) to decouple **P'x** (sub-routes) and **P'z** (service). The recovered feasible solution–after lightweight repairs to satisfy service constraints–yields the iteration's upper bound $UB^{(k)}$ (see Section 4.2 and Algorithm 3).

Step 4 – Gap computation: Compute the optimality gap $\text{Gap}^{(k)} = (UB^{(k)} - LB^{(k)})/UB^{(k)}$.

Step 5 – Convergence check and updates: If stopping criteria (primal feasibility and gap tolerance) are not met, update multipliers and return to Step 2.

Step 6 – Output: Report the best feasible solution UB together with the final LB certificate, which validates solution quality, and the corresponding routing and service plan.

Algorithm 1 ALR-ADMM solution framework.

Step 1: Initialization
Initialize iteration counter $k = 0$
Initialize Lagrangian multipliers $\lambda^{s,\text{lower},(k)}_{i,j,v} = 0.1$, $\lambda^{s,\text{lower},(k)}_{i,j} = \lambda^{s,\text{upper},(k)}_{i,j} = 0.1$
Set penalty parameters $\rho_1 = 1.0,\ \rho_2 = 1.0$
Define step size $\theta^{(k)} = \frac{1}{k+1}$
Set maximum iterations $k_{\max}$ and convergence tolerance ϵ
Step 2: Lower bound evaluation
Calculate lower bound $LB^{(k)}$ via Algorithm 2
Pass $y^{(k)}_{i,j,v}$ values to **Step 3** for consistency between master and sub-routes
Step 3: Upper bound construction
Calculate upper bound $UB^{(k)}$ via Algorithm 3
Step 4: Gap computation
Compute $\text{Gap}^{(k)} = (UB^{(k)} - LB^{(k)})/UB^{(k)}$
Step 5: Convergence check and updates
if $\text{Gap}^{(k)} \le \epsilon$ or $k \ge k_{\max}$ **then**
Continue to **Step 6**
else

$$\text{Update } \lambda^{s,\text{lower},(k+1)}_{i,j,v} = \lambda^{s,\text{lower},(k)}_{i,j,v} + \theta^{(k)}\left(y^{(k)}_{i,j,v} - \sum_{a\in A_v} x^{s,\text{lower},(k)}_{a,v}\right)$$

$$\text{Update } \lambda^{s,\text{lower},(k+1)}_{i,j} = \lambda^{s,\text{lower},(k)}_{i,j} + \theta^{(k)}\left(\sum_{v\in V\cup V^*}\sum_{a\in A_d} x^{s,\text{lower},(k)}_{a,v}\times w(a) - q^s_{i,j} z^{s,\text{lower},(k)}_{i,j}\right)$$

$$\text{Update } \lambda^{s,\text{upper},(k+1)}_{i,j} = \lambda^{s,\text{upper},(k)}_{i,j} + \theta^{(k)}\left(\sum_{v\in V\cup V^*}\sum_{a\in A_d} x^{s,\text{upper},(k)}_{a,v}\times w(a) - q^s_{i,j} z^{s,\text{upper},(k)}_{i,j}\right)$$

Update $k = k + 1$
Back to **Step 2** ▷ Feedback loop
end if
Step 6: Output
The final converged route and service planning are obtained as $x^{s,\text{upper},(k)}_{a,v}$, $y^{(k)}_{i,j,v}$, $z^{s,\text{upper},(k)}_{i,j}$

4.1. Lower bound calculation

To efficiently solve the decomposed problem, we first establish a lower bound using the ALR-ADMM framework, as elaborated in Algorithm 2. To simplify notation, the arc index $(i, j, t, t', w, w') \in A_v$ is replaced with $a \in A_v$, and the original objective function (1) is reformulated as follows:

$$\min Z = \sum_{s\in S}\sum_{v\in V\cup V^*}\sum_{a\in A_v} c_a x^s_{a,v} \tag{12}$$

The consistency constraints are reformulated to align sub-route decisions with the master route:

$$\sum_{a\in A_v} x^s_{a,v} \ge y_{i,j,v}, \forall s \in S, v \in V \cup V^*, (i,j) \in L_v \tag{13}$$

At the first-level LR-based decomposition, the consistency constraint (13) is relaxed and incorporated into the objective function (12) using Lagrangian multipliers $\lambda^s_{i,j,v} \geq 0$. This reformulation yields the following LR objective function:

$$\min Z = \sum_{s\in S}\sum_{v\in V\cup V^*}\sum_{a\in A_v} c_a x^s_{a,v} + \sum_{s\in S}\sum_{v\in V\cup V^*}\sum_{(i,j)\in L_v} \lambda^s_{i,j,v}\left(y_{i,j,v} - \sum_{a\in A_v} x^s_{a,v}\right), \tag{14}$$

subject to flow balance constraints and state-changing constraints (6)–(10). This relaxation allows the problem to be decomposed into two sub-problems: (1) master route planning **Py** and (2) sub-route and service planning **Pxz**, which are solved separately as follows.

The master route planning sub-problem **Py** determines the core SFT paths that vehicles must follow, while allowing deviations in response to demand variations. Its objective function is given by:

$$\min g(y) = \sum_{s\in S}\sum_{v\in V\cup V^*}\sum_{(i,j)\in L_v} \lambda^s_{i,j,v} y_{i,j,v}, \tag{15}$$

subject to flow balance constraints (3)–(5).

The sub-route and service planning sub-problem **Pxz** determines how vehicles deviate from the master route to serve demand stops under different scenarios. The corresponding objective function is:

$$\min P(x,z) = \sum_{s\in S}\sum_{v\in V\cup V^*}\sum_{a\in A_v} \left(c_a - \lambda^s_{i,j,v}\right) x^s_{a,v} \tag{16}$$

subject to flow balance constraints (6)–(8) and state-changing constraints (9)–(10).

Although the initial decomposition separates master route planning from sub-route and service planning, the sub-problem **Pxz** still contains two interdependent decision variables x (sub-routes) and z (stops to be served). Their dependency is coupled by the hard state-changing constraints (9), which enforce consistency in passenger pickups along links with demand stops. To further simplify the problem, a second relaxation is applied. We define the pickup state changing value $(w' - w)$ for arc a as $w(a)$. This allows the state-changing constraints to be reformulated as follows:

$$\sum_{v\in V\cup V^*}\sum_{a\in A_d} x^s_{a,v} \times w(a) = q^s_{i,j} z^s_{i,j}, \forall s \in S, (i,j) \in L_d \tag{17}$$

In the objective function (16) for **Pxz**, the Lagrangian multipliers $\lambda^s_{i,j,v}$ are distinct for different vehicles, allowing for differentiation in their sub-problems. However, when the state-changing constraint (17) is relaxed into the objective function, a single shared Lagrangian multiplier $\lambda^s_{i,j}$ is introduced across all vehicles for demand-related decisions. This results in all vehicles facing the same incentive structure on these links, which can lead to partially identical shortest path sub-problems, where vehicles independently converge towards similar routes under the same cost influence. To prevent such similarity in routing and improve computational efficiency, we use the ALR, which introduces quadratic penalty terms (Yao et al., 2019; Zhang et al., 2019; Yang et al., 2022b; Wang et al., 2024). This results in the following ALR-based sub-problem (18) where another Lagrangian multiplier $\lambda^s_{i,j} \in R^*$ and a positive quadratic penalty parameter ρ_1 are introduced:

$$\begin{aligned}\min P(x,z) = &\sum_{s\in S}\sum_{v\in V\cup V^*}\sum_{a\in A_v} \left(c_a - \lambda^s_{i,j,v}\right) x^s_{a,v} + \\ &\sum_{s\in S}\sum_{(i,j)\in L_d} \lambda^s_{i,j}\left(\sum_{v\in V\cup V^*}\sum_{a\in A_d} x^s_{a,v}\times w(a) - q^s_{i,j} z^s_{i,j}\right) + \\ &\frac{\rho_1}{2}\sum_{s\in S}\sum_{(i,j)\in L_d}\left(\sum_{v\in V\cup V^*}\sum_{a\in A_d} x^s_{a,v}\times w(a) - q^s_{i,j} z^s_{i,j}\right)^2\end{aligned} \tag{18}$$

With this relaxation, the objective function **Pxz** is further decomposed into two sub-problems: (1) sub-route planning problem (**Px**) and (2) service planning problem (**Pz**). The first determines the optimal sub-routes for vehicles while considering penalties for deviations and state-changing constraints. The second identifies the optimal demand stops to be served, maximizing the total served demand when capacity and time windows are allowed, which reduces to a Knapsack problem. To prevent symmetry issues from persisting, the penalty term is assigned to sub-problem **Px**, where $z^s_{i,j}$ is regarded as a known parameter. The objective function of sub-problem **Px** on sub-route planning is given by:

$$\begin{aligned}\min f(x)_{\rho_1} = &\sum_{s\in S}\sum_{v\in V\cup V^*}\sum_{a\in A_v} \left(c_a - \lambda^s_{i,j,v}\right) x^s_{a,v} + \sum_{s\in S}\sum_{v\in V\cup V^*}\sum_{a\in A_d} \lambda^s_{i,j} x^s_{a,v}\times w(a) + \\ &\frac{\rho_1}{2}\sum_{s\in S}\sum_{(i,j)\in L_d}\left(\sum_{v\in V\cup V^*}\sum_{a\in A_d} x^s_{a,v}\times w(a) - q^s_{i,j} z^s_{i,j}\right)^2,\end{aligned} \tag{19}$$

subject to flow balance constraints (6)–(8) and state-changing constraints (10) for links without demand stops.

The objective function of sub-problem **Pz** on service planning is hence formulated as:

$$\max h(z) = \sum_{s\in S}\sum_{(i,j)\in L_d} \lambda^s_{i,j} q^s_{i,j} z^s_{i,j}, \tag{20}$$

subject to the domain of the variable $z^s_{i,j} \in \{0,1\}$.

Since sub-problem **Px** contains only variables $x^s_{a,v}$ but introduces additional quadratic terms, solving it directly remains computationally challenging due to the resulting nonlinearity and variable coupling. To address this, we adopt the n-block ADMM framework. It decomposes the ALR model into a sequence of linear sub-problems **SPx**, each corresponding to an individual vehicle. These sub-problems **SPx** are then iteratively solved using a Dynamic Programming (DP) algorithm. The iterative pattern is illustrated in Fig. 5.

In sub-problem $\mathbf{SPx}_i$, the decision variables of all vehicles, except vehicle v_i, are treated to be fixed, and only the decision variables associated with vehicle v_i are optimized. This approach ensures that the interdependencies introduced by the state-changing constraints are handled iteratively. To facilitate this process, an auxiliary variable $\mu^s_{v'}$ is introduced to represent the total passenger requests satisfied by all vehicles except v_i. This is calculated as follows:

$$\mu^s_{v'} = \sum_{v' \in V \cup V^* | \{v\}} \sum_{a \in A_d} x^s_{a,v'} \times w(a) \tag{21}$$

The quadratic penalty term in the objective function (19) enforces the state-changing constraints by penalizing deviations. To simplify the optimization, this term is expanded. Since $x^s_{a,v}$ are binary variables, the property $(x^s_{a,v})^2 = x^s_{a,v}$ is effective in linearization. Then, the objective function (19) is further decomposed into a series of sub-problems, leading to the expanded formulation given in objective function (22). This expanded objective function corresponds to sub-problem **SPx**, which aims to minimize the cost of sub-routes for a single vehicle while incorporating penalties for deviations and violations of state-changing constraints:

$$\begin{aligned} \min f(x)_{\rho_1,v} = & \sum_{s \in S} \sum_{a \in A_v} \left(c_a - \lambda^s_{i,j,v} \right) x^s_{a,v} + \sum_{s \in S} \sum_{a \in A_d} \lambda^s_{i,j} x^s_{a,v} \times w(a) + \\ & \frac{\rho_1}{2} \sum_{s \in S} \sum_{(i,j) \in L_d} \left(\sum_{a \in A_d} x^s_{a,v} \times w(a)^2 + 2\left(\mu^s_{v'} - q^s_{i,j} z^s_{i,j} \right) \sum_{a \in A_d} x^s_{a,v} \times w(a) + \left(\mu^s_{v'} - q^s_{i,j} z^s_{i,j} \right)^2 \right) \end{aligned} \tag{22}$$

Sub-problem **SPx** (22) is further simplified into Eqs. (23) to (25) by extracting the coefficients of $x^s_{a,v}$ and isolating the constant terms. These equations remain subject to flow balance constraints (6)–(8) and state-changing constraints (10) for links without demand stops:

$$\min f(x) = \sum_{s \in S} \sum_{a \in A_v} \hat{c_a} x^s_{a,v} + \tau \tag{23}$$

$$\hat{c_a} = \begin{cases} c_a - \lambda^s_{i,j,v} + \lambda^s_{i,j} \times w(a) + \frac{\rho_1}{2} \left(w(a)^2 + 2\left(\mu^s_{v'} - q^s_{i,j} z^s_{i,j} \right) \times w(a) \right), & a \in A_d \\ c_a - \lambda^s_{i,j,v}, & \text{otherwise} \end{cases} \tag{24}$$

$$\tau = \begin{cases} \frac{\rho_1}{2} \sum_{s \in S} \sum_{(i,j) \in L_d} \left(\mu^s_{v'} - q^s_{i,j} z^s_{i,j} \right)^2, & a \in A_d \\ 0, & \text{otherwise} \end{cases} \tag{25}$$

Since the consistency constraints are relaxed at the first-level LR-based decomposition, sub-route planning for the lower bound solution does not enforce master route or service planning restrictions. As a result, vehicles are free to select any shortest path within the road network, disregarding route-consistency constraints. This relaxation is essential for revealing the theoretical minimum possible cost under the given cost structure. Through iterative updates of the Lagrangian multipliers, the algorithm incrementally enforces the relaxed consistency constraints, guiding the lower bound solution towards feasibility and progressively closing the optimality gap. While the solution obtained from the lower bound may not satisfy all original problem constraints, it serves as a benchmark against which coordinated, feasible solutions are evaluated throughout the iterative process.

4.2. Upper bound calculation

In contrast to the lower bound calculation, the upper bound solution explicitly enforces the consistency constraints (13) during sub-route planning. Specifically, the master route planning variable $y_{i,j,v}$ obtained from solving sub-problem **Py** (objective function (15)) in the lower bound calculation is now fixed and used as input. Using this information, sub-problem **P'xz** is initially formulated by incorporating the state-changing constraints (17) directly into the objective function (12):

$$\min P'(x,z) = \sum_{s \in S} \sum_{v \in V \cup V^*} \sum_{a \in A_v} c_a x^s_{a,v} + \sum_{s \in S} \sum_{(i,j) \in L_d} \lambda^s_{i,j} \left(\sum_{v \in V \cup V^*} \sum_{a \in A_d} x^s_{a,v} \times w(a) - q^s_{i,j} z^s_{i,j} \right) \tag{26}$$

However, in the objective function (26), the same Lagrangian multipliers are applied across all vehicles, resulting in identical shortest path sub-problems. This symmetry issue causes multiple vehicles to follow the same routing pattern, which can hinder convergence (Niu et al., 2018; Yao et al., 2019; Zhang et al., 2019; Yang et al., 2022b; Wang et al., 2024). To address this, the ALR method is applied, which adds quadratic penalty terms to differentiate vehicle-specific sub-problems. The detailed solution process is outlined in Algorithm 3. The resulting penalized objective function for sub-problem **P'xz** on combined sub-route and service planning

Algorithm 2 Lower bound calculation.

Step 1: Decompose the initial problem into sub-problems Pxz and Py
  Relax consistency constraint Eq. (13) into objective function Eq. (12)
  Decompose the initial problem into:
  • **Pxz** (sub-route and service planning)
  • **Py** (master route planning): Eq. (15), subject to constraints (3)–(5)
Step 2: Further decompose Pxz into sub-problems Px and Pz
  Relax state-changing constraint Eq. (17) into objective function Eq. (16)
  Decompose **Pxz** into:
  • **Pz** (service planning): Eq. (20), subject to variable domain constraints
  • **Px** (sub-route planning): Eq. (19), subject to constraints (6–8), (10)
  Linearize quadratic penalty term in **Px** and solve the sub-problem using ADMM
  Obtain **SPx** problems (Eq. (23))
Step 3: Solve the master route planning sub-problem Py
for each vehicle $v \in V \cup V^*$ **do**
    Solve sub-problem Py for master route planning
    Update $y_{i,j,v}$ and store master path topology for upper bound calculation
end for
Step 4: Solve sub-problems Px and Pz
for each scenario $s \in S$ **do**
    Solve the service planning Knapsack sub-problem Pz:
    Obtain $z_{i,j}^{s,\text{lower}}$ and prepare for **Px** calculation
    Solve the sub-route planning sub-problem Px:
    for each vehicle $v \in V \cup V^*$ **do**
        Solve Eq. (23) using DP
        Obtain $\lambda_{i,j,v}^{s,\text{lower}}$ and $\lambda_{i,j}^{s,\text{lower}}$ and backtrack optimal path $x_{a,v}^{s,\text{lower}}$
    end for
end for
Step 5: Compute the lower bound for iteration k
  Calculate lower bound by:
  $LB^{(k)} = \sum_{s\in S}\sum_{v\in V\cup V^*}\sum_{a\in A_v}\left(c_a - \lambda_{i,j,v}^{s,\text{lower}}\right)x_{a,v}^{s,\text{lower}} + \sum_{s\in S}\sum_{v\in V\cup V^*}\sum_{a\in A_d}\lambda_{i,j}^{s,\text{lower}}x_{a,v}^{s,\text{lower}} \times w(a)$
  Obtain the final route and service planning for iteration k

is given as follows:

$$\begin{aligned}\min P'(x,z)_{\rho_2} = &\sum_{s\in S}\sum_{v\in V\cup V^*}\sum_{a\in A_v} c_a x_{a,v}^s + \\ &\sum_{s\in S}\sum_{(i,j)\in L_d}\lambda_{i,j}^s\left(\sum_{v\in V\cup V^*}\sum_{a\in A_d} x_{a,v}^s \times w(a) - q_{i,j}^s z_{i,j}^s\right) + \\ &\frac{\rho_2}{2}\sum_{s\in S}\sum_{(i,j)\in L_d}\left(\sum_{v\in V\cup V^*}\sum_{a\in A_d} x_{a,v}^s \times w(a) - q_{i,j}^s z_{i,j}^s\right)^2\end{aligned} \tag{27}$$

The quadratic penalty term is linearized according to the characteristics of binary variables as well. This transformation facilitates decomposition by separating decision variables x and z into two sub-problems: sub-route planning **P'x** and service planning **P'z**. The objective function of **P'x** is formulated as follows:

$$\begin{aligned}\min f'(x)_{\rho_2} = &\sum_{s\in S}\sum_{v\in V\cup V^*}\sum_{a\in A_v} c_a x_{a,v}^s + \sum_{s\in S}\sum_{v\in V\cup V^*}\sum_{a\in A_d}\lambda_{i,j}^s x_{a,v}^s \times w(a) + \\ &\frac{\rho_2}{2}\sum_{s\in S}\sum_{(i,j)\in L_d}\left(\sum_{v\in V\cup V^*}\sum_{a\in A_d} x_{a,v}^s \times w(a) - q_{i,j}^s z_{i,j}^s\right)^2,\end{aligned} \tag{28}$$

subject to flow balance constraints (6)–(8), state-changing constraints (10), and consistency constraints (11). The sub-problem **P'x** is also solved using the n-block ADMM framework, where variables $z_{i,j}^s$ are initialized. The formulation of sub-problem **P'z** on service planning is the same formulation as Eq. (20).

Similarly, the sub-problem **SP'x** for the upper bound calculation is formulated as objective function (29):

$$\min f'(x) = \sum_{s\in S}\sum_{a\in A_v} \hat{c_a}' x^s_{a,v} + \tau' \tag{29}$$

$$\hat{c_a}' = \begin{cases} c_a + \lambda^s_{i,j} \times w(a) + \frac{\rho_2}{2}\left(w(a)^2 + 2\left(\mu^s_{v'} - q^s_{i,j} z^s_{i,j}\right) \times w(a)\right), & a \in A_d \\ c_a, & \text{otherwise} \end{cases} \tag{30}$$

$$\tau' = \begin{cases} \frac{\rho_2}{2} \sum_{s\in S}\sum_{(i,j)\in L_d} \left(\mu^s_{v'} - q^s_{i,j} z^s_{i,j}\right)^2, & a \in A_d \\ 0, & \text{otherwise} \end{cases} \tag{31}$$

subject to flow balance constraints (6)–(8), state-changing constraints (10) and consistency constraints (13).

Compared to the lower bound objective function (23), Lagrangian multipliers $\lambda^s_{i,j,v}$ are not included, because consistency constraints (13) remain enforced in the upper bound calculation. Algorithm 1 iteratively updates Lagrangian multipliers and adjusts penalty terms to steer towards the convergence of the upper and lower bounds. The process terminates when the gap falls below a predefined threshold ϵ or the maximum number of iterations $k_{\max}$ is reached.

Algorithm 3 Upper bound calculation.

Step 1: Decompose the initial problem into sub-problems P'x and P'z
Fix master route planning variable $y_{i,j,v}$ obtained from Algorithm 2, Step 3
Relax state-changing constraint Eq. (17) into objective function Eq. (16)
Decompose the initial problem into:
• **P'z** (service planning): Eq. (20), subject to variable domain constraints
• **P'x** (sub-route planning): Eq. (28), subject to constraints (6–8), (10–11)
Linearize quadratic penalty term in **P'x** and solve the sub-problem using ADMM
Obtain **SP'x** problems Eq. (29)
Step 2: Solve sub-problems P'x and P'z
for each scenario $s \in S$ **do**
Solve the service planning Knapsack P'z (Eq. (20)):
Obtain $z^{s,\text{upper}}_{i,j}$ and prepare for **P'x** calculation
Solve the sub-route planning sub-problem P'x:
for each vehicle $v \in V \cup V^*$ **do**
Solve Eq. (29) using DP
Obtain $\lambda^{s,\text{upper}}_{i,j}$ and backtrack optimal path $x^{s,\text{upper}}_{a,v}$
end for
end for
Step 3: Compute the upper bound for iteration k
Calculate upper bound by:
$$UB^{(k)} = \sum_{s\in S}\sum_{v\in V\cup V^*}\sum_{a\in A_v} c_a x^{s,\text{upper}}_{a,v} + \sum_{s\in S}\sum_{v\in V\cup V^*}\sum_{a\in A_d} \lambda^{s,\text{upper}}_{i,j} x^{s,\text{upper}}_{a,v} \times w(a)$$
Obtain the final route and service planning for iteration k

5. Numerical experiments

In this section, we evaluate the performance of the proposed ALR-ADMM-based algorithm through numerical experiments using the 24-node Sioux-Falls network. This network provides a structured yet computationally manageable test-bed to analyze the algorithm's performance under different parameter settings. The evaluation focuses on two aspects. First, the algorithmic performance is assessed by measuring computational efficiency, convergence rate, and solution quality across different configurations. Second, a sensitivity analysis examines the impact of key parameters on solution stability and overall performance. The experiments are implemented in Python and executed on a Windows system with an Intel(R) Core(TM) Ultra 7 155H (3.80 GHz) processor and 32 GB RAM.

5.1. Description and results

Fig. 6 depicts the structure of the Sioux-Falls network, with the cost of each link labeled. In this 24-node configuration, vehicles depart from a common origin (node 13) to pick up passengers at designated demand stops and proceed to a shared destination (node 2). A total of 8 candidate stops are distributed across the network, with 4 originally planned vehicles ready for transit tasks. Table 6 lists the expected departure time windows for each demand stop. In this experiment, as SFT usually operates in low-demand areas, it

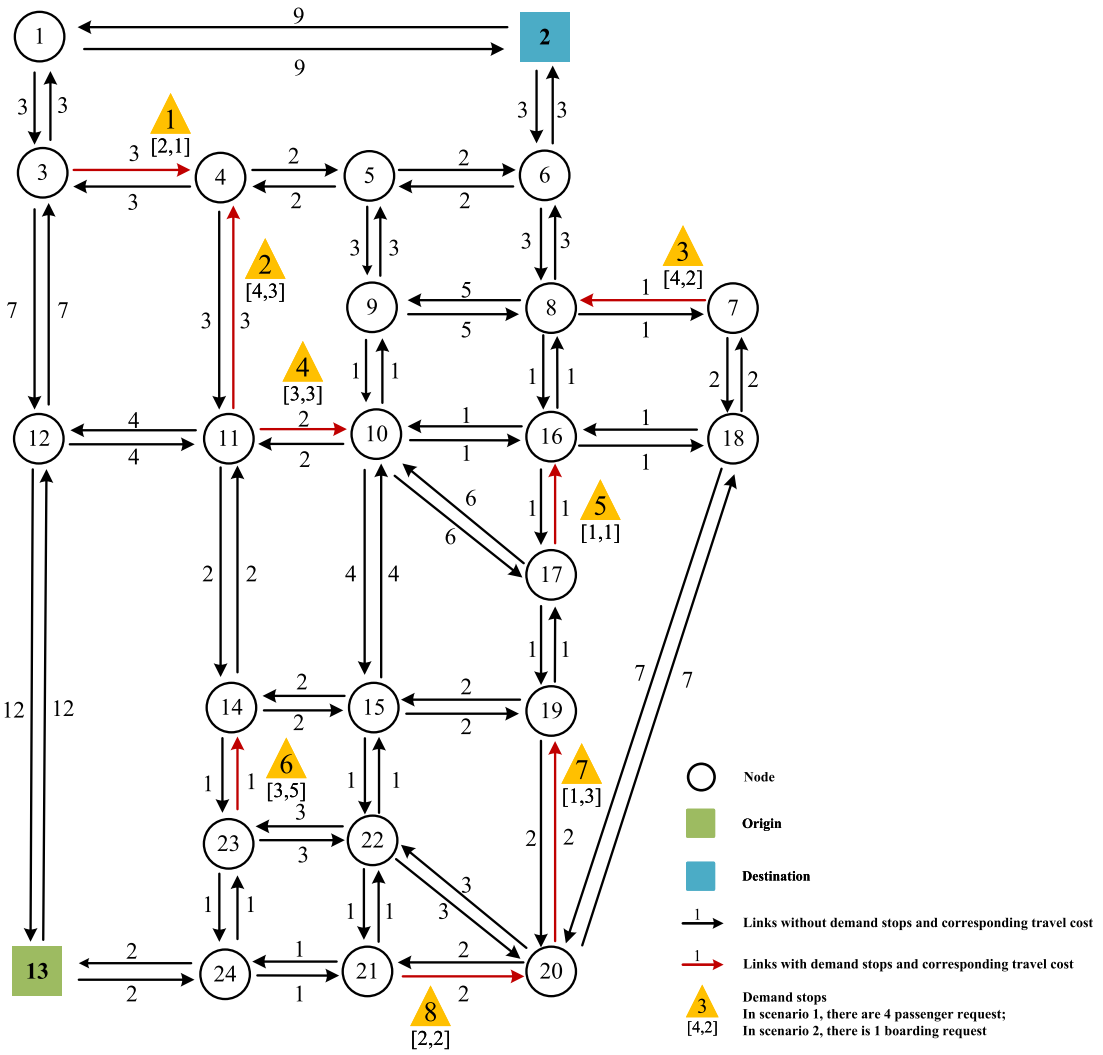


Fig. 6. 24-node Sioux-Falls road network.

Table 6
Expected departure time windows of demand links in the Sioux-Falls case study.

Stop	Source node	Sink node	Earliest departure time	Latest departure time
1	3	4	12	14
2	11	4	6	11
3	7	8	11	13
4	11	10	6	8
5	17	16	8	10
6	23	14	3	5
7	20	19	5	7
8	21	20	3	5

is assumed that link travel times are constant, depending only on segment length and unaffected by external factors, such as boarding time or traffic congestion. Each vehicle starts its journey without any passengers, and the maximum vehicle capacity is set to $Q_v = 6$. The delivery process is limited to 30 time units, and deploying an additional re-employed vehicle incurs a cost of 10 (Yang et al., 2022a,b).

We begin by testing a case with 4 vehicles and 2 demand scenarios. The notation [2,1] in Fig. 6 indicates that this demand stop is associated with two passenger requests in scenario 1 and one passenger request in scenario 2, with all requests sharing the same common destination. For the ALR-ADMM-based method, the penalty parameters are set to $\rho_1 = \rho_2 = 1$, and the maximum number of iterations $k_{\max}$ is limited to 20. Given the characteristics of the LR method, even small differences in Lagrangian multipliers can

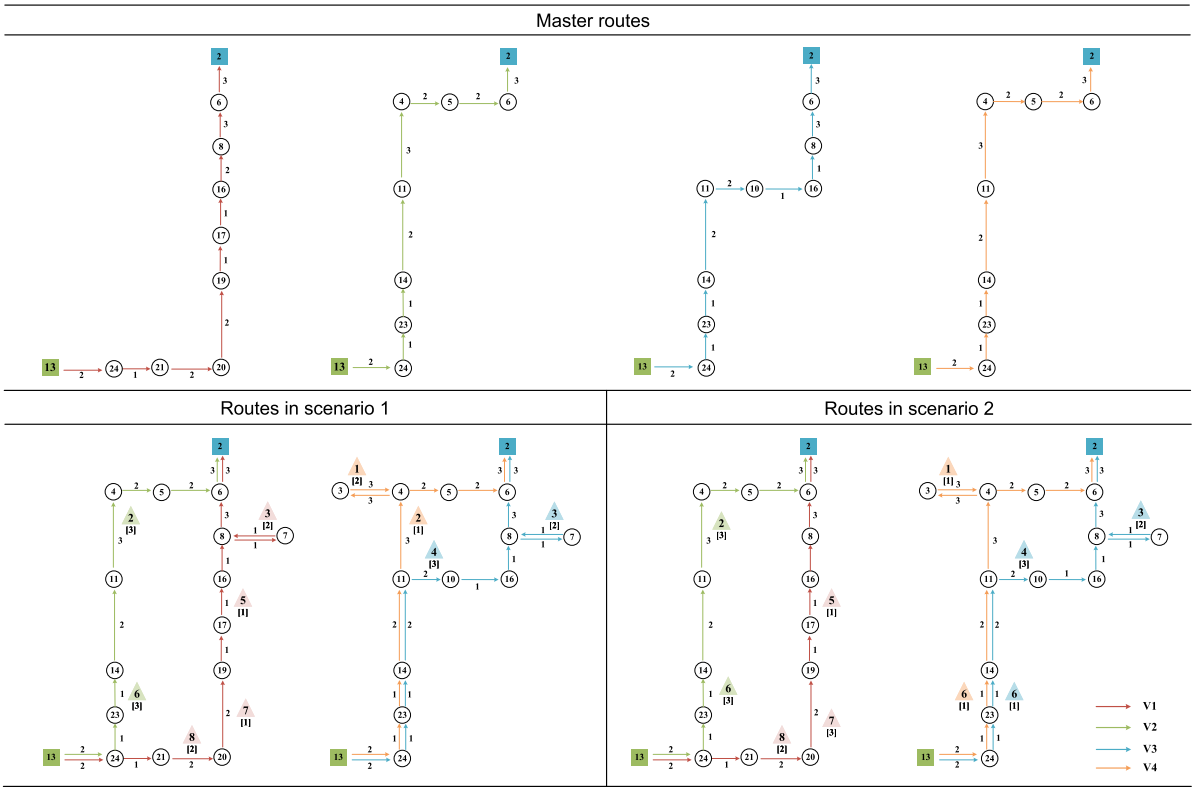


Fig. 7. Visualization of the vehicle routing optimization results for 24-node Sioux-Falls network.

impact the final solution gap. In this study, we consider a solution with Gap $\leq 5\,\%$ to be acceptable, following the criteria provided in Yao et al. (2019).

The optimal solution is obtained by the 5th iteration within 12.54 seconds, with the final Gap = 1.18 %. Fig. 7 demonstrates the planned master routes and sub-routes of the four vehicles under different scenarios. All 40 passenger requests are successfully accommodated within their respective time windows, ensuring that every demand stop is served, i.e., $z^s_{i,j} = 1$ for all demand stops in all scenarios.

As shown in Fig. 7, vehicle 1 picks up four passengers along its master route and detours to node 7 to pick up two additional passengers in scenario 1, reaching its full capacity of 6. Vehicle 2 follows the master route and also serves six passengers. As for vehicle 3, it initially follows the same master route as vehicle 2, but diverts at node 11 and then bends to node 10 to serve demand stop 4. Subsequently, it further detours to pick up two passengers at demand stop 3, carrying a total of five passengers. Vehicle 4 shares the same master route as vehicle 2 but deviates at node 3 to serve two passengers at demand stop 1. While vehicles 1 and 2 are fully loaded, vehicles 3 and 4 have one and three vacant seats, respectively, indicating their potential to accommodate additional passengers.

In scenario 2, both vehicle 1 and vehicle 2 follow their master routes, with vehicle 1 serving six passengers at three demand stops and vehicle 2 picking up six passengers at two demand stops. Both vehicle 3 and vehicle 4 deviate from their master routes, where vehicle 3 detours at node 8, while vehicle 4 deviates to serve demand stop 1. All vehicles are fully loaded except for vehicle 4, which still has four vacant seats, again suggesting potential capacity for additional passengers.

5.2. Computation efficiency

In this subsection, we evaluate the computational efficiency of our proposed ALR-ADMM algorithm by comparing it against the Gurobi optimizer 12.0.0, with a solution gap of 0.0 % and a maximum running time of 30 minutes. If the solution cannot be obtained within this time, we consider the problem infeasible under the given constraints. For reference, our previous experiment showed that the ALR-ADMM algorithm converged to the optimal solution within 12.54 seconds after 5 iterations. In contrast, Gurobi takes over 307.76 seconds to obtain the optimal solution. Despite both methods achieving a similar gap value, the ALR-ADMM algorithm reduces computation time by 95.93 %, demonstrating significant computational efficiency.

To further assess the performance of both methods under varying conditions, we conduct 13 experiments with different passenger demand and service patterns. The results are presented in Table 7. For consistency, settings of the 8 designated demand stops remain

Table 7
Computation efficiency comparison between Gurobi optimizer and ALR-ADMM-based algorithm.

Instance	Number of scenarios	Number of scheduled vehicles	Gurobi			ALR-ADMM				Obj difference from Gurobi (%)	Time reduction over Gurobi (%)
			Number of operating vehicles	Obj	Time (s)	Number of operating vehicles	Obj	Gap (%)	Time (s)		
1	1	1	2	45	62.48	2	45	1.1	4.22	0.0	93.25
2	2	1	4	90	125.10	4	90	0.0	3.98	0.0	96.82
3	2	2	4	72	131.61	4	72	0.6	3.86	0.0	97.07
4	2	3	5	86	200.77	5	86	4.4	26.92	0.0	86.59
5	2	4	8	146	307.76	8	148	1.2	12.54	1.4	95.93
6	3	1	6	142	184.31	6	142	1.8	6.18	0.0	96.65
7	3	2	6	116	187.63	6	116	0.6	5.47	0.0	97.08
8	3	3	9	174	318.89	9	180	2.6	11.01	3.4	96.55
9	3	4	10	195	422.94	10	195	1.9	40.27	0.0	90.48
10	4	1	4	84	133.02	4	84	0.0	1.74	0.0	98.69
11	4	2	8	154	268.05	8	154	0.2	10.66	0.0	96.02
12	4	3	9	176	426.14	9	176	1.6	16.49	0.0	96.13
13	4	4	13	230	644.35	13	236	3.5	46.19	2.6	92.83

* Gap (%) $= (UB - LB)/UB \times 100\,\%$;
* Obj difference from Gurobi (%) $= (\text{Obj}_{\text{ALR-ADMM}} - \text{Obj}_{\text{Gurobi}})/\text{Obj}_{\text{Gurobi}} \times 100\,\%$;
* Time reduction over Gurobi (%) $= (\text{Time}_{\text{Gurobi}} - \text{Time}_{\text{ALR-ADMM}})/\text{Time}_{\text{Gurobi}} \times 100\,\%$

unchanged as those used in the basic experiment, and the capacity for all vehicles is fixed at $Q_v = 6$. In Table 7, the column "Number of scheduled vehicles" specifies the number of originally planned vehicles for each scenario, whereas the "Number of operating vehicles" columns under both Gurobi and ALR-ADMM indicate the number of vehicles actually deployed to provide service across all scenarios, including any re-employed vehicles. Objective function values derived from Eq. (1) are documented under the "Obj" columns. For the ALR-ADMM method, the "Gap (%)" column quantifies the optimality gaps between upper and lower bounds at convergence. Computational times in seconds for both algorithms are provided under the "Time (s)" columns. To evaluate computational reliability and efficiency, the column labeled "Obj difference from Gurobi (%)" indicates the relative difference in objective values, while the "Time reduction over Gurobi (%)" column highlights the percentage reduction in computation time achieved by ALR-ADMM compared to Gurobi.

The results show that the proposed ALR-ADMM algorithm delivers substantial computational gains, achieving an average 94.93 % reduction in runtime across all instances (relative to Gurobi). Importantly, this advantage is robust to scale: the reduction remains high from instance 1 (with 1 vehicle and 1 scenario; 93.25 %) to instance 13 (with 4 vehicles and 4 scenarios; 92.83 %), indicating strong scalability for large SFT planning problems. Despite the considerable speedups, solution quality is preserved, as objective values are nearly identical to Gurobi's, with a mean optimality difference of 0.57 %. While three instances exhibit small cost differences, ALR-ADMM consistently attains reliable convergence with drastic time savings, meeting key requirements for real-time decision support in large transit networks where computational tractability is critical.

5.3. Sensitivity analyses

Building on the basic experiment in Section 5.1, this subsection further examines the influence of demand stop selection, vehicle capacity, and penalty parameters ρ_1 and ρ_2 on the optimization results. To ensure a controlled analysis, each parameter is varied independently while keeping others constant.

5.3.1. Impact of demand stop selection

Intuitively, increasing the number of demand stops provides greater flexibility in determining a fixed master route. In the basic experiment, all 8 candidate stops are treated as demand stops, and both master routes and sub-routes are successfully planned. This sensitivity analysis investigates the impact of demand stops by varying their selection. For vehicle 1, stops 5, 7, and 8 are located on its master route, while stop 3 appears on its sub-links. Similarly, for vehicle 3, stops 4 and 6 are part of its master route, while stop 3 is on sub-links. Specifically, we remove stop 7 for vehicle 1 to observe the effect on the master route, and remove stop 3 for vehicle 3 to assess the impact on the sub-route. Passenger requests and departure time windows at other stops remain unchanged.

The optimal solution is obtained after 5 iterations, with the final Gap = 0.91 %. The optimized results are displayed in Fig. 8. Compared to Fig. 7, the master routes of all vehicles remain unchanged, indicating the robustness of the algorithm in maintaining consistent route planning. However, as expected, vehicle 3 no longer deviates due to the removal of stop 3, which indicates that demand variations on the sub-links only affect local deviations, rather than the master route structure. These findings illustrate the ability of the algorithm to adapt to changing demand conditions while maintaining stability in master route planning, which is crucial for its practical application in real-world scenarios.

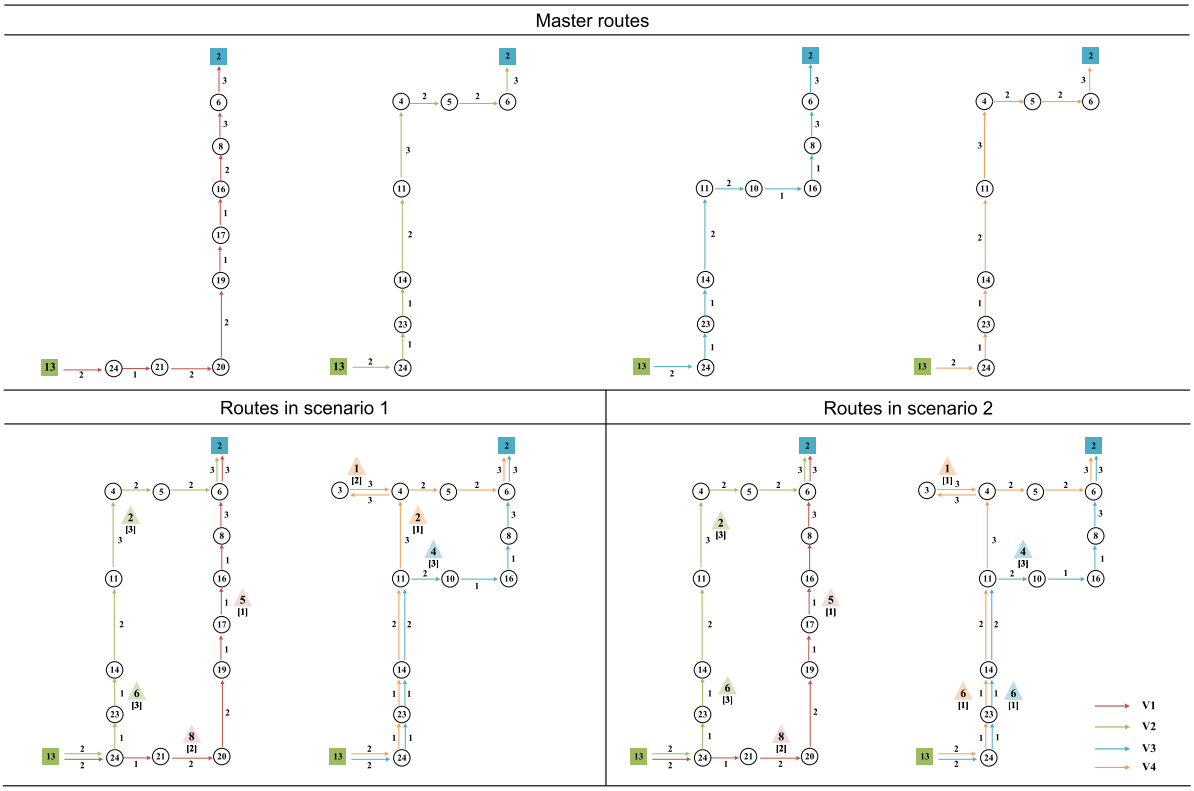


Fig. 8. Impact of demand stop selection on the vehicle routing optimization results for the Sioux-Falls network.

Table 8
Impact of originally scheduled vehicle capacity on their service for the Sioux-Falls network.

Vehicle capacity	Unserved stops	Unserved scenarios	Unsatisfied requests	Routing cost	Re-employed penalty cost	Additional cost (%)
4	3	1	4	243	30	10.99
	3	2	2			
	5	2	1			
	8	2	1			
5	3	2	2	214	20	8.55
	5	1	1			
	5	2	1			
6	–	–	0	148	0	0
7	–	–	0	150	0	0
8	–	–	0	148	0	0

5.3.2. Impact of vehicle capacity

The capacity of vehicles is crucial in determining both loading efficiency and the number of operating vehicles required. As discussed in Section 3, re-employed vehicles are introduced when the originally scheduled vehicles are insufficient to accommodate all passengers. However, the use of re-employed vehicles incurs additional costs, making it essential to optimize the capacity of originally scheduled vehicles to minimize unnecessary expenses. To investigate the impact of vehicle capacity, we vary the capacity Q_v from 4 to 8 and present the results in Table 8 and Fig. 9. The columns "Unserved stops" and "Unserved scenarios" refer to the stops and scenarios that fail to provide service, requiring re-employed vehicles instead. The values in the "Additional cost" column represent the percentage of re-employed penalty cost relative to the total operating cost.

As shown in Table 8, when $Q_v = 4$, only five stops are fully served by originally scheduled vehicles in all scenarios. Stop 3 is entirely unserved, while stops 5 and 8 receive partial service in scenario 2. Due to capacity limitations, three re-employed vehicles are needed to accommodate eight additional passengers, leading to a penalty cost of 30, which accounts for 10.99 % in the total operating cost. Besides, the additional routing cost associated with re-employed vehicles further increases the total expense.

As defined in constraint (9), when variable $z^s_{i,j} = 1$, vehicles must serve all passengers at the demand stop on link (i, j). Consequently, the unserved stops tend to fall into two categories: (1) master stops (i.e., stops on master routes) with relatively low passenger

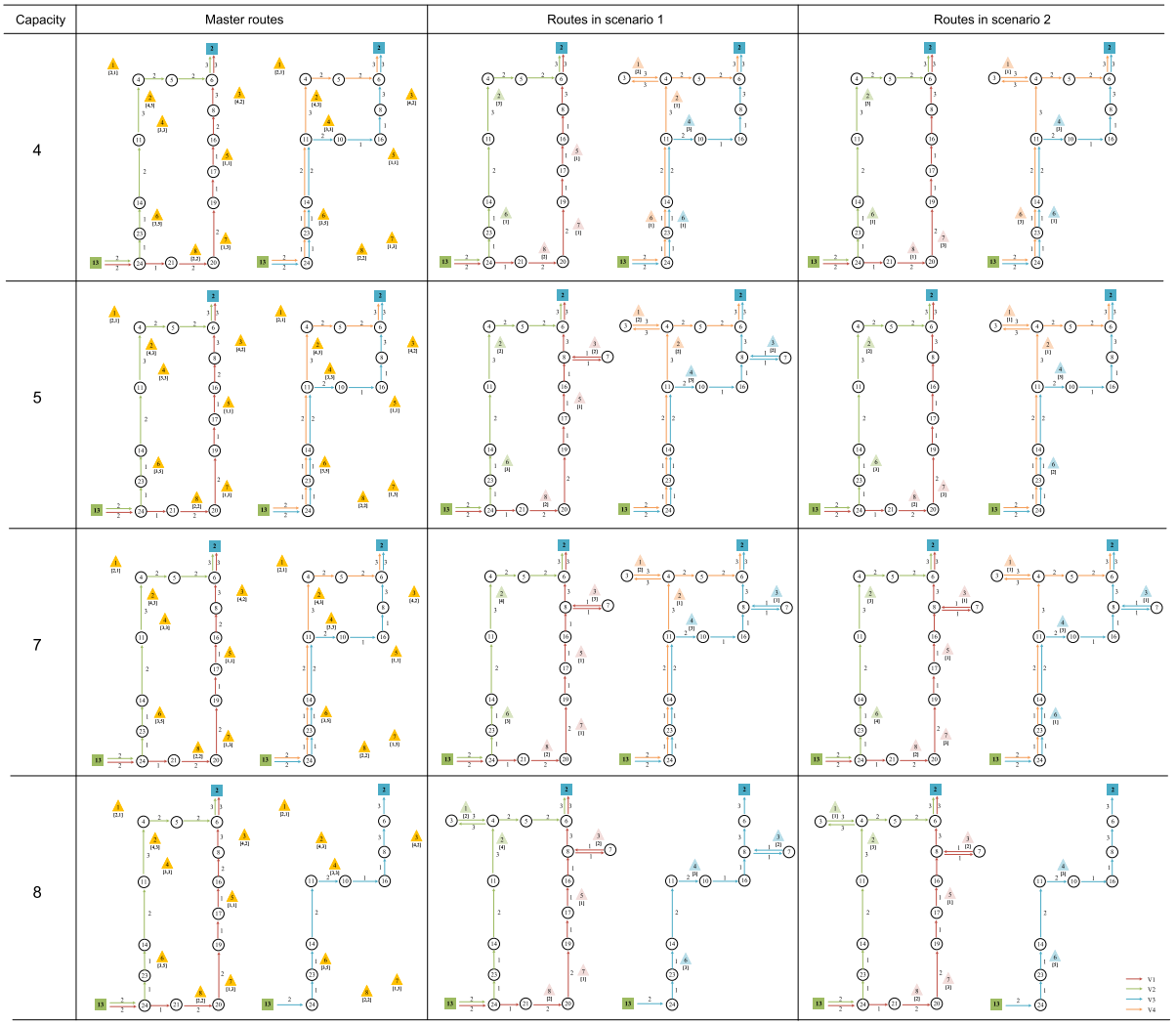


Fig. 9. Impact of vehicle capacity on vehicle routing optimization results for the Sioux-Falls network.

Table 9
Impact of quadratic penalty parameters ρ_1 and ρ_2 on computing time and solution quality.

Instance	ρ_1	ρ_2	Iterations	Satisfied all requests	Gap (%)
1	0.0	0.0	5	✓	4.38
2	0.2	0.2	3	✓	0.41
3	0.4	0.4	11	✓	1.11
4	0.6	0.6	4	✓	2.86
5	0.8	0.8	10	✓	3.18
6	1.0	1.0	8	✓	1.18
7	1.2	1.2	30	✓	3.54
8	1.4	1.4	60	✓	1.62
9	1.6	1.6	144	✓	2.75
10	1.8	1.8	373	✓	4.00
11	2.0	2.0	800+	✓	non-convergence

* Gap (%) $= (UB - LB)/UB \times 100\,\%$

Table 10
Impact of quadratic penalty parameter ρ_1 on computing time and solution quality.

Instance	ρ_1	ρ_2	Iterations	Satisfied all requests	Gap (%)
1	0.0	1.0	17	×	1.18
2	0.2	1.0	7	✓	0.60
3	0.4	1.0	11	✓	1.93
4	0.6	1.0	4	✓	3.30
5	0.8	1.0	10	✓	3.62
6	1.0	1.0	8	✓	1.18
7	1.2	1.0	30	✓	3.09
8	1.4	1.0	60	✓	0.50
9	1.6	1.0	144	✓	0.47
10	1.8	1.0	373	✓	0.57
11	2.0	1.0	800+	✓	non-convergence

* Gap (%) = $(UB - LB)/UB \times 100\,\%$

requests, and (2) sub-stops (i.e., stops on sub-links) with excessively high demand. The former category, such as stop 5 and stop 8, may be deprioritized if other master stops have higher demand. Specifically, vehicles prioritize serving high-demand master stops first, and only remaining seats are allocated to lower-demand stops, often resulting in service gaps. The latter category, such as stop 3, has demand levels that could disrupt the consistency of master routes. In these cases, serving all passengers at these sub-stops may compromise the ability to meet the demand at other master stops, leading to their exclusion. This behavior indicates that the algorithm prioritizes fulfilling passenger requests at stops with higher demand along master routes for cost minimization. A similar pattern is observed when $Q_v = 5$, where two re-employed vehicles are used with a penalty cost of 20.

When $Q_v \geq 6$, vehicles can serve all passengers without the need for re-employed vehicles. The master routes of the vehicles remain consistent up to $Q_v = 8$, at which point three vehicles suffice to meet the total demand. Under this setup, vehicle 4 stays in the depot rather than performing a task. Optimizing vehicle capacity allows the SFT system to minimize unnecessary costs while ensuring efficient service. By selecting an appropriate capacity, operators can reduce dependency on re-employed vehicles and improve cost-effectiveness in real-world applications.

5.3.3. Impact of penalty parameters

The penalty parameters ρ_1 and ρ_2 control the degree to which violation of pickup constraints is penalized. In the basic experiment, both ρ_1 and ρ_2 are set to 1. In this subsection, we vary the values of ρ_1 and ρ_2 from 0 to 2 in increments of 0.2, while keeping all other settings identical to those in Subsection 5.1.1.

The effects of different penalty values on the number of iterations and service strategy are presented in Table 9. The results reveal that as ρ_1 and ρ_2 gradually increase, their impact on the pickup service remains relatively modest, leading to similar solution gaps. This suggests a saturation effect, where further increases in penalties yield decreasing improvements. However, when $\rho_1 > 1$ and $\rho_2 > 1$, the number of iterations required to reach convergence increases significantly. Beyond $\rho_1 > 1.6$ and $\rho_2 > 1.6$, the algorithm struggles to converge, highlighting its sensitivity to large penalty values. This occurs because the algorithm prioritizes satisfying the pick-up constraints to an extreme degree, thereby potentially overriding the route-planning objective itself. As a result, the search for an optimal solution becomes inefficient and leads to slower convergence or even stagnation. Accordingly, the choice of penalty values has a direct impact on algorithm efficiency. Properly balancing these parameters is essential to avoid excessive computational burden while maintaining a feasible solution.

Further, we adjust ρ_1 or ρ_2 independently to test their impact on computation time and solution quality, as shown in Table 10 and Table 11. In each test, the other penalty parameter remains fixed at 1.0. We observe that the outcomes are identical when adjusting ρ_1 or ρ_2 independently, as both parameters penalize the same state-changing constraint. Relative to the case where both penalty parameters are adjusted simultaneously, the overall solution quality remains similar, but the convergence behavior differs between the two analyses. When adjusting only one penalty parameter while keeping the other fixed, the number of iterations required for convergence varies more inconsistently, particularly in the first two instances where the adjusted parameter increases from 0.0 to 0.2. This is likely due to the imbalance in constraint enforcement between the lower and upper bounds, leading to fluctuations in ALR-ADMM updates. In contrast, when both penalty terms are increased together, their scaling effect generally stabilizes the iterative process, leading to smoother convergence. Additionally, when $\rho_1 > 1.6$ or $\rho_2 > 1.6$, the algorithm struggles to converge, regardless of whether the parameters are adjusted separately or together. This suggests that excessively large penalties may over-restrict constraint violations, potentially suppressing the optimization process and preventing convergence. In instance 11, the algorithm fails to converge when only one penalty parameter is increased while the other remains fixed. This result highlights the risk of excessive penalties disrupting the iterative updates and preventing the algorithm from reaching a feasible solution. Thus, these results indicate that the optimization is most efficient when $\rho_1 <= 1$ and $\rho_2 <= 1$, as this prevents excessive iterations while maintaining solution quality. Appropriate selection of these penalty parameters is critical to achieving a balanced enforcement of constraints, avoiding insufficient penalties that lead to sub-optimal solutions with unsatisfied passenger requests and excessively large penalties that lead to convergence issues.

Table 11
Impact of quadratic penalty parameter ρ_2 on computing time and solution quality.

Instance	ρ_1	ρ_2	Iterations	Satisfied all requests	Gap (%)
1	1.0	0.0	17	×	1.18
2	1.0	0.2	7	✓	0.60
3	1.0	0.4	11	✓	1.93
4	1.0	0.6	4	✓	3.30
5	1.0	0.8	10	✓	3.62
6	1.0	1.0	8	✓	1.18
7	1.0	1.2	30	✓	3.09
8	1.0	1.4	60	✓	0.50
9	1.0	1.6	144	✓	0.47
10	1.0	1.8	373	✓	0.57
11	1.0	2.0	800+	✓	non-convergence

* Gap (%) $= (UB - LB)/UB \times 100\,\%$

Table 12
Expected departure time windows of demand stops in the West Jordan case study.

Stop	Earliest departure time	Latest departure time	Stop	Earliest departure time	Latest departure time
1	27	34	9	19	21
2	42	46	10	12	16
3	18	22	11	26	30
4	25	29	12	2	6
5	17	19	13	4	8
6	24	28	14	22	26
7	31	35	15	5	9
8	9	13	16	17	19

6. Case study: West Jordan, Utah

To evaluate the practical applicability of the proposed ALR-ADMM algorithm for the SFT routing problem, we apply it to a real-world case study on the West Jordan network in Utah, USA. This case study involves a significantly larger network, incorporating realistic traffic conditions and operational constraints.

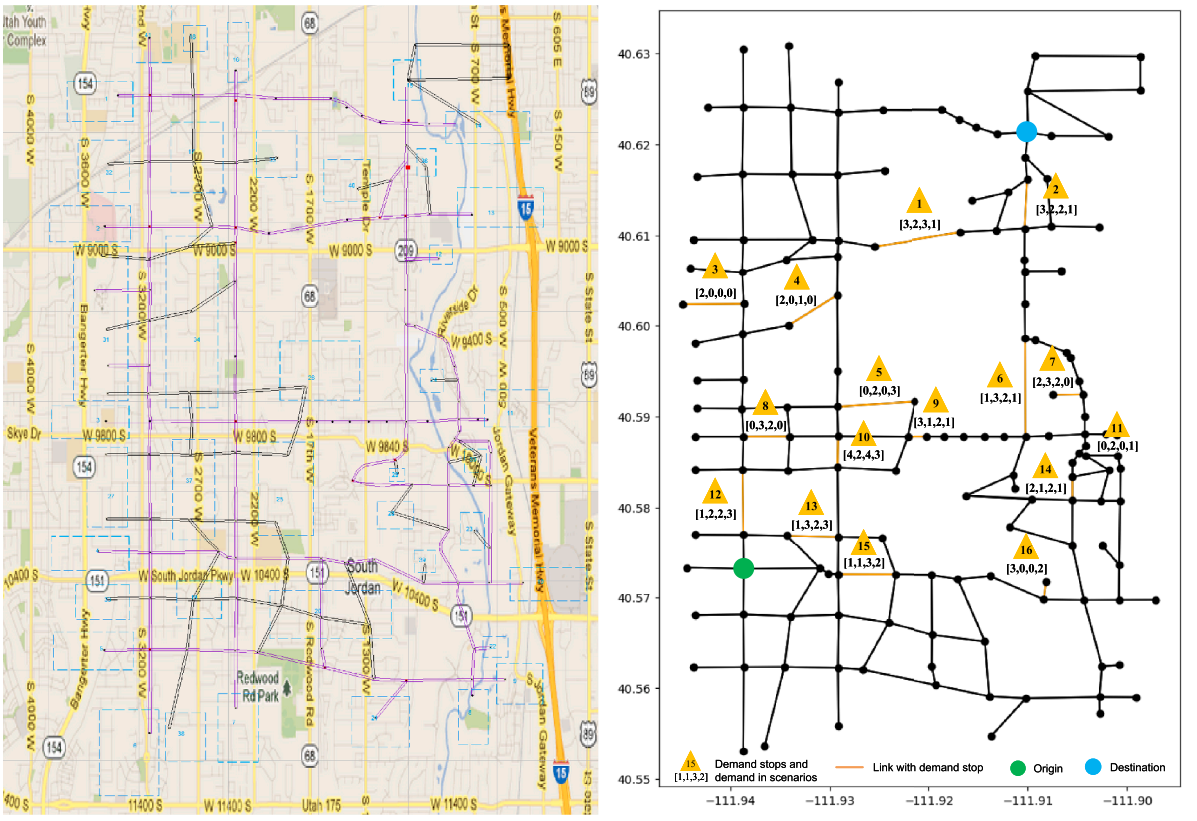
6.1. Case settings

The physical structure of the West Jordan road network is shown in Fig. 10. The network consists of 152 nodes and 388 road segments, with travel time proportional to the road segment length. Four demand scenarios are considered, with a total of 105 passenger requests distributed across 16 candidate stops, marked as orange links and triangles in Fig. 10. For each scenario, four vehicles depart from the same depot and head towards a shared destination. Inspired by the zone-based strategy of Yang et al. (2020), we adopt a corridor-based design to enhance computational tractability (Mancini and Gansterer, 2022). As shown in Fig. 11, each vehicle is assigned to a predefined service corridor. Corridors are delineated along arterial roads that approximate the principal axes of passenger demand. Intentional overlaps between adjacent corridors create shared high-demand stops where vehicles from different corridors can converge, strengthening robustness through redundant capacity and flexible vehicle deployment. The expected departure time windows at each demand stop are presented in Table 12. Link travel times are solely dependent on segment length and unaffected by external factors. The entire routing process is restricted to 60 minutes to reflect practical operational limits. Each vehicle starts empty, with individual vehicle capacities set as follows: $Q_{v_1} = 8$, $Q_{v_2} = 10$, $Q_{v_3} = 6$ and $Q_{v_4} = 8$. Both penalty parameters ρ_1 and ρ_2 are set to 1 in the ALR-ADMM algorithm. The optimization terminates after a maximum of 30 iterations.

6.2. Computational performance and results analysis

Under these settings, the dimensionality of the variables exceeds 2.2×10^9, resulting in substantial memory consumption and preventing Gurobi from completing the solving process. However, the proposed ALR-ADMM algorithm successfully obtains an optimal solution by the 6th iteration within 583.24 seconds, with a solution gap of 1.51 % as shown in Fig. 12. Fig. 13 illustrates the planned master routes for the four vehicles, while Figs. 14 and 15 provide the corresponding sub-routes for each vehicle. In Fig. 15, triangles denote served demand stops, with the numbers inside representing the boarding passengers.

The corridor-based design offers a practical, scalable basis for implementing DP, enabling a structured and efficient vehicle-route assignment. In our case study, all four vehicles are assigned optimal master routes, yielding stable operations that satisfy all required service time windows. All 105 passenger requests are served without deploying supplemental (re-employed) vehicles, indicating that the selected vehicle capacities are well matched to demand. Among the four vehicles, vehicle 3 strictly follows its master route, whereas the others make controlled deviations to serve passengers at optional sub-link stops. Notably, stop 1, located at the



(a) Real-world example of the road network (b) Schematic representation of the network and case study settings

Fig. 10. West Jordan road network structure.

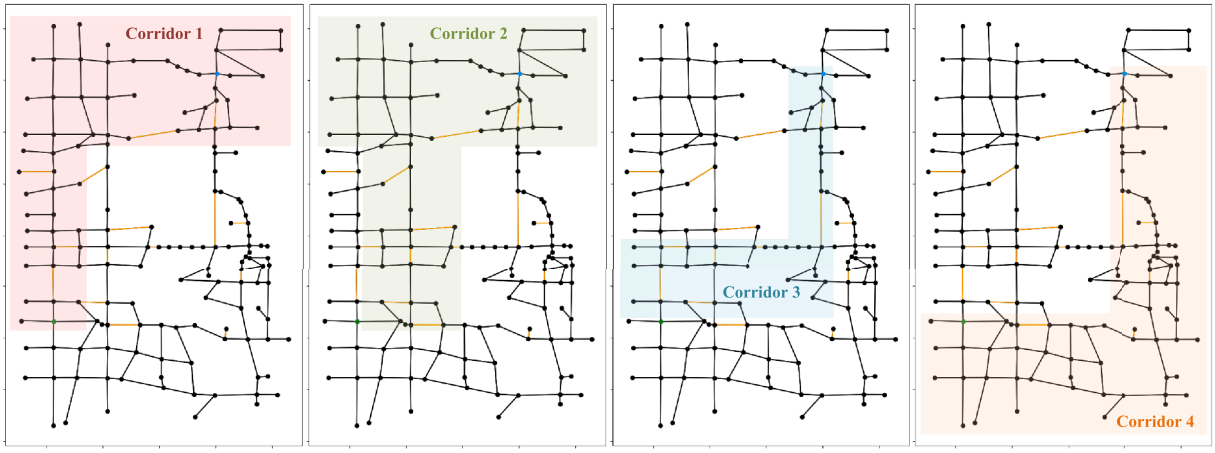


Fig. 11. Predefined vehicle service corridors in the West Jordan network.

intersection of corridors 1 and 2, is served by vehicles from these two corridors in different scenarios, confirming the effectiveness of intentional corridor overlap. This flexibility in route design enhances accessibility–especially for riders at non-master-route stops–ensuring comprehensive coverage even in less accessible areas.

In this experiment, most vehicles operate below full capacity, suggesting the potential to serve more passengers along or near the master routes. This provides an opportunity for flexible optimization in future operations, where additional passengers could be accommodated without significantly altering the planned vehicle routes in accordance with the service schedules. One possible refinement is to introduce additional candidate stops within each service corridor, which would enhance service coverage without

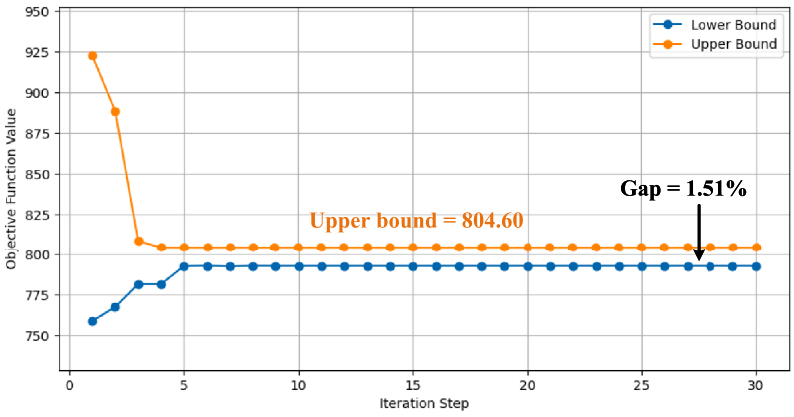


Fig. 12. Upper and lower bound progression over iterations for the West Jordan network.

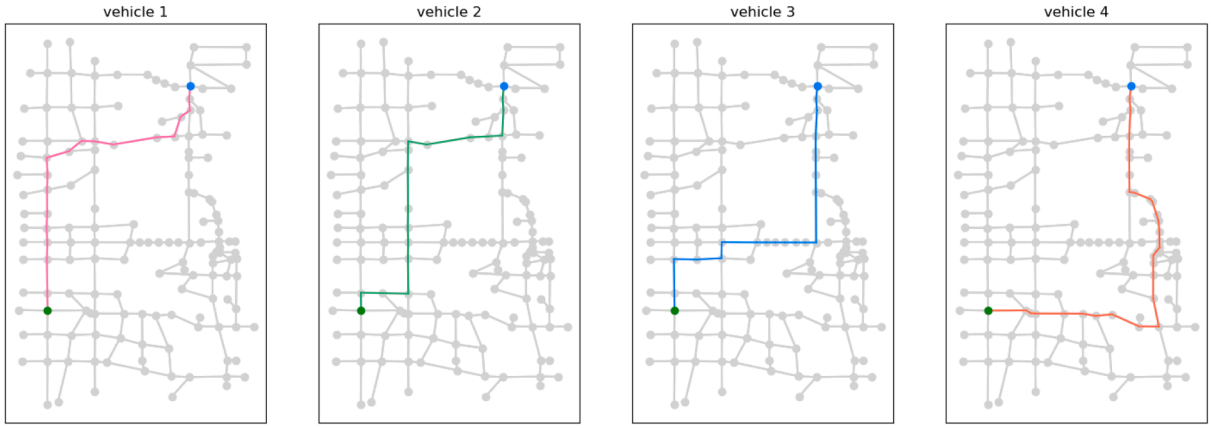


Fig. 13. Planned master routes in the West Jordan network.

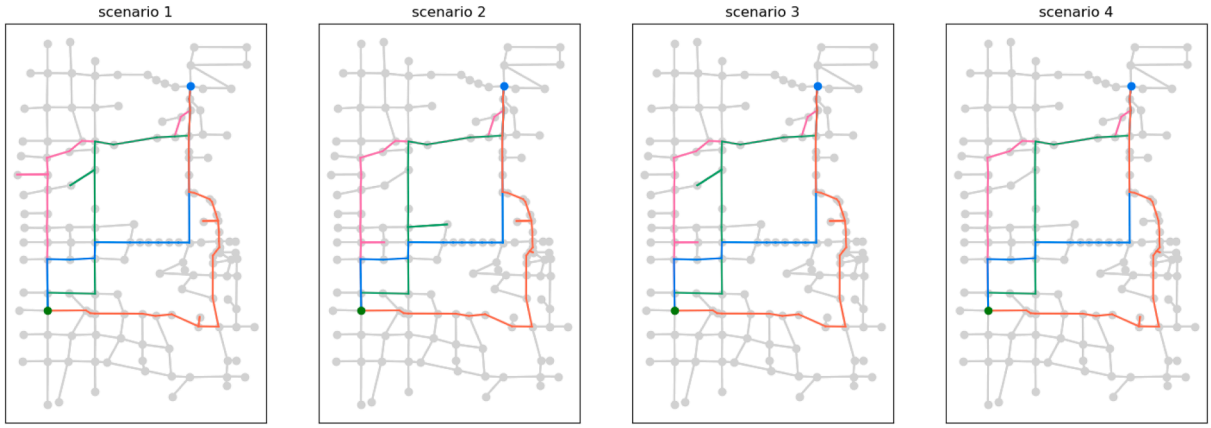


Fig. 14. Planned sub-routes in the West Jordan network.

disrupting the consistency of the routes. Furthermore, slightly relaxing the departure time windows for downstream stops could offer additional flexibility, allowing vehicles to adjust their routes dynamically and thus serve more passengers effectively.

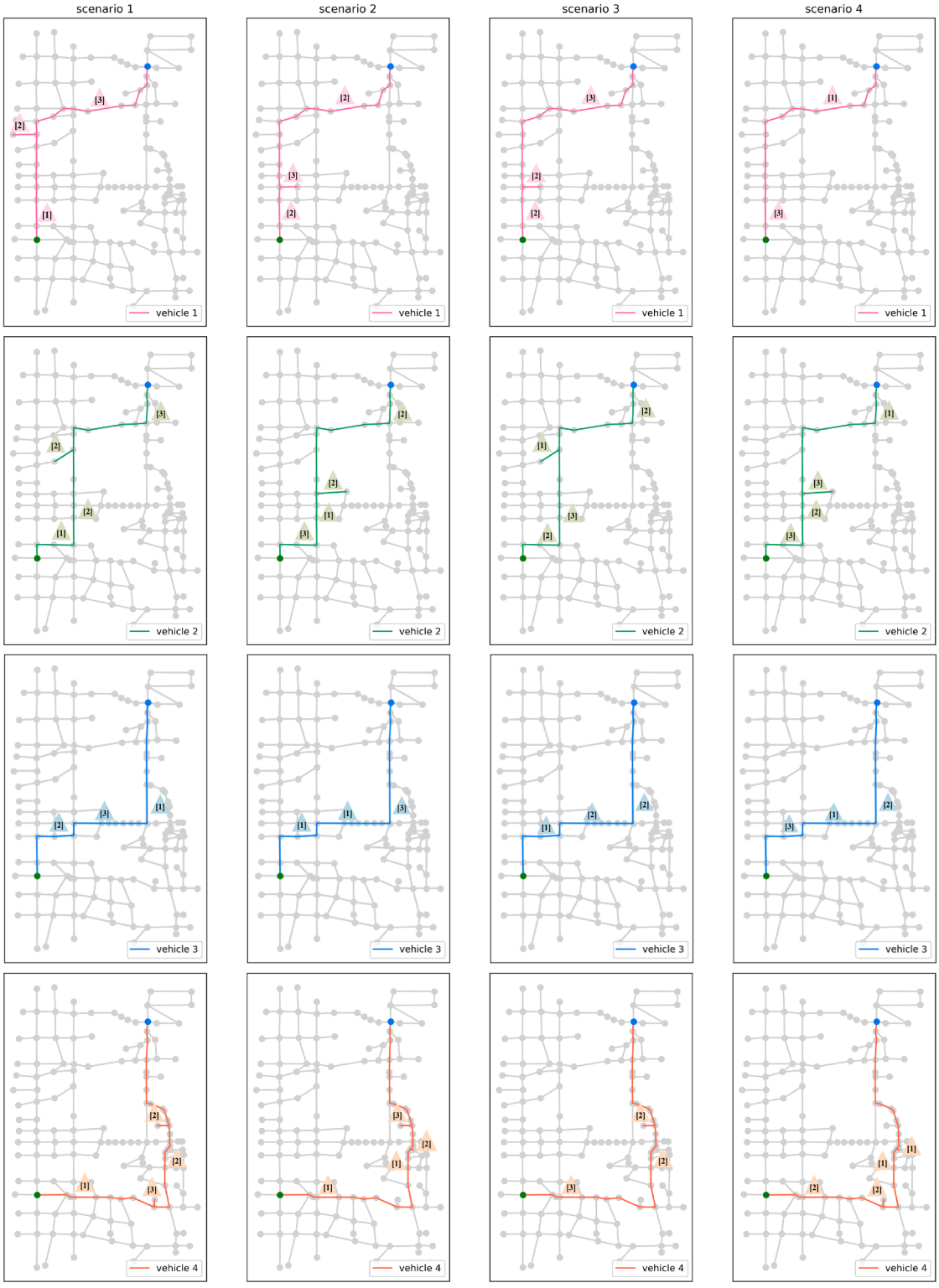


Fig. 15. Detailed sub-routes for each vehicle with served demand stops in the West Jordan network.

7. Conclusions

The growing complexity of urban mobility calls for innovative transit solutions that balance flexibility and reliability. Traditional fixed-route services and purely demand-responsive transit face challenges in addressing the first- and last-mile problem efficiently. Semi-Flexible Transit (SFT), which integrates fixed master routes with dynamically adjusted sub-routes, offers a promising hybrid solution. However, existing models often fail to jointly optimize fixed and flexible components under fluctuating demands, leading to inefficiencies and operational uncertainty.

To address this gap, we proposed a scenario-based optimization approach that collectively considers routing and service planning across both tactical fixed and operational flexible components. By formulating the problem as a multi-scenario Vehicle Routing Problem with Time Windows (VRPTW) and solving it with Augmented Lagrangian Relaxation (ALR) under the Alternating Direction Method of Multipliers (ADMM) algorithm, our model efficiently coordinated master routes, sub-routes, and service strategies while minimizing total operational cost and endogenously determining the required fleet size. Compared to the Gurobi solver, the proposed algorithm reduced computation time by an average of 94.93 %, with an average optimality difference of 0.57 % on the 24-node network. In a large-scale case with over 2.2 billion variables, the algorithm solved the problem in 583 seconds with a 1.51 % gap, serving all passenger requests without deploying additional vehicles.

While this study advances SFT optimization, several limitations offer avenues for future research. First, although our model provides a robust day-ahead plan by integrating tactical and operational decisions based on reserved demand, a valuable extension would be to develop a real-time dynamic route adjustment module. Building on the optimized baseline generated by our framework, such a module could make intra-day adjustments to handle operational uncertainties such as last-minute reservations, cancellations, or unexpected traffic delays, further enhancing system responsiveness and efficiency. Second, the settings of constant travel times is reasonable in the low-demand SFT context in our experiments, but could be extended to better capture operations in more complex traffic environments. Future work could implement the scenario- and time-dependent model, where link travel time is a function of both the scenario and the time of entry (Lu et al., 2022). Finally, the current framework is primarily designed for many-to-one systems with a single common destination. Extending the model to accommodate many-to-many passenger flows would significantly broaden its applicability to more diverse urban transit networks.

Disclosure statement

No potential conflict of interest was reported by the authors.

CRediT authorship contribution statement

Yating Liu: Writing – review & editing, Writing – original draft, Visualization, Software, Conceptualization; **Ziyulong Wang:** Writing – review & editing, Validation, Conceptualization; **Oded Cats:** Writing – review & editing, Supervision; **Xin Pei:** Writing – review & editing; **Pan Shang:** Supervision, Software, Methodology, Funding acquisition.

Data availability

The data that has been used is confidential.

Declaration of competing interest

The 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 research was supported by the Beijing Natural Science Foundation [Grant L241081, L241080, 824201, and L251010], the National Natural Science Foundation of China [Grants 72471021 and 72001020].

References

Archetti, C., Speranza, M.G., Weyland, D., 2018. A simulation study of an on-demand transportation system. Int. Transact. Operat. Res. 25 (4), 1137–1161.
Bakas, I., Drakoulis, R., Floudas, N., Lytrivis, P., Amditis, A., 2016. A flexible transportation service for the optimization of a fixed-route public transport network. Transp. Res. Procedia 14, 1689–1698.
Berrada, J., Poulhès, A., 2021. Economic and socioeconomic assessment of replacing conventional public transit with demand responsive transit services in low-to-medium density areas. Transport. Res. Part A: Policy Pract. 150, 317–334.
Bruni, M.E., Guerriero, F., Beraldi, P., 2014. Designing robust routes for demand-responsive transport systems. Transport. Res. Part E: Logist. Transport. Rev. 70, 1–16.
Calabrò, G., Araldo, A., Oh, S., Seshadri, R., Inturri, G., Ben-Akiva, M., 2023. Adaptive transit design: optimizing fixed and demand responsive multi-modal transportation via continuous approximation. Transport. Res. Part A: Policy Pract. 171, 103643.
Chen, P.W., Nie, Y.M., 2017. Analysis of an idealized system of demand adaptive paired-line hybrid transit. Transport. Res. Part B: Methodolog. 102, 38–54.
Chen, X., Wang, Y., Wang, Y., Qu, X., Ma, X., 2021. Customized bus route design with pickup and delivery and time windows: model, case study and comparative analysis. Expert Syst. Appl. 168, 114242.
Cordeau, J.-F., Laporte, G., 2003. The dial-a-ride problem (DARP): variants, modeling issues and algorithms. Q. J. Belg. Fren. Ital. Operat. Res. Societ. 1 (2).

Crainic, T.G., Errico, F., Malucelli, F., Nonato, M., 2012. Designing the master schedule for demand-adaptive transit systems. Ann. Oper. Res. 194 (1), 151–166.
Daganzo, C.F., 1984. Checkpoint dial-a-ride systems. Transport. Res. Part B: Methodolog. 18 (4), 315–327.
Errico, F., Crainic, T.G., Malucelli, F., Nonato, M., 2013. A survey on planning semi-flexible transit systems: methodological issues and a unifying framework. Transport. Res. Part C: Emerg. Technolog. 36, 324–338.
Flusberg, M., 1976. An innovative public transportation system for a small city: the Merrill, Wisconsin, case study. Transport. Res. Board 606, 54–59.
Frei, C., Hyland, M., Mahmassani, H.S., 2017. Flexing service schedules: assessing the potential for demand-adaptive hybrid transit via a stated preference approach. Transport. Res. Part C: Emerg. Technolog. 76, 71–89.
Guo, H., Wang, Y., Shang, P., Yan, X., Guan, Y., 2023. Customised bus route design with passenger-to-station assignment optimisation. Transportmetr. A: Transp. Sci. 20 (3), 2214631.
Guo, R., Guan, W., Zhang, W., 2018. Route design problem of customized buses: mixed integer programming model and case study. J. Transport. Eng. Part A: Syst. 144 (11), 04018069.
Huang, A., Dou, Z., Qi, L., Wang, L., 2020. Flexible route optimization for demand-responsive public transit service. J. Transport. Eng. Part A: Syst. 146 (12), 04020132.
Jaw, J.-J., Odoni, A.R., Psaraftis, H.N., Wilson, N. H.M., 1986. A heuristic algorithm for the multi-vehicle advance request dial-a-ride problem with time windows. Transport. Res. Part B: Methodolog. 20 (3), 243–257.
Jorgensen, R.M., Larsen, J., Bergvinsdottir, K.B., 2007. Solving the dial-a-ride problem using genetic algorithms. J. Operat. Res. Soc. 58 (10), 1321–1331.
Koffman, D., 2004. Operational Experiences with Flexible Transit Services. Vol. 53 of TCRP Synthesis Report. Transportation Research Board.
Kohl, N., Madsen, O. B.G., 1997. An optimization algorithm for the vehicle routing problem with time windows based on lagrangian relaxation. Oper. Res. 45 (3), 395–406.
Leffler, D., Burghout, W., Cats, O., Jenelius, E., 2024. An adaptive route choice model for integrated fixed and flexible transit systems. Transportmetr. B: Transp. Dyn. 12 (1), 2303047.
Leffler, D., Burghout, W., Jenelius, E., Cats, O., 2021. Simulation of fixed versus on-demand station-based feeder operations. Transport. Res. Part C: Emerg. Technolog. 132, 103401.
Li, X., Liu, W., Qiao, J., Li, Y., Hu, J., 2023. An enhanced semi-flexible transit service with introducing meeting points. Netw. Spat. Econ. 23 (3), 487–527.
Li, X., Wang, T., Xu, W., Hu, J., 2021. A novel model for designing a demand-responsive connector (DRC) transit system with consideration of users' preferred time windows. IEEE Trans. Intell. Transp. Syst. 22 (4), 2442–2451.
Liaw, C.-F., White, C.C., Bander, J., 1996. A decision support system for the bimodal dial-a-ride problem. IEEE Transact. Syst. Man Cybernet. Part A: Syst. Human. 26 (5), 552–565.
Liu, B., Ji, Y., Cats, O., 2025. Integrating ride-hailing services with public transport: a stochastic user equilibrium model for multimodal transport systems. Transportmetr. A: Transp. Sci. 21 (1), 2236240.
Liu, T., Ceder, A.A., 2015. Analysis of a new public-transport-service concept: customized bus in China. Transp. Policy (Oxf.) 39, 63–76.
Lu, J., Nie, Q., Mahmoudi, M., Ou, J., Li, C., Zhou, X., 2022. Rich arc routing problem in city logistics: models and solution algorithms using a fluid queue-based time-dependent travel time representation. Transport. Res. Part B: Methodolog. 166, 143–182.
Lu, W., Xie, Y., Wang, W., Quadrifoglio, L., 2011. An analytical model to select the fleet size for MAST systems. In: 2011 IEEE Forum on Integrated and Sustainable Transportation Systems, pp. 152–158.
Ma, C., Wang, C., Xu, X., 2021. A multi-objective robust optimization model for customized bus routes. IEEE Trans. Intell. Transp. Syst. 22 (4), 2359–2370.
Madsen, O. B.G., Ravn, H.F., Rygaard, J.M., 1995. A heuristic algorithm for a dial-a-ride problem with time windows, multiple capacities, and multiple objectives. Ann. Oper. Res. 60 (1), 193–208.
Malucelli, F., Nonato, M., Gabriel Crainic, T., Guertin, F., 2001. Adaptive memory programming for a class of demand responsive transit systems. In: Voß, S., Daduna, J.R. (Eds.), Computer-Aided Scheduling of Public Transport. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 253–273.
Mancini, S., Ganasterer, M., 2022. Bundle generation for last-mile delivery with occasional drivers. Omega (Westport) 108, 102582.
Mishra, S., Mehran, B., 2024. Cost analysis of different vehicle technologies for semi-flexible transit operations. Transport. Res. Part D: Transp. Environ. 130, 104159.
Narayan, J., Cats, O., van Oort, N., Hoogendoorn, S.P., 2022. On the scalability of private and pooled on-demand services for urban mobility in amsterdam. Transport. Plann. Technol. 45 (1), 2–18.
Narayan, J., Cats, O., van Oort, N., Hoogendoorn, S., 2020. Integrated route choice and assignment model for fixed and flexible public transport systems. Transport. Res. Part C: Emerg. Technolog. 115, 102631.
Niu, H., Zhou, X., Tian, X., 2018. Coordinating assignment and routing decisions in transit vehicle schedules: a variable-splitting lagrangian decomposition approach for solution symmetry breaking. Transport. Res. Part B: Methodolog. 107, 70–101.
Nourbakhsh, S.M., Ouyang, Y., 2012. A structured flexible transit system for low demand areas. Transport. Res. Part B: Methodolog. 46 (1), 204–216.
Petit, A., Ouyang, Y., 2022. Design of heterogeneous flexible-route public transportation networks under low demand. Transport. Res. Part C: Emerg. Technolog. 138, 103612.
Qiu, F., Li, W., Zhang, J., 2014. A dynamic station strategy to improve the performance of flex-route transit services. Transport. Res. Part C: Emerg. Technolog. 48, 229–240.
Quadrifoglio, L., Dessouky, M.M., Ordóñez, F., 2008. Mobility allowance shuttle transit (MAST) services: MIP formulation and strengthening with logic constraints. Eur. J. Oper. Res. 185 (2), 481–494.
Quadrifoglio, L., Dessouky, M.M., Palmer, K., 2007. An insertion heuristic for scheduling mobility allowance shuttle transit (MAST) services. J. Schedul. 10 (1), 25–40.
Quadrifoglio, L., Hall, R.W., Dessouky, M.M., 2006. Performance and design of mobility allowance shuttle transit services: bounds on the maximum longitudinal velocity. Transport. Sci. 40 (3), 351–363.
Shahin, R., Hosteins, P., Pellegrini, P., Vandanjon, P.-O., Quadrifoglio, L., 2024. A survey of flex-route transit problem and its link with vehicle routing poblem. Transport. Res. Part C: Emerg. Technolog. 158, 104437.
Shang, H., Chang, Y., Huang, H., Zhao, F., 2022. Integration of conventional and customized bus services: an empirical study in Beijing. Phys. A 605, 127971.
Shen, C., Sun, Y., Bai, Z., Cui, H., 2021. Real-time customized bus routes design with optimal passenger and vehicle matching based on column generation algorithm. Phys. A 571, 125836.
Shrivastava, P., O'Mahony, M., 2006. A model for development of optimized feeder routes and coordinated schedules–a genetic algorithms approach. Transp. Policy (Oxf.) 13 (5), 413–425.
Sörensen, L., Bossert, A., Jokinen, J.-P., Schlüter, J., 2021. How much flexibility does rural public transport need? – Implications from a fully flexible DRT system. Transp. Policy (Oxf.) 100, 5–20.
Tong, L.C., Zhou, L., Liu, J., Zhou, X., 2017. Customized bus service design for jointly optimizing passenger-to-vehicle assignment and vehicle routing. Transport. Res. Part C: Emerg. Technolog. 85, 451–475.
Vansteenwegen, P., Melis, L., Aktaş, D., Montenegro, B. D.G., Sartori Vieira, F., Sörensen, K., 2022. A survey on demand-responsive public bus systems. Transport. Res. Part C: Emerg. Technolog. 137, 103573.
Wang, C., Ma, C., Xu, X., 2020. Multi-objective optimization of real-time customized bus routes based on two-stage method. Phys. A 537, 122774.
Wang, S., Zhu, X., Zhuo, S., Shang, P., 2024. Logistics vehicle routing optimisation with synchronised transfer. Transportmetr. A: Transp. Sci. 140, 1–36.
Yang, H., Zhang, Z., Fan, W., Xiao, F., 2021a. Optimal design for demand responsive connector service considering elastic demand. IEEE Trans. Intell. Transp. Syst. 22 (4), 2476–2486.
Yang, L., Shang, P., Yao, Y., Zeng, Z., 2022a. A dynamic scheduling process and methodology using route deviations and synchronized passenger transfers for flexible feeder transit services. Comput. Oper. Res. 146, 105917.
Yang, S., Ning, L., Shang, P., (Carol) Tong, L., 2020. Augmented lagrangian relaxation approach for logistics vehicle routing problem with mixed backhauls and time windows. Transport. Res. Part E: Logist. Transport. Rev. 135, 101891.

Yang, S., Ning, L., Tong, L.C., Shang, P., 2021b. Optimizing electric vehicle routing problems with mixed backhauls and recharging strategies in multi-dimensional representation network. Expert Syst. Appl. 176, 114804.
Yang, S., Ning, L., Tong, L.C., Shang, P., 2022b. Integrated electric logistics vehicle recharging station location–routing problem with mixed backhauls and recharging strategies. Transport. Res. Part C: Emerg. Technolog. 140, 103695.
Yao, Y., Zhu, X., Dong, H., Wu, S., Wu, H., Carol Tong, L., Zhou, X., 2019. Admm-based problem decomposition scheme for vehicle routing problem with time windows. Transport. Res. Part B: Methodolog. 129, 156–174.
Zhang, Y., Peng, Q., Yao, Y., Zhang, X., Zhou, X., 2019. Solving cyclic train timetabling problem through model reformulation: extended time-space network construct and alternating direction method of multipliers methods. Transport. Res. Part B: Methodolog. 128, 344–379.
Zhao, J., Sun, S., Cats, O., 2023. Joint optimisation of regular and demand-responsive transit services. Transportmetr. A: Transp. Sci. 19 (2), 1987580.